ਕੰਪਿਊਟਰ ਸਾਇੰਸ

(ਬਾਰ੍ਹਵੀਂ ਸ਼੍ਰੇਣੀ ਲਈ ਪਾਠ-ਪੁਸਤਕ)

ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ

ਸਾਹਿਬਜ਼ਾਦਾ ਅਜੀਤ ਸਿੰਘ ਨਗਰ

© ਪੰਜਾਬ ਸਰਕਾਰ

ਐਡੀਸ਼ਨ 2016 .... 1, 91,000 ਕਾਪੀਆਂ

All rights, including those of translation, reproduction
and annotation etc., are reserved by the
Punjab Government

ਚੇਤਾਵਨੀ

1. ਕੋਈ ਵੀ ਏਜੰਸੀ-ਹੋਲਡਰ ਵਾਧੂ ਪੈਸੇ ਵਸੂਲਣ ਦੇ ਮੰਤਵ ਨਾਲ ਪਾਠ-ਪੁਸਤਕਾਂ ਤੇ ਜਿਲਦ-ਸਾਜ਼ੀ ਨਹੀਂ ਕਰ ਸਕਦਾ। (ਏਜੰਸੀ-ਹੋਲਡਰਾਂ ਨਾਲ ਹੋਏ ਸਮਝੌਤੇ ਦੀ ਧਾਰਾ ਨੰ. 7 ਅਨੁਸਾਰ)
2. ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ ਦੁਆਰਾ ਛਪਾਈਆਂ ਅਤੇ ਪ੍ਰਕਾਸ਼ਿਤ ਪਾਠ-ਪੁਸਤਕਾਂ ਦੇ ਜਾਅਲੀ ਨਕਲੀ ਪ੍ਰਕਾਸ਼ਨਾਂ (ਪਾਠ-ਪੁਸਤਕਾਂ) ਦੀ ਛਪਾਈ, ਪ੍ਰਕਾਸ਼ਨ, ਸਟਾਕ ਕਰਨਾ, ਜਮ੍ਹਾਂ-ਖੋਰੀ ਜਾਂ ਵਿਕਰੀ ਆਦਿ ਕਰਨਾ ਭਾਰਤੀ ਦੰਡ ਪ੍ਰਣਾਲੀ ਦੇ ਅੰਤਰਗਤ ਫ਼ੌਜਦਾਰੀ ਜੁਰਮ ਹੈ।

   (ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ ਦੀਆਂ ਪਾਠ-ਪੁਸਤਕਾਂ ਬੋਰਡ ਦੇ 'ਵਾਟਰ ਮਾਰਕ' ਵਾਲੇ ਕਾਗਜ਼ ਉੱਪਰ ਹੀ ਛਪਵਾਈਆਂ ਜਾਂਦੀਆਂ ਹਨ।)

ਮੁੱਲ : ₹ 73.00

ਸਕੱਤਰ, ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ, ਵਿੱਦਿਆ ਭਵਨ, ਫੇਜ਼-8 ਸਾਹਿਬਜ਼ਾਦਾ ਅਜੀਤ ਸਿੰਘ ਨਗਰ-160062 ਰਾਹੀਂ ਪ੍ਰਕਾਸ਼ਿਤ ਅਤੇ **ਮੈਸ. : ਮੌਡੇਸਟ ਪ੍ਰਿੰਟਰਜ਼, ਜਲੰਧਰ ਦੁਆਰਾ ਛਾਪੀ ਗਈ।**

ਮੁੱਖ-ਬੰਧ

ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ ਆਪਣੀ ਸਥਾਪਨਾ ਦੇ ਸਮੇਂ ਤੋਂ ਹੀ ਸਕੂਲ ਪੱਧਰ ਦੇ ਪਾਠ-ਕ੍ਰਮਾਂ ਨੂੰ ਆਧੁਨਿਕ ਸੋਚ ਅਤੇ ਖੋਜ ਅਨੁਸਾਰ ਢਾਲਣ ਅਤੇ ਇਹਨਾਂ ਪਾਠ-ਕ੍ਰਮਾਂ ਅਨੁਸਾਰ ਪਾਠ-ਪੁਸਤਕਾਂ ਤਿਆਰ ਕਰਨ ਲਈ ਯਤਨਸ਼ੀਲ ਰਿਹਾ ਹੈ। ਰਾਸ਼ਟਰੀ ਪੱਧਰ ਤੇ ਵਰਤਮਾਨ ਸੋਚ ਨੂੰ ਮੁੱਖ ਰਖਦੇ ਹੋਏ ਬੋਰਡ ਨੇ ਪਾਠ-ਪੁਸਤਕਾਂ ਦੀ ਨਵ-ਰਚਨਾ ਦਾ ਵਿਸ਼ੇਸ਼ ਪ੍ਰੋਗਰਾਮ ਉਲੀਕਿਆ ਹੋਇਆ ਹੈ। ਇਹ ਪਾਠ-ਪੁਸਤਕ ਇਸੇ ਪ੍ਰੋਗਰਾਮ ਦੀ ਇੱਕ ਕੜੀ ਹੈ।

ਕੰਪਿਊਟਰ ਸਾਇੰਸ ਵਿਸ਼ੇ ਦੇ ਅਧਿਐਨ ਦੀ ਲੋੜ ਦਿਨੋ-ਦਿਨ ਵੱਧਦੀ ਜਾ ਰਹੀ ਹੈ ਕਿਉਂਕਿ ਵਿਗਿਆਨ ਅਤੇ ਤਕਨੀਕੀ ਉੱਨਤੀ ਦੇ ਯੁਗ ਵਿੱਚ ਹਰ ਕਾਰਜ ਖੇਤਰ ਵਿੱਚ ਕਾਰਜ ਕੁਸ਼ਲਤਾ ਵਧਾਉਣ ਲਈ ਕੰਪਿਊਟਰ ਸਿੱਖਿਆ ਦੀ ਲੋੜ ਹੈ। ਸੂਚਨਾ ਤਕਨਾਲੋਜੀ ਦੇ ਪ੍ਰਸਾਰ ਅਤੇ ਸੰਚਾਰ ਦੀ ਤਰੱਕੀ ਨਾਲ਼, ਹਰ ਵਿਭਾਗ ਵਿੱਚ ਕੰਪਿਊਟਰੀਕਰਨ ਕੀਤਾ ਜਾ ਰਿਹਾ ਹੈ। ਵੱਖ-ਵੱਖ ਮਹਿਕਮਿਆਂ ਤੋਂ ਲੋੜੀਂਦੀ ਸੂਚਨਾ ਪ੍ਰਾਪਤ ਕਰਨ, E-Ticketing ਅਤੇ ਇੰਟਰਨੈੱਟ ਦੀ ਵਰਤੋਂ ਦੀ ਸੁਵਿਧਾ ਲੈਣ ਲਈ ਕੰਪਿਊਟਰ ਸਿੱਖਿਆ ਹਰ ਵਿਅਕਤੀ ਲਈ ਲਾਜ਼ਮੀ ਹੋ ਗਈ ਹੈ।

ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ ਨੇ ਅਜਿਹੀਆਂ ਜ਼ਰੂਰਤਾਂ ਨੂੰ ਮੁੱਖ ਰੱਖਦੇ ਹੋਏ, ਪੰਜਾਬ ਸਰਕਾਰ ਦੇ ਦਿਸ਼ਾ-ਨਿਰਦੇਸ਼ ਅਨੁਸਾਰ ਕੰਪਿਊਟਰ ਸਾਇੰਸ ਵਿਸ਼ਾ ਐਲੀਮੈਂਟਰੀ ਅਤੇ ਸੈਕੰਡਰੀ ਪੱਧਰ ਤੇ ਲਾਜ਼ਮੀ ਵਿਸ਼ੇ ਵਜੋਂ ਲਾਗੂ ਕੀਤਾ ਹੈ। ਇਸ ਵਿਸ਼ੇ ਦਾ ਪਾਠਕ੍ਰਮ ਕੁਝ ਸਰਕਾਰੀ ਸਕੂਲਾਂ ਵਿੱਚ ਪਿਕਟਸ ਦੁਆਰਾ ਪਹਿਲਾਂ ਹੀ ਪੜ੍ਹਾਇਆ ਜਾ ਰਿਹਾ ਸੀ। ਅਧਿਆਪਕਾਂ ਦੀ ਮੰਗ ਅਨੁਸਾਰ ਪਾਠਕ੍ਰਮ ਵਿੱਚ ਲੋੜੀਂਦੀ ਸੋਧ ਕਰਕੇ ਪੁਸਤਕ ਪੰਜਾਬੀ ਮਾਧਿਅਮ ਵਿੱਚ ਸੋਧੇ ਹੋਏ ਰੂਪ ਵਿੱਚ ਛਪਾਈ ਗਈ ਹੈ। ਇਸ ਵਿਸ਼ੇ ਸੰਬੰਧੀ ਸਾਰੀ ਜਾਣਕਾਰੀ ਇਸ ਪੁਸਤਕ ਵਿੱਚ ਉਪਲੱਬਧ ਕਰਵਾਉਣ ਦੀ ਕੋਸ਼ਿਸ਼ ਕੀਤੀ ਗਈ ਹੈ। ਆਸ ਹੈ ਇਹ ਪੁਸਤਕ ਵਿਸ਼ੇ ਦੇ ਵਿਦਿਆਰਥੀਆਂ ਅਤੇ ਅਧਿਆਪਕਾਂ ਲਈ ਉਪਯੋਗੀ ਸਿੱਧ ਹੋਵੇਗੀ। ਪੁਸਤਕ ਨੂੰ ਚੰਗੇਰਾ ਬਣਾਉਣ ਲਈ ਖੇਤਰ ਵਿੱਚੋਂ ਆਏ ਸੁਝਾਵਾਂ ਦਾ ਸਤਿਕਾਰ ਕੀਤਾ ਜਾਵੇਗਾ

ਪੰਜਾਬ ਸਕੂਲ ਸਿੱਖਿਆ ਬੋਰਡ

ਵਿਸ਼ਾ ਸੂਚੀ

ਲੜੀ ਨੰ : ਅਧਿਆਇ		ਪੰਨਾ
1.	**ਗਿਆਰ੍ਹਵੀਂ ਕਲਾਸ ਦੀ ਦੁਹਰਾਈ**	1-10
	1.0 ਸੀ ਭਾਸ਼ਾ ਵਿੱਚ ਪ੍ਰੋਗਰਾਮਿੰਗ ਲਈ ਭੂਮਿਕਾ	
	1.1 ਕਮਪਾਇਲਰ ਨਿਰਦੇਸ਼	
	1.2 ਐਰੇ: ਐਰੇ ਵਿੱਚ ਡਾਟਾ ਭਰਨਾ, ਐਰੇ ਦੇ ਮੁੱਲਾਂ ਦੀ ਪਹੁੰਚ ਕਰਨੀ, ਦੋ ਡਾਇਮੈਂਸ਼ਨਲ ਐਰੇ ਦੀ ਨਿਰਦੇਸ਼ਨ	
2.	**ਸਟਰਿੰਗ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ**	11-31
	2.0 ਭੂਮਿਕਾ	
	2.1 ਡਿਕਲੇਰੇਸ਼ਨ ਅਤੇ ਸਟਰਿੰਗ ਵੇਰੀਏਬਲ	
	2.2 ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਦੀ ਪ੍ਰਿੰਟਿਗ ਪਰਿਭਾਸ਼ਤ ਕਰਨੀ	
	2.3 ਸਟਰਿੰਗ ਪੜ੍ਹਨਾ ਅਤੇ ਲਿਖਣਾ: ਸਟਰਿੰਗ ਪੜ੍ਹਨਾ,ਸਟਰਿੰਗ ਲਿਖਣਾ, ਮੋਨੀਟਰ ਤੋਂ ਸਟਰਿੰਗ ਪੜ੍ਹਨਾ	
	2.4 ਸਧਾਰਣ ਇਨਪੁੱਟ ਆਉਟਪੁੱਟ	
	2.5 ਸਟਰਿੰਗ ਉਪਰੇਸ਼ਨਜ਼	
3.	**ਯੂਜਰ ਡਿਫਿਇੰਨਡ ਫੰਕਸ਼ਨਜ਼**	32-46
	3.0 ਭੂਮਿਕਾ	
	3.1 ਬਿਲਟ ਇੰਨ ਫੰਕਸ਼ਨ/ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ : ਯੂਜਰ ਡੀਫਾਈਨ ਫੰਕਸ਼ਨ, ਫੰਕਸ਼ਨ ਦੀ ਜ਼ਰੂਰਤ	
	3.2 ਆਰਗੂਮੈਂਟ ਅਤੇ ਪੈਰਾਮੀਟਰ	
	3.3 ਕੋਡ ਲਾਈਨ	
	3.4 ਮਲਟੀਫੰਕਸ਼ਨ ਪ੍ਰੋਗਰਾਮ	
	3.5 ਫੰਕਸ਼ਨ ਦੀ ਬਣਤਰ	
	3.6 ਰਿਟਰਨ ਮੁੱਲ ਅਤੇ ਉਹਨਾਂ ਦੇ ਟਾਈਪਸ: ਸਿੰਪਲ ਰਿਟਰਨ	
	3.7 ਫੰਕਸ਼ਨ ਨੂੰ ਕਾਲ ਕਰਨਾ	
	3.8 ਸਟੋਰੇਜ ਸ਼੍ਰੇਣੀਆਂ	
	3.9 ਸਟੇਟਿਕ ਵੇਰੀਏਬਲ	
	3.10 ਐਕਸਟਰਨਲ ਵੇਰੀਏਬਲ	
4.	**ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ-1**	47-61

4.0 ਭੂਮਿਕਾ

4.1 ਫਾਈਲ ਦੀਆਂ ਕਿਸਮਾਂ :ਆਡੀਓ ਫਾਈਲ, ਵੀਡੀਓ ਫਾਈਲ, ਪਿਕਚਰ ਫਾਈਲ

4.2 ਕਲੈਕਸ਼ਨ, ਪ੍ਰੋਜੈਕਟ ਅਤੇ ਮੂਵੀਜ਼ ਨੂੰ ਜਾਨਣਾ

4.3 ਵੀਡੀਓ ਕੈਪਚਰ ਡੀਵਾਈਸ

4.4 ਆਡੀਓ ਕੈਪਚਰ ਡੀਵਾਈਸ

4.5 ਕੈਪਚਰ ਡੀਵਾਈਸ ਨੂੰ ਜੋੜਨ ਲਈ ਪੇਰੀਫੀਰਲ : ਵੈਬ ਕੈਮਰਾ, ਵੀ.ਸੀ ਆਰ, ਡੀਜ਼ੀਟਲ ਕੈਮਰਾ, ਸਾਊਂਡ ਕਾਰਡ

4.6 ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਇੰਨਟਰਫੇਸ

4.7 ਨਿਊ ਪ੍ਰੋਜੈਕਟ

4.8 ਪਿਕਚਰ ਇੰਪੋਰਟ ਕਰਨਾ

4.9 ਆਡੀਓ ਜਾ ਵੀਡੀਓ ਨੂੰ ਸਪੱਲਿਟ ਕਰਨਾ

5. **ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ-2** 62-77

5.0 ਭੂਮਿਕਾ

5.1 ਟਾਈਮਲਾਈਨ

5.2 ਸਟੋਰੀ ਬੋਰਡ ਵਿੱਚ ਵੀਡੀਓ ਕਲਿੱਪ ਜੋੜਨਾ

5.3 ਕਲਿਪਸ ਦੀ ਟਰੀਮਿੰਗ

5.4 ਮੂਵੀ ਨੂੰ ਐਡਿਟ ਕਰਨਾ

5.5 ਵੀਡੀਓ ਟਰਾਂਜਿਸ਼ਨ

5.6 ਵੀਡੀਓ ਇਫੈਕਟਸ

5.7 ਟਾਈਟਲ ਅਤੇ ਕਰੈਟਿਡਸ

5.8 ਪ੍ਰੀ ਵਿਊ ਦੇਖਣਾ

5.9 ਆਡੀਓ ਨੂੰ ਪਿਕਚਰ ਨਾਲ ਅਲਾਈਨ ਕਰਨਾ

5.10 ਐਡਿਟਿੰਗ ਆਡੀਓ

6. ਨੈਟਵਰਕਿੰਗ ਸੰਕਲਪ 78-92

6.0 ਭੂਮਿਕਾ

6.1 ਨੈਟਵਰਕ ਕੀ ਹੁੰਦਾ ਹੈ; ਨੈਟਵਰਕਿੰਗ ਲਈ ਲੋੜ, ਨੈਟਵਰਕਿੰਗ ਟਰਮੀਨੋਲੋਜੀ

6.2 ਨੈਟਵਰਕ ਦੀਆਂ ਕਿਸਮਾਂ

6.3 ਨੈਟਵਰਕ ਟੋਪੋਲੋਜੀ : ਟੋਪੋਲੋਜੀ ਕੀ ਹੁੰਦੀ ਹੈ, ਭੌਤਿਕ ਟੋਪੋਲੋਜੀ ਦੀਆਂ ਮੁੱਖ ਕਿਸਮਾਂ, ਲੀਨੀਅਰ ਬੱਸ ਟੋਪੋਲੋਜੀ, ਸਟਾਰ ਟੋਪੋਲੋਜੀ, ਟ੍ਰੀ-ਟੋਪੋਲੋਜੀ, ਮੈਸ਼ ਟੋਪੋਲੋਜੀ, ਪਾਰਸ਼ਲ ਮੈਸ਼ ਟੋਪੋਲੋਜੀ

6.4 ਪ੍ਰੋਟੋਕਲਜ਼ : ਯੂਨੀਫਾਰਮ ਰਿਸੋਰਸ ਲੋਕੇਟਰ (ਯੂ.ਆਰ.ਐਲ) ਫਾਈਲ ਟਰਾਂਸਫਰ ਅਤੇ ਹਾਈਪਰਟੈਕਸਟ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ (ਫਾਈਲ ਟ੍ਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ, ਹਾਈਪਰ ਟੈਕਸਟ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ, ਪ੍ਰਾਈਵੇਟ ਨੈਟਵਰਕਸ)

6.5 ਚੈਨਲ ਦੀ ਵਿਚਾਰਧਾਰਾ :ਸੰਚਾਰ ਚੈਨਲ ਕੀ ਹੁੰਦਾ ਹੈ, ਡਾਟਾ ਬਦਲੀ ਦਰ

6.6 ਸੰਚਾਰ ਮਾਧਿਅਮ : ਗਾਈਡਿਡ ਮੀਡੀਆ, ਅਨਗਾਈਡਿਡ ਮੀਡੀਆ, ਆਰ.ਐਫ ਪ੍ਰਸਾਰ

6.7 ਨੈਟਵਰਕਿੰਗ ਡਿਵਾਇਸ : ਹੱਬ ਅਤੇ ਮੋਡਮ

7. ਲਾਈਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨਜ਼ 93-101

7.1 ਫੰਕਸ਼ਨ

7.2 ਸਟਰਲੇਨ ਫੰਕਸ਼ਨ (Strlen())

7.3 ਸਟਰੇਪੀ ਫੰਕਸ਼ਨ (Strepy())

7.4 ਸਟਰੀਟ ਫੰਕਸ਼ਨ (Streat())

7.5 ਸਟਰੈਮਪ ਫੰਕਸ਼ਨ (Stremp())

7.6 ਗੈਟਚਰ ਫੰਕਸ਼ਨ (getchar())

7.7 ਪੁਟਚਰ ਫੰਕਸ਼ਨ (putchar())

7.8 ਪ੍ਰਿੰਟਐਫ ਫੰਕਸ਼ਨ (Printf())

7.9 ਸਕੈਨਐਫ ਫੰਕਸ਼ਨ (Scanf())

7.10 ਕੋਸ ਫੰਕਸ਼ਨ (Cos())

7.11 ਐਕਸਪ ਫੰਕਸ਼ਨ (Exp())

7.12 ਪੋਵ ਫੰਕਸ਼ਨ (Pow())

7.13 ਸਕਰਟ ਫੰਕਸ਼ਨ (Sqrt())

7.14 ਐਫਮੋਡ ਫੰਕਸ਼ਨ (fmod())
7.15 ਫੇਬਸ ਫੰਕਸ਼ਨ (Fabs())
7.16 ਸੀਲ ਫੰਕਸ਼ਨ (Ceil())
7.17 ਫਲੋਰ ਫੰਕਸ਼ਨ (floor())
7.18 ਇਜ਼ਲੋਅਰ ਅਤੇ ਇਜ਼ਅਪਰ ਫੰਕਸ਼ਨ (islower()) and (isupper())
7.19 ਟੂਲੋਅਰ ਅਤੇ ਟੂਅਪਰ ਫੰਕਸ਼ਨ (tolower()) and (toupper())
7.20 ਫੰਕਸ਼ਨਾਂ ਦੇ ਲਾਭ

8. ਈ-ਗਵਰਨੈਂਸ 102-115
8.1 ਈ–ਗਵਰਨੈਂਸ ਦਾ ਪ੍ਰਯੋਗ : ਘੱਟ–ਖਰਚ, ਤੇਜ਼ ਰਫ਼ਤਾਰ, ਕਿਤੇ ਵੀ ਕਿਸੇ ਵੀ ਟਾਈਮ, ਆਸਾਨ ਪ੍ਰਬੰਧ
8.2 ਈ–ਗਵਰਨੈਂਸ ਦੇ ਖੇਤਰ
8.3 ਈ–ਗਵਰਨੈਂਸ ਦੇ ਅਲੱਗ–ਅਲੱਗ ਖੇਤਰ : ਇੰਟਰਨੈਟ ਬੈਂਕਿੰਗ, ਆਨ–ਲਾਈਨ ਰੇਲਵੇ ਅਤੇ ਹਵਾਈ ਟਿਕਟਿੰਗ (ਆਈ–ਟਿਕਟਿੰਗ, ਈ ਇਕਟਿੰਗ), ਪਾਸਪੋਰਟ ਆਫਿਸ ਕੰਮ–ਕਾਜ
8.4 ਸੁਵਿਧਾ ਸੈਂਟਰ : ਸੁਵਿਧਾ

ASC II Character Set 116
ਸ਼ਬਦਾਵਲੀ 117-123

ਸੰਪਾਦਕੀ ਕਮੇਟੀ

ਤਰੁਨ ਅਗਰਵਾਲ
ਸਰਕਾਰੀ ਕੰਨਿਆ ਮਲਟੀਪਰਪਜ਼ ਸੀ. ਸੈ. ਸਕੂਲ, ਮਾਡਲ ਟਾਊਨ, ਪਟਿਆਲਾ

ਜਗਪ੍ਰੀਤ ਸਿੰਘ ਸਿੱਧੂ
ਸਰਕਾਰੀ ਸੀ. ਸੈ. ਸਕੂਲ ਨੰਦਪੁਰ ਕੇਸ਼ੋ, ਪਟਿਆਲਾ

ਗਗਨਦੀਪ ਸਿੰਘ
ਸਰਕਾਰੀ ਮਾਡਲ ਸੀ. ਸੈ. ਸਕੂਲ ਫੇਜ਼ 3 ਬੀ-1, ਮੁਹਾਲੀ

ਵਿਕਰਮ ਪਾਲ
ਸ਼ਹੀਦ ਲਾਂਸ ਨਾਇਕ ਰਣਜੋਧ ਸਿੰਘ ਸਰਕਾਰੀ ਸੀ. ਸੈ. ਸਕੂਲ, ਤਿਊੜ, ਐਸ. ਏ. ਐਸ. ਨਗਰ, ਮੁਹਾਲੀ

ਸ਼ਿਵਾਨੀ ਨੰਦਾ
ਸਰਕਾਰੀ ਸੀ. ਸੈ. ਸਕੂਲ, ਬਹਾਦਰਗੜ੍ਹ, ਪਟਿਆਲਾ

ਪ੍ਰੀਤੀ ਚੱਮ
ਸਰਕਾਰੀ ਹਾਈ ਸਕੂਲ, ਬੁਲੇਪੁਰ, ਲੁਧਿਆਣਾ

ਹਰਪ੍ਰੀਤ ਸਿੰਘ
ਸਰਕਾਰੀ ਹਾਈ ਸਕੂਲ, ਭਮੱਦੀ, ਲੁਧਿਆਣਾ

ਗੌਰਵ ਸੂਦ
ਸਰਕਾਰੀ ਸੀ. ਸੈ. ਸਕੂਲ, ਮਾਣਕ ਮਾਜਰਾ, ਲੁਧਿਆਣਾ

ਬਿੰਦੂ
ਸਰਕਾਰੀ ਮਾਡਲ ਸੀ. ਸੈ. ਸਕੂਲ ਫੇਜ਼ 3 ਬੀ-1, ਮੁਹਾਲੀ

ਹਰਪ੍ਰੀਤ ਸਿੰਘ
ਸਰਕਾਰੀ ਕੰਨਿਆ ਹਾਈ ਸਕੂਲ, ਤਪਾ, ਬਰਨਾਲਾ

ਪੁਨਰ ਮੁਲਾਂਕਣ ਅਤੇ ਤਸਦੀਕ ਕਰਤਾ

ਸੀ. ਪੀ. ਕੰਬੋਜ
ਸਰਕਾਰੀ ਹਾਈ ਸਕੂਲ, ਕਿੜਿਆਂ ਵਾਲਾ, ਫਿਰੋਜ਼ਪੁਰ

ਧਰਮਜੀਤ ਸਿੰਘ
ਸਰਕਾਰੀ ਹਾਈ ਸਕੂਲ, ਗਹਿਰੀ ਦੇਵੀ ਨਗਰ, ਬਠਿੰਡਾ

ਅਨੁਵਾਦਕ

ਸੀ. ਪੀ. ਕੰਬੋਜ

ਸੰਧਕ

ਸ੍ਰੀਮਤੀ ਨਵਦੀਪ ਗਰੇਵਾਲ
ਸ.ਸ.ਸ.ਸ. ਮੋਹਾਲੀ (ਐਸ.ਏ.ਐਸ. ਨਗਰ)

ਅਧਿਆਇ

1

Review of Class - XI

ਸੀ ਭਾਸ਼ਾ ਵਿਚ ਪ੍ਰੋਗਰਾਮਿੰਗ ਲਈ ਭੂਮਿਕਾ *(Introduction to Programming in C Language)*

C ਇਕ ਆਮ ਮੰਤਵ ਦੀ ਪ੍ਰੋਗਰਾਮਿੰਗ ਭਾਸ਼ਾ ਹੈ। ਇਹ ਪ੍ਰੋਗਰਾਮਿੰਗ ਭਾਸ਼ਾਵਾਂ ਜਿਵੇਂ B ਭਾਸ਼ਾ ਅਤੇ BCPL ਦਾ ਪਰਿਣਾਮ *(outcome)* ਹੈ। ਸਮੂਹ ਤਿੰਨ ਪ੍ਰੋਗਰਾਮਿੰਗ ਭਾਸ਼ਾਵਾਂ ਬੈਲ ਲੈਬਾਰਟਰੀ USA ਵਿਚ ਵਿਕਸਿਤ ਹੋਈਆਂ ਸਨ। ਇਹ 1978 ਵਿਚ ਡੇਨਿਸ ਰਿਚ ਅਤੇ ਬਰੇਨ ਕੇਰਮਿੰਘਨ *(Dennis Ritchie & Brain Kerninghan)* ਦੁਆਰਾ ਲਿਖੀਆਂ ਗਈਆਂ ਸਨ। ਸੀ **'C'** ਨੂੰ ਦੋਵੇਂ (1) ਸਿਸਟਮ ਪ੍ਰੋਗਰਾਮ ਅਤੇ (2) ਐਪਲੀਕੇਸ਼ਨ-ਪ੍ਰੋਗਰਾਮ *(application program)* ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਇਹ ਹਾਈ ਅਤੇ ਲੋਅ ਲੈਵਲ ਦੇ ਦਰਮਿਆਨੀ ਭਾਸ਼ਾ ਹੈ ਇਸ ਲਈ ਇਸ ਨੂੰ ਮਿਡਲ-ਲੈਵਲ *(middle level)* ਭਾਸ਼ਾ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। C ਚਿੰਨ੍ਹ ਸੈੱਟ *(character set)* ਵਰਣਮਾਲਾ *(alphabets)*, ਡਿਜੀਟਸ *(digits)* ਅਤੇ ਵਿਸ਼ੇਸ਼ ਚਿੰਨ੍ਹਾਂ *(special symbols)* ਦੇ ਬਣੇ ਹੁੰਦੇ ਹਨ। C ਵਿਚ 91 ਚਿੰਨ੍ਹ *(character)* ਹੁੰਦੇ ਹਨ। C ਟੋਕਨ *(token)* C ਪ੍ਰੋਗਰਾਮ ਦੇ ਬੁਨਿਆਦੀ ਯੂਨਿਟ ਹੁੰਦੇ ਹਨ ਅਤੇ ਉਹ ਕੀ-ਵਰਡਸ, ਆਈਡੈਂਨਟੀਫਾਈਰਸ *(identifiers)*, ਕਾਂਨਸਟੈਂਟ ਸਟਰਿੰਗਜ਼ *(constants strings)*, ਅੱਪਰੇਟਰਸ ਅਤੇ ਵਿਸ਼ੇਸ਼ ਚਿੰਨ੍ਹਾਂ *(special symbols)* ਵਿਚ ਸ਼੍ਰੇਣੀਬੱਧ ਹੁੰਦੇ ਹਨ। C ਵਿਚ ਸਾਰੇ ਕੀ-ਵਰਡਸ ਦੇ ਇਕ ਜਾਂ ਵਧੇਰੇ ਨਿਸ਼ਚਿਤ ਅਰਥ ਹੁੰਦੇ ਹਨ ਅਤੇ ਉਹ ਬਦਲੇ ਨਹੀਂ ਜਾ ਸਕਦੇ। ਕੀ-ਵਰਡਸ ਛੋਟੇ ਅੱਖਰਾਂ ਵਿਚ ਲਿਖੇ ਜਾਣੇ ਚਾਹੀਦੇ ਹਨ। ਆਈਡੈਂਨਟੀਫਾਈਰਸ *(identifiers)* ਉਹ ਨਾਮ ਹਨ ਜੋ ਪ੍ਰੋਗਰਾਮ ਦੇ ਅੰਸ਼ਾਂ ਨੂੰ ਦਿਤੇ ਜਾਂਦੇ ਹਨ ਜਿਵੇਂ ਕਿ ਵੇਰੀਏਬਲਸ *(variables)*, ਐਰੇਜ਼*(arrays)* ਅਤੇ ਫੰਕਸ਼ਨਸਜ਼ *(functions)*। ਆਈਡੈਂਨਟੀਫਾਈਰਸ *(identifiers)* ਬਦਲੇ ਨਹੀਂ ਜਾ ਸਕਦੇ। ਸਮੂਹ ਵੇਰੀਏਬਲਸ *(variables)* ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਵਰਤਣ ਤੋਂ ਪਹਿਲਾਂ ਉਨ੍ਹਾਂ ਦੀਆਂ ਕਿਸਮਾਂ ਲਈ ਡਿਕਲੇਅਰ *(declare)* ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਕਾਂਨਸਟੈਂਟ *(constant)* ਇਕ ਮਾਤਰਾ (quantity) ਹੁੰਦੀ ਹੈ ਜੋ ਕਿ ਇਕ ਪ੍ਰੋਗਰਾਮ ਲਾਗੂਕਰਨ *(execution)* ਦੇ ਦੌਰਾਨ ਨਹੀਂ ਬਦਲਦੀ ਜਦ ਕੀ ਫਲੋਟਿੰਗ *(floating)* ਬਦਲ ਜਾਂਦੀ ਹੈ। ਹਰ ਇਕ ਹਿਦਾਇਤ ਅਰਧਵਿਰਾਮ-ਚਿੰਨ੍ਹ *(semi colon character)* ";" ਦੇ ਨਾਲ ਖਤਮ ਹੁੰਦੀ ਹੈ ਅਤੇ ਟਿੱਪਣੀਆਂ *(comments)* ਕਿਤੇ ਵੀ ਲਿਖੀਆਂ ਜਾ ਸਕਦੀਆਂ ਹਨ। ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਸਮੂਹ ਸ਼ਬਦ ਟੈਬ ਜਾਂ ਵਿਰਾਮ ਚਿੰਨ੍ਹ *(punctuation character)* ਘਟੋ ਘਟ ਇਕ space / tab ਦੁਆਰਾ ਇਕ ਦੂਜੇ ਤੋਂ ਵੱਖ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ। ਹਰ ਇਕ C ਪ੍ਰੋਗਰਾਮ ਵਿਚ main() ਫੰਕਸ਼ਨ ਦੀ ਲੋੜ ਉਸ ਸਥਾਨ ਤੇ ਹੁੰਦੀ ਹੈ ਜਿੱਥੇ ਪ੍ਰੋਗਰਾਮ ਲਾਗੂ-ਕਰਨ ਤੇ ਸ਼ੁਰੂ ਹੁੰਦਾ ਹੈ। **ਫੰਕਸ਼ਨ (function)** ਦੀ ਸ਼ੁਰੂਆਤ *execution function opening brace* ਤੋਂ ਸ਼ੁਰੂ ਅਤੇ *closing brace* ਤੇ ਖਤਮ ਹੁੰਦੀ ਹੈ। ਸਾਨੂੰ ਇਸ ਗੱਲ ਨੂੰ ਯਕੀਨੀ ਬਣਾਉਣਾ ਚਾਹੀਦਾ ਹੈ ਕਿ # ਦੀ ਵਰਤੋਂ ਨਾਲ ਹੈਡਰ ਫਾਈਲਾਂ ਸ਼ਾਮਲ ਕੀਤੀਆਂ ਜਾਣ ਜਦੋਂ ਕਿ ਪ੍ਰੋਗਰਾਮ ਵਿਸ਼ੇਸ਼ ਨਾਵਾਂ ਅਤੇ ਫੰਕਸ਼ਨਸ *(functions)* ਦਾ ਹਵਾਲਾ ਦਿੰਦਾ ਹੋਵੇ ਕਿ ਇਹ ਪਰਿਭਾਸ਼ਤ *(define)* ਨਹੀਂ ਕਰ ਸਕਦਾ।

ਕੰਪਾਈਲਰ-ਨਿਰਦੇਸ਼ *(Compiler directives)*

ਜਿਵੇਂ ਕਿ ਪਰਿਭਾਸ਼ਾ *(definition)* ਅਤੇ ਵਿਸ਼ੇਸ਼ ਹਿਦਾਇਤਾਂ ਸ਼ਾਮਲ ਹੁੰਦੀਆਂ ਹਨ ਜੋ ਕੰਪਾਈਲਰ *(Compiler)* ਨੂੰ ਇਕ ਪ੍ਰੋਗਰਾਮ ਦੇ ਕੰਪਾਈਲ *(Compile)* ਕਰਨ ਲਈ ਮਦਦ ਕਰਦੇ ਹਨ। ਉਹ ਅਰਧ ਵਿਰਾਮ *(semi colon)* ਦੇ ਨਾਲ ਖਤਮ ਨਹੀਂ ਹੁੰਦੇ। ਚਿੰਨ੍ਹ # ਜੋ ਕੰਪਾਈਲਰ *(compiler)* ਦਾ ਹੁੰਦਾ ਹੈ ਲਾਈਨ ਦੇ ਪਹਿਲੇ ਕਾਲਮ ਉਤੇ ਸ਼ੁਰੂ ਹੁੰਦਾ ਹੈ। ਜਦੋ ਬਰੇਸਿਸ (braces) ਨੂੰ ਸਟੇਟਮੇਂਟ *(statement)* ਬਣਾਉਣ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਇਸ ਗੱਲ ਨੂੰ ਯਕੀਨੀ ਬਣਾਓ ਕਿ *opening brace* ਦਾ ਪੂਰਕ *closing brace* ਹੋਵੇ। ਇਨਪੁਟ ਡਿਵਾਈਸ *(input device)* ਤੋਂ ਡਾਟਾ ਪੜ੍ਹਣ ਲਈ ਇਨਪੁਟ ਸਟੇਟਮੈਂਟਸ *(input statements)* ਵਰਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ਅਤੇ ਪ੍ਰੋਸੈਸ ਡਾਟਾ ਨੂੰ ਇਨਪੁਟ ਡਿਵਾਈਸ *(input device)* ਤੋਂ ਆਊਟਪੁਟ ਡਿਵਾਈਸ *(output device)* ਪ੍ਰਿੰਟ ਕਰਨ ਲਈ ਆਊਟਪੁਟ ਸਟੇਟਮੈਂਟਸ *(output statements)* ਵਰਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ। ਇਹ ਇਨਪੁਟ/ਆਊਟਪੁਟ (I/O) ਸਟੇਟਮੈਂਟਸ ਫਾਰਮੈਟਿਡ *(formatted)* ਜਾਂ ਗੈਰ-ਫਾਰਮੈਟਿਡ (non formatted) ਹੋ ਸਕਦੀਆਂ ਹਨ। ਗੈਰ-ਫਾਰਮੈਟਿਡ ਇਨਪੁਟ/ਆਊਟਪੁਟ (I/O) ਸਟੇਟਮੈਂਟਸ *(statements)* ਵਿਚ ਡਾਟਾ ਟਾਈਪ ਦੀ ਕੋਈ ਵਸ਼ਿਸ਼ਠਤਾ (specification) ਨਹੀਂ ਹੁੰਦੀ ਅਤੇ ਜਿਸ ਢੰਗ ਨਾਲ ਉਹ ਪੜ੍ਹੇ ਜਾਂਦੇ ਹਨ ਜਾਂ ਲਿਖੇ ਜਾਂਦੇ ਹਨ, ਦਾ ਵਰਣਨ ਨਹੀਂ ਕੀਤਾ ਗਿਆ ਹੁੰਦਾ। ਪਰੰਤੂ ਫਾਰਮੈਟਿਡ ਇਨਪੁਟ/ਆਊਟਪੁਟ (I/O) ਸਟੇਟਮੈਂਟਸ *(statements)* ਇਹ ਨਿਸ਼ਚਿਤ ਕਰਦੀਆਂ ਹਨ ਕਿ ਇਨਪੁਟ ਡਾਟਾ ਕਿਸ ਟਾਈਪ ਦਾ ਹੈ ਅਤੇ ਪ੍ਰਦਰਸ਼ਿਤ ਕੀਤਾ ਜਾ ਰਿਹਾ ਇਨਪੁਟ ਡਾਟਾ ਉਸੇ ਢੰਗ ਨਾਲ ਹੋ ਰਿਹਾ ਹੈ

ਜਿਸ ਤਰ੍ਹਾਂ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ। scanf( ) ਅਤੇ printf( ) ਦੋ ਫੰਕਸ਼ਨਸ *(functions)* ਹਨ ਜੋ ਡਾਟਾ ਪੜ੍ਹਨ ਅਤੇ ਪ੍ਰਿੰਟ ਕਰਨ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨੇਂ ਉਹ ਹਰ ਟਾਈਪ ਦੇ ਡਾਟਾ ਦੀਆਂ ਕਿਸਮਾਂ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ ਜਿਵੇਂ int, char, float, double, ਆਦਿ। ਆਮ ਤੋਰ ਤੇ ਇਹ ਸੰਭਵ ਨਹੀਂ ਹੈ ਕਿ ਬਹੁ-ਸ਼ਬਦੀ ਸਟਰਿੰਗ ਨੂੰ scanf () ਫੰਕਸ਼ਨ ਨਾਲ ਪੜ੍ਹਿਆ ਜਾਵੇ। Escape sequence ਦੀ ਵਰਤੋਂ ਨਾਲ ਅਸੀ ਆਪਣੀਆਂ ਜਰੂਰਤਾਂ ਅਨੁਸਾਰ ਆਉਟਪੁਟ ਨੂੰ ਕੰਟਰੋਲ ਕਰ ਸਕਦੇ ਹਾਂ।

\n	The next line character
\t	Tab position
\a	Beep sound
\b	Shift cursor one position left
\r	Carriage return

ਡਾਟਾ ਟਾਈਪ ਆਗਿਆਯੋਗ (permissible) ਅੱਪਰੇਸ਼ਨ ਦੇ ਨਾਲ ਮੁਲ (value) ਦਾ ਇਕ ਸੈੱਟ ਹੁੰਦਾ ਹੈ। ਸਾਧਾਰਣ ਡਾਟਾ ਟਾਈਪ ਵਿਚ ਪੂਰਣ ਅੰਕ, (ਸੰਖੇਪ, int ਜਾਂ long) ਅਸਲ ਅੰਕ (ਅਸਥਾਈ, ਦੁਗਣਾ) ਅਤੇ ਚਿੰਨ੍ਹ ਕੈਰ (char) ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ। ਹਰ ਡਾਟਾ ਟੈਕਸਟ *(text)* ਜਿਹੜਾ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਵਰਤਿਆ ਜਾਣਾ ਹੁੰਦਾ ਹੈ ਨੂੰ ਡਿਕਲੇਅਰ ਅਤੇ ਆਰੰਭ ਕਰਨ ਦੀ ਲੋੜ ਹੁੰਦੀ ਹੈ। ਬੈਕਸਲੈਸ਼ (\) ਚਿੰਨ੍ਹ ਨਿਰੰਤਰ ਇਕ ਵਿਸ਼ੇਸ ਕੰਮ ਕਰਦੇ ਹਨ। ਇਹ ਬੈਕਸਲੈਸ਼ (\) ਦੇ ਨਾਲ ਸ਼ੁਰੂ ਹੁੰਦੇ ਹਨ। ਦੋਵੇਂ ਬੈਕਸਲੈਸ਼ ਅਤੇ ਸੰਬੰਧਿਤ ਚਿੰਨ੍ਹ ਇਕਹਰੇ ਕੋਟਸ (quotes) ' ' ਦੇ ਨਾਲ ਨੱਥੀ ਹੁੰਦੇ ਹਨ।

(The size and range of each data type is given in the table below)

Data Type	Label	Size (Bits)	Range
Char or Signed Char	Char	8	-128 to 127
Unsigned Char		8	0 to 255
Int or Signed int	Int	16	-32768 to 32767
Unsigned int		16	0 to 65535
Short int or Signed short int		8	-128 to 127
Unsigned short int		8	0 to 255
Long int or signed long int		32	-2147483648 to 2147483647
Unsigned long int		32	0 to 4294967295
Float	Float	32	3.4 e-38 to 3.4 e+38
Double	Double	64	1.7e-308 to 1.7e+308
Long Double		80	3.4 e-4932 to 3.4 e+4932
Void	Void		

ਡਾਟਾ ਕਿਸਮਾਂ ਚਾਰ ਟਾਈਪ ਦੀਆਂ ਹੁੰਦੀਆਂ ਹਨ ਜਿਵੇਂ signed, unsigned, short & long ਆਦਿ। ਇਹ ਚਿੰਨ੍ਹ ਟਾਈਪ ਡਾਟਾ ਅਤੇ ਇੰਟੀਜਰਸ (integers) ਦੀ ਸ਼ੁਧਤਾ ਪ੍ਰਦਾਨ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

ਨੋਟ : %*C specification ਦੀ ਵਰਤੋਂ newline (\n) ਨੂੰ ਪੜ੍ਹਨ ਲਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ, ਜਿਸਨੂੰ ਐਂਟਰ ਕੀ (enter key) ਦਬਾ ਕੇ ਹਾਸਿਲ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਐਂਟਰ ਕੀ (enter key) ਦੀ ਵਰਤੋਂ character ਨੂੰ ਐਂਟਰ (enter) ਕਰਨ ਲਈ ਨਵੀਂ ਰੇਖਾ ਉੱਪਰ ਜਾਣ ਲਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।

ਮਿਸਾਲ : ਕੋਡ GR 1055 ਵਿਚੋਂ GR ਨੂੰ ਹਾਸਿਲ ਕਰਨ ਲਈ ਅਸੀ %*C%*C ਦੀ ਵਰਤੋਂ ਦੋ ਵਾਰ, ਜਿਵੇਂ ਕਿ ਹੇਠਾਂ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਦਿਖਾਇਆ ਗਿਆ ਹੈ

```
/* Use of Assignment -suppression character
#include<stdio.h>
main()
{
int number;
puts( "Enter code number - first 2 chrs non-numeric");
scanf("%*c %*c%d",&number);
printf("%d\n",number);
puts("The first two digits are suppressed");
return(0);
}
```

Output:

Enter code number – first 2 chrs non-numeric
GR1055
1055
The first two digits are suppressed

ਚਿੰਨ੍ਹਾਤਮਕ ਕਾਂਨਸਟੈਂਟ ਸਮਰੂਪਾਂ ਦੇ ਮੁੱਲ ਹੁੰਦੇ ਹਨ ਜੋ ਪ੍ਰੀ-ਪ੍ਰੋਸੈਸਰ ਹਿਦਾਇਤ # define ਦੇ ਨਾਲ ਪਰਿਭਾਸ਼ਤ ਹੁੰਦੇ ਹਨ।

ਮਿਸਾਲ : # define Max 100

ਸੀ (C) ਔਪਰੇਟਰ ਬੁਨਿਆਦੀ ਤੌਰ ਤੇ ਉਨੇਰੀ *(unary)*, ਬਾਈਨਰੀ *(binary)* ਅਤੇ ਟਰਨਰੀ *(ternary)* ਉਪਰੇਸ਼ਨਾਂ ਵਿਚ ਸ਼੍ਰੇਣੀਬੱਧ ਕੀਤੇ ਜਾਂਦੇ ਹਨ। ਉਨੇਰੀ *(unary)* ਵਿਚ ਪੰਜ ਓਪਰੇਟਰ ਹੁੰਦੇ ਹਨ, ਜਿਸ ਵਿਚ ਚਾਰ ਦੋਹਰੈਪਨ ਅਤੇ ਇਕ ਟਰਨਰੀ *(ternary)* ਔਪਰੇਟਰ ਜਿਥੇ ਉਨੇਰੀ *(unary)* !,++, - ਅਤੇ – - ਬਾਈਨਰੀ *(binary)* ਔਪਰੇਟਰ ਹਨ। +,-,*/ ਬੁਨਿਆਦੀ ਅਰਥਮੇਟਿਕ ਔਪਰੇਟਰ ਹਨ। ਮਾਡਯੂਲ ਓਪਰੇਟਰ ਇਕ ਹੋਰ ਅਰਥਮੇਟਿਕ ਔਪਰੇਟਰ ਹੈ ਜਿਸਦਾ ਪਰਿਣਾਮ ਪੂਰਨ-ਅੰਕ ਵੰਡ ਦੇ ਬਾਅਦ ਰਿਮੇਂਡਰ *(remainder)* ਵਿਚ ਹੁੰਦਾ ਹੈ।<, <=, >, =>, = =, != ਰਿਲੇਸ਼ਨਲ-ਔਪਰੇਟਰ (relational operator) ਹਨ, ਇਹ ਕਾਂਨਸਟੈਂਟ ਅਤੇ ਵੇਰੀਏਬਲਸ ਦੇ ਦਰਮਿਆਨ ਰਿਲੇਸ਼ਨ (relation) ਨੂੰ ਪਰਿਭਾਸ਼ਤ ਕਰਦੇ ਹਨ।

ਕਿਉਂਕਿ C ਭਾਸ਼ਾ ਅਸਾਈਨਡ (assigned) ਸਟੈਟਮੈਂਟ ਦਾ ਅਨੁੰਸਰਨ (follows) ਕਰਦੀ ਹੈ ਇਸ ਲਈ ਔਪਰੇਟਰਾਂ ਦਾ order of precedence ਹੁੰਦਾ ਹੈ ਜੋ ਐਕਸਪ੍ਰੇਸ਼ਨ ਦਾ ਮੁਲਾਂਕਣ ਕਰਦੇ ਹਨ।

Statement with simple assignment operator	Statement with shorthand operator
a = a + 1	a += 1
a = a – 1	a -= 1
a = a * (n+1)	a *= (n+1)
a = a / (n+1)	a /= (n+1)
a = a % b	a %= b

ਅਸੀਂ ਇਸ ਅਧਿਆਇ ਵਿਚ ਦੇਖਿਆ ਸੀ ਕਿ ਇਕੋ ਸਥਿਤੀਆਂ ਦੀਆਂ ਤਿੰਨ ਕਿਸਮਾਂ ਹੁੰਦੀਆਂ ਹਨ ਜਿਹੜੀਆਂ ਕੰਟਰੋਲ ਦੇ ਫਲੋ ਵਿਚ ਤਬਦੀਲੀ ਦੀ ਮੰਗ ਕਰਦੇ ਹਨ । ਇਹ ਹਨ-

1. ਪ੍ਰੋਗਰਾਮ ਦੇ ਕੁਝ ਭਾਗਾਂ ਨੂੰ *ਲਾਗੂ-ਨਾ-ਕਰਨ (skip)* ਜਾਂ ਉਸੇ ਭਾਗ ਨੂੰ ਮੁੜ ਲਾਗੂ-ਕਰਨ ਲਈ ਪਿੱਛੇ ਮੁੜਨਾ।
2. ਦੋ ਜਾਂ ਵਧੇਰੇ ਭਾਗਾਂ ਵਿਚੋਂ ਇਕ ਭਾਗ ਲਾਗੂ ਕਰਨਾ ਜੋ ਕਿ ਕਿਸੇ ਕੰਡੀਸ਼ਨ ਦੇ ਸਿੱਟੇ ਤੇ ਨਿਰਭਰ ਕਰਦਾ ਹੈ।
3. ਦਿਤੇ ਗਏ ਸਮੇਂ ਲਈ ਜਾਂ ਤਾਂ ਇਕ ਭਾਗ ਨੂੰ ਦੁਹਰਾਉਣ ਲਈ ਲਾਗੂ ਕਰਨਾ ਜਾਂ ਜਦ ਤੱਕ ਕੁਝ ਸਥਿਤੀ ਦੀ ਪੂਰਤੀ ਨਹੀਂ ਹੁੰਦੀ।

ਇਨ੍ਹਾਂ ਸਥਿਤੀਆਂ ਲਈ (c) ਕੋਲ ਚਾਰ ਟਾਈਪ ਦੇ ਸਟੇਟਮੇਂਟ ਹੁੰਦੇ ਹਨ ਜਿਵੇਂ ਡਿਕਲੇਅਰ ਸਟੇਟਮੇਂਟ, ਇਨਪੁਟ-ਪਰਿਣਾਮ ਸਟੇਟਮੇਂਟ, ਗਣਿਤ, ਲੌਜਿਕਲ ਸਟੇਟਮੇਂਟ ਅਤੇ ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ ਆਦਿ।

ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ ਦੀ ਪਰਿਭਾਸ਼ਾ ਹੈ, ਪ੍ਰੋਗਰਾਮ ਸਟੇਟਮੇਂਟ ਦੇ ਲਾਗੂਕਰਨ ਦਾ ਕ੍ਰਮ ਪ੍ਰਦਾਨ ਕਰਦਾ ਹੈ। ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ ਤਿੰਨ ਟਾਈਪ ਦੀਆਂ ਹੁੰਦੀਆਂ ਹਨ : ਕੰਡੀਸ਼ਨਲ, ਲੂਪਿੰਗ ਅਤੇ ਅਨਕੰਡੀਸ਼ਨਲ ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ।
ਚਾਰ ਬੁਨਿਆਦੀ ਕੰਡੀਸ਼ਨਲ ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ ਹਨ if, if-else, if-else-if ਅਤੇ switch ਸਟੇਟਮੇਂਟ।

ਇਨ੍ਹਾਂ ਨੂੰ ਚੋਣਵੇਂ ਜਾਂ ਫੈਸਲਾ ਕਰਨ ਵਾਲੇ ਕੰਟਰੋਲ ਸਟੇਟਮੇਂਟ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। "goto conditional control statement" ਹੈ ਜਿਹੜੀ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਨਿਸ਼ਚਿਤ ਸਟੇਟਮੇਂਟ ਦਾ ਕੰਟਰੋਲ ਬਦਲਦੀ ਹੈ।

ਮਿਸਾਲ : // Program find whether a number is negative or positive */

```
#include<stdio.h>
void main()
{
int num;
printf("Enter the number ");
scanf("%d",&num);
if(num<0)
printf("The number is negative");
else
printf("The number is positive"); }
```

Output:

Enter the number 5
The number is positive

Enter the number - 90
The number is negative

ਉੱਪਰ ਲਿਖੇ ਪ੍ਰੋਗਰਾਮ ਵਿਚ if ਸਟੇਟਮੇਂਟ ਇਹ ਚੈੱਕ ਕਰਦੀ ਹੈ ਕਿ ਦਿੱਤਾ ਗਿਆ ਨੰਬਰ 0 ਤੋਂ ਘੱਟ ਹੈ ਕਿ ਨਹੀਂ। ਜੇਕਰ ਇਹ ਨੰਬਰ 0 ਤੋਂ ਘੱਟ ਹੈ ਤਾਂ ਇਹ ਨੰਬਰ ਨੈਗੇਟਿਵ (negative) ਹੈ, ਅਤੇ ਕੰਡੀਸ਼ਨ (condition) ਸਹੀ ਹੋਵੇਗੀ ਅਤੇ ਇਹ "the number is negative" ਸਟੇਟਮੇਂਟ ਚਲਾਏਗੀ। ਜੇਕਰ ਨੰਬਰ 0 ਤੋਂ ਘੱਟ ਨਹੀਂ ਹੈ ਤਾਂ if else ਸਟੇਟਮੇਂਟ ਦਾ ਭਾਗ ਪਹਿਲੀ ਸਟੇਟਮੇਂਟ ਨੂੰ ਲਾਗੂ ਨਹੀਂ ਕਰੇਗਾ ਅਤੇ ਦੂਜੀ ਸਟੇਟਮੇਂਟ ਨੂੰ ਜਿਸ ਵਿਚ "number is positive" ਹੈ ਨੂੰ ਲਾਗੂ ਕਰੇਗਾ।

ਲੂਪਿੰਗ ਨੂੰ ਇਟੀਰੇਟਿਵ ਜਾਂ ਰਿਪੀਟਿਵ (iterative or repetive) ਕੰਟਰੋਲ ਡਿਵਾਈਸ ਸਟਰਕਚਰ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਜਿਥੇ int, ਸਟੇਟਮੇਂਟਸ ਦਾ ਸੈੱਟ ਬਾਰ ਬਾਰ ਲਾਗੂ ਹੁੰਦਾ ਹੈ ਚਾਹੇ ਨਿਸ਼ਚਿਤ ਸਮੇਂ ਲਈ ਜਾ ਲੌਜਿਕਲ ਕੰਡੀਸ਼ਨ ਲਈ ਸਹੀ ਹੋਵੇ।

While, do-while ਅਤੇ for loop ਸਟੇਟਮੈਂਟ ਤਿੰਨ ਸ਼ਕਤੀਸ਼ਾਲੀ ਲੂਪ ਹਨ ਜੋ C ਵਿਚ ਉਪਲੱਬਧ ਸਟਰਕਚਰ ਤੇ ਕੰਟਰੋਲ ਕਰਦੇ ਹਨ। while ਸਟੇਟਮੈਂਟ ਪ੍ਰੀ-ਟੈਸਟ ਸਟੇਟਮੈਂਟ ਹੈ (ਇਹ ਤਾਂ ਹੀ ਲਾਗੂ ਹੁੰਦੀ ਹੈ ਜਦੋਂ ਐਕਸਪ੍ਰੇਸ਼ਨ ਸਹੀ ਹੋਵੇ) ਅਤੇ do-while ਇਕ ਪੋਸਟ-ਟੈਸਟ ਲੂਪ ਹੈ (ਇਹ ਇਕੋ ਵਾਰ ਲਾਗੂ ਹੁੰਦੀ ਹੈ ਫਿਰ ਏਕਸਪ੍ਰੇਸ਼ਨ-ਲੂਪ ਲਈ ਦੁਹਰਾਏ ਜਾਣ ਤੇ ਮੁਲਾਂਕਣ ਕੀਤਾ ਜਾਂਦਾ ਹੈ)। for ਲੂਪ ਦੀ ਵਰਤੋਂ ਉਦੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਜਦੋਂ ਯੂਜ਼ਰ ਜਾਣਦਾ ਹੋਵੇ ਕਿ ਇਟੀਰੇਸ਼ਨ ਕਿੰਨੀ ਵਾਰ ਲਾਗੂ ਹੋਣੇ ਹਨ। ਕਈ ਵਾਰ ਲੂਪ ਤੋਂ ਬਾਹਰ ਆਉਣ ਦੀ ਜਾਂ ਇਟੀਰੇਟਿਵ ਵਿਚ ਲੂਪ ਦੇ ਕਿਸੇ ਭਾਗ ਨੂੰ skip ਕਰਨ ਦੀ ਜ਼ਰੂਰਤ ਪੈਂਦੀ ਹੈ। ਅਜਿਹਾ ਕਰਨ ਲਈ break ਅਤੇ continue ਸਟੇਟਮੈਂਟ ਵਰਤੇ ਜਾਂਦੇ ਹਨ।

ਮਿਸਾਲ : **//program to illustrate the use of while loop with variables**

```
#include<stdio.h>
#define N 100
#define A 2
main( )
{
 int a;
 a = A;
 while (a<N)
 {
  printf("%d \n",a=a*a);
 }
  return(0);
}
```

Output

```
4
16
256
```

ਐਰੇ ਇਕ ਡਾਟਾ ਸਟਰਕਚਰ ਹੁੰਦੀ ਹੈ ਜਿਹੜੀ ਇਕ ਦਿਤੇ ਗਏ ਨਾਮ ਤੇ ਅਜਿਹੇ ਮੁੱਲਾਂ ਦੇ ਸੰਗ੍ਰਹਿ ਦਾ ਸਟੋਰ ਕਰਦੀ ਹੈ, ਇਨ੍ਹਾਂ ਨੂੰ ਸਬ-ਸਕਰਿਪਟ (subscript) ਵੇਰਿਏਬਲ ਵੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ।

ਐਰੇ ਦੇ ਅੰਸ਼ ਮੈਮੋਰੀ ਸਥਿਤੀਆਂ ਵਿਚ ਸਟੋਰ ਕੀਤੇ ਜਾਂਦੇ ਹਨ। ਇਨ੍ਹਾਂ ਅੰਸ਼ਾਂ ਵਿਚ ਐਰੇ ਵਿਚ ਉਨ੍ਹਾਂ ਦੇ ਨਾਵਾਂ ਨਾਲ ਪਹੁੰਚ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਜੋ ਕਿ ਬਰੈਕਟਾਂ ਵਿਚ ਸਬ-ਸਕਰਿਪਟ (subscript) ਅੰਕ ਦੁਆਰਾ ਅਪਣਾਏ ਜਾਂਦੇ ਹਨ।

ਵੇਰੀਏਬਲ (variable) ਦੀ ਤਰ੍ਹਾਂ ਐਰੇ (array) ਨੂੰ ਇਸਦੀ ਵਰਤੋਂ ਤੋਂ ਪਹਿਲਾਂ ਡਿਕਲੇਅਰ ਕੀਤਾ ਜਾਣਾ ਚਾਹੀਦਾ ਹੈ।

ਡਾਟਾ_ਟਾਈਪ ਏਰੇ_ਨੇਮ [ਐਲੀਮੈਂਟਸ ਦੇ ਨੰਬਰ ਐਰੇਜ਼ ਵਿਚ]
Data_type array_name [number of elements in array]

ਜਿਥੇ ਡਾਟਾ ਟਾਈਪ (data type), ਇਕ ਵੈਲਿਡ-ਟਾਈਪ (valid type) ਹੈ (ਜਿਵੇਂ int, float...), ਨੇਮ ਵੈਲਿਡ-ਆਈਡੇਂਟੀਫਾਈਰ (valid identifier) ਹੈ ਅਤੇ ਐਲੀਮੈਂਟ (element) ਖੇਤਰ (field) ਹੈ (ਜੋ ਹਮੇਸ਼ਾ ਚੌਰਸ ਬਰੈਕਟਾਂ [ ] ਵਿਚ ਹੁੰਦੇ ਹਨ) ਨਿਸ਼ਚਿਤ ਕਰਦੀ ਹੈ ਕਿ ਐਰੇ (array) ਵਿਚ ਇਨ੍ਹਾਂ ਐਲੀਮੈਂਟਸ (elements) ਚੋਂ ਕਿੰਨੇ ਸ਼ਾਮਲ ਕੀਤੇ ਜਾਣੇ ਹਨ।

ਐਰੇ ਵਿਚ ਡਾਟਾ ਭਰਨਾ (Entering Data into an Array)

ਕਿਉ ਜੋ ਐਰੇ (array) ਮੁਲਾਂ (values) ਦਾ ਸੰਗ੍ਰਹਿ ਕਰਦੀ ਹੈ, ਇਸ ਲਈ ਐਰੇ (array) ਦੇ ਵਿਅਕਤੀਗਤ ਐਲੀਮੈਂਟਸ (elements) ਵਿਚ ਡਾਟਾ ਇਨਪੁਟ (input) ਕਰਨ ਲਈ ਅਸੀਂ ਲੂਪ ਸਟੇਟਮੈਂਟਸ (loop statements) ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹਾਂ। ਜੇਕਰ ਐਰੇ (array) ਦੇ ਅਸਲ ਆਕਾਰ ਦਾ ਪਹਿਲਾਂ ਹੀ ਪਤਾ ਲੱਗ ਜਾਵੇ ਤਾਂ while loop ਜਾਂ do-while-loop ਦੀ ਜਗ੍ਹਾ for loop ਨੂੰ ਪਹਿਲ ਦਿਤੀ ਜਾਂਦੀ ਹੈ।

ਐਰੇ ਦੇ ਮੁੱਲਾਂ ਦੀ ਪਹੁੰਚ-ਕਰਨੀ (Accessing the values of an array)

ਇਹ ਪ੍ਰੋਗਰਾਮ ਦੇ ਕਿਸੇ ਵੀ ਬਿੰਦੂ ਤੇ ਜਿਥੇ ਐਰੇ ਪ੍ਰਤੱਖ (visible) ਹੁੰਦਾ ਹੈ ਅਸੀਂ ਇਨ੍ਹਾਂ ਦੇ ਵਿਅਕਤੀਗਤ ਐਲੀਮੈਂਟਸ (elements) ਵਿਚੋਂ ਕਿਸੇ ਦੇ ਵੀ ਮੁੱਲ ਤਕ ਪਹੁੰਚ (access) ਕਰ ਸਕਦੇ ਹਾਂ ਜਿਵੇਂ ਕਿ ਇਹ ਆਮ ਵੈਰੀਏਬਲ (variable) ਹੁੰਦਾ ਹੈ। ਇਸ ਪ੍ਰਕਾਰ ਪੜ੍ਹਣ ਅਤੇ ਇਸਦੇ ਮੁੱਲ ਨੂੰ ਸੋਧਣ ਦੇ ਕਾਬਿਲ ਹੋ ਜਾਂਦੇ ਹਾਂ। ਇਸਦੀ ਬਣਤਰ ਬਹੁਤ ਸਧਾਰਣ ਹੁੰਦੀ ਹੈ।

name[index];

ਐਰੇ ਦੀਆਂ ਦੋ ਕਿਸਮਾਂ ਹੁੰਦੀਆਂ ਹਨ – 1-Dimentional ਐਰੇ ਅਤੇ ਦੂਸਰੀ Multi-Dimentional ਐਰੇਜ਼।

ਇਕ square ਬਰੈਕਟ ਦੇ ਜੋੜੇ ਨੂੰ 1-Dimentional ਐਰੇ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ 1-Dimentional ਐਰੇ ਦਾ ਕੇਵਲ ਇਕ ਸਬ-ਸਕਰਿਪਟ (subscript) ਹੁੰਦਾ ਹੈ ਅਤੇ ਇਹ ਮੈਟਰਿਕਸ ਜਾਂ ਕਾਲਮ ਮੈਟਰਿਕਸ ਵਜੋਂ ਦੇਖਿਆ ਜਾਂਦਾ ਹੈ।

Multi-Dimentional ਐਰੇਜ਼ ਦੇ ਇਕ ਤੋਂ ਵੱਧ ਸਬ-ਸਕਰਿਪਟ (subscript) ਹੁੰਦੇ ਹਨ। C ਵੱਲੋਂ ਸਟਰਿੰਗ ਡਾਟਾ ਦੀ ਸਹਾਇਤਾ ਨਹੀਂ ਕੀਤੀ ਜਾਂਦੀ। ਪਰੰਤੂ ਸਟਰਿੰਗਸ ਚਿੰਨ੍ਹਾਂ ਵੀ ਐਰੇ ਵੱਜੋਂ ਇਸਤੇਮਾਲ ਕੀਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ।
ਸਟਰਿੰਗਸ ਦੀ ਜਰੂਰਤਾਂ ਲਈ ਫੰਕਸ਼ਨ ਦੀ ਭਰਪੂਰ ਲਾਇਬ੍ਰੇਰੀ ਹੁੰਦੀ ਹੈ।

ਦੋ-ਡਾਇਮੈਂਸ਼ਨਲ ਐਰੇ ਦੀ ਡਿਕਲੇਅਰੇਸ਼ਨ (declaration of Two dimensional array)

C ਭਾਸ਼ਾ ਵਿਚ ਦੋ ਡਾਇਮੈਂਸ਼ਨਲ (dimensional) ਐਰੇ ਨੂੰ ਪ੍ਰਭਾਸ਼ਿਤ ਕਰਨ ਲਈ ਵਾਕ ਰਚਨਾ :

data type array _ name [row] [columns];

ਜਿਥੇ ਰੋ (row) -- > ਐਲੀਮੈਂਟਸ (elements) ਦੀ ਸੰਖਿਆ ਜੋ ਸਬਸਕ੍ਰਿਪਟ (subscript) 1 ਦੇ ਵਿਚ ਪ੍ਰੋਸੈਸ ਹੁੰਦੇ ਹਨ।

ਕਾਲਮ (column) --> ਐਲੀਮੈਂਟਸ (elements) ਦੀ ਸੰਖਿਆ ਜੋ ਸਬਸਕ੍ਰਿਪਟ (subscript) 2 ਦੇ ਵਿਚ ਪ੍ਰੋਸੈਸ ਹੋਣੇ ਹਨ।

ਮਿਸਾਲ: Example program to sum elements separately with odd & even subscript

```
#include<stdio.h>
main( )
{
int x[5][5] = {{ 15,6,7,28,12} , {50,5,4,3,65}, {34, 32, 21, 29, 31}, {20, 49, 18, 7, 35}, {6, 4, 19, 7, 6}};
int sum1 = 0, sum2 =0,m,n;
for (m =0;m<5;++m)
{
for(n =0;n<5;++n)
{
if(m % 2 == 0)
sum1=sum1+ x[m][n];
else
sum2 = sum2 + x[m][n];
}
}
for (m =0;m<5;++m)
{
for(n =0;n<5;++n)
printf("%5d" , x[m][n]);
pritnf("\n") ;
}
printf("Total of elements with even first subscript %5d\n" ,sum1);
printf("Total of elements with odd first subscript %5d\n" ,sum2);
}
```

Sample output

15	6	7	28	12
50	5	4	3	65
34	32	21	29	31
20	49	18	7	35
6	4	19	7	6

Total of elements with even first subscript 257

Total of elements with odd first subscript 256

ਮਿਸਾਲ : // To print the number 5 five times, number 4 times, number 3 times & so on.

```
#include<stdio.h>
void main()
{
 int row,a,b;
printf("\n Enter the number of rows:");
scanf("%d",&row);
for(a=row;a>=1;a--)
{
 for(b=1;b<=a;b++)
  {
    printf(" %d",a);
  }
    printf("\n");
}
}
```

Output

```
5    5    5    5    5
4    4    4    4
3    3    3
2    2
1
```

ਮਿਸਾਲ : // Remove the errors in following program to read 5 words and print them

```
#include<stdio.h>
 main()
{
char words[5][50];
unsigned m;
puts("Enter 5 words");
for (m=0;m<5;++m)
scanf("%s",words[m]);
for (m=0;m<5;++m)
printf("%s\n",words[m]);
return(0);
}
```

Sample Output

Input : Data Types Used In ChapterI

Output :

Data
Types
Used
In
ChapterI

ਮਿਸਾਲ : /* Program exchanges the value of two variables*/

```
#include <stdio.h>
main()
{
 int x,y,dummy;
printf("Enter the value of x and y \n");
scanf("%d %d",&x ,&y);
printf("x = %d and y= %d\n",x,y);
printf("Using a dummy variables \n");
dummy = x;
x= y;
y= dummy;
printf("x = %d and y = %d\n",x,y);
printf("Using a without dummy variables \n");
x = x+y;
y = x-y;
x = x-y;
printf("x = %d and y = %d\n",x,y);
return(0);
}
```

Output

```
Enter the value of x and y
30
40
Using a dummy variables
 x =40 and y =30
Using a without dummy variables
 x =30 and y =40
```

ਡੈਸਕਟਾਪ ਪਬਲਿਸ਼ਿੰਗ

1. ਡੈਸਕਟਾਪ ਪਬਲਿਸ਼ਿੰਗ ਸਾਫਟਵੇਅਰ ਰਿਪੋਰਟਾਂ, ਵਿਜਿਟਿੰਗ ਕਾਰਡ, ਕੈਲੰਡਰ, ਇਸ਼ਤਿਹਾਰ, ਮੈਗਜ਼ੀਨ ਆਦਿ ਬਣਾਉਣ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ।

2. ਕੋਈ ਵੀ ਡਾਕੂਮੈਂਟ ਤਿਆਰ ਕਰਨ ਤੋਂ ਪਹਿਲਾਂ ਸਾਡਾ ਪਹਿਲਾ ਕੰਮ ਇਹ ਹੈ ਕਿ ਇਹ ਪਤਾ ਕਰ ਲਿਆ ਜਾਵੇ ਕਿ ਕਿਸ ਤਰ੍ਹਾਂ ਦਾ ਡਾਕੂਮੈਂਟ ਤਿਆਰ ਹੋਣ ਜਾ ਰਿਹਾ ਹੈ।

3. ਪ੍ਰਿੰਟਿੰਗ ਦਾ ਕੰਮ ਟੈਕਸਟ, ਤਸਵੀਰ ਜਾਂ ਗ੍ਰਾਫਿਕ ਨੂੰ ਬਾਰ-ਬਾਰ ਛਾਪਣਾ ਹੈ।

4. ਲੇਜ਼ਰ ਪ੍ਰਿੰਟਿੰਗ ਅਤੇ ਆਫਸੈਟ ਪ੍ਰਿੰਟਿੰਗ ਡਾਕੂਮੈਂਟ ਪ੍ਰਿੰਟ ਕਰਨ ਦੇ ਆਮ ਤਰੀਕੇ ਹਨ।

5. ਫੌਂਟ Typeface ਅਤੇ ਹੋਰ ਵਿਸ਼ੇਸ਼ਤਾਵਾਂ ਦਾ ਇੱਕਠਾ ਹੈ ਜਿਵੇਂ ਕਿ ਇਸਦਾ ਅਕਾਰ, ਪੁਆਇੰਟ (Points) ਅਤੇ ਚੌੜਾਈ (Width) ਆਦਿ।

6. ਲੀਡਿੰਗ (Leading) ਦਾ ਮਤਲਬ ਹੈ ਦੋ ਜਾਂ ਜ਼ਿਆਦਾ ਲਾਈਨਾਂ ਵਿਚਕਾਰ ਦੂਰੀ।

7. ਪੇਜ ਲੇਆਉਟ (Page Layout) ਦਾ ਕੰਮ ਹੈ, ਓਬਜੈਕਟਸ ਨੂੰ ਲੜੀਬੱਧ ਤਰੀਕੇ ਨਾਲ ਰੱਖਣਾ।

8. WYSIWYG ਦਾ ਮਤਲਬ ਹੈ What You see Is What You Get.

9. ਜੇ ਤੁਹਾਨੂੰ ਡਾਕੂਮੈਂਟ ਵਿਚ ਜ਼ਿਆਦਾ Layout control ਦੀ ਲੋੜ ਹੈ ਤਾਂ ਤੁਹਾਨੂੰ ਡੈਸਕਟਾਪ ਪਬਲਿਸ਼ਿੰਗ ਸਾਫਟਵੇਅਰ ਦੀ ਲੋੜ ਪਵੇਗੀ। ਉਦਾਹਰਨ ਦੇ ਤੌਰ ਤੇ ਜੇ ਤੁਸੀ ਬਰਾਊਚਰ, ਅਖਬਾਰ, ਮੈਗਜ਼ੀਨ ਜਾਂ ਕੋਈ ਬਿਜਨਸ ਕਾਰਡ ਆਦਿ ਤਿਆਰ ਕਰਨੇ ਹੋਣ।

10. ਮਾਰਜਨ ਟੈਕਸਟ ਦੀ ਸੱਜੇ ਅਤੇ ਖੱਬੇ, ਉੱਪਰ ਤੇ ਹੇਠਾਂ ਤੋਂ ਦੂਰੀ ਨੂੰ ਕਿਹਾ ਜਾਂਦਾ ਹੈ।

ਅਧਿਆਇ

2

ਸਟਰਿੰਗ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ

String handling functions

ਭੂਮਿਕਾ (Introduction)

ਇਕ ਚਿੰਨ੍ਹ (character) ਇਕਹਿਰਾ ਅੱਖਰ (letter), ਨਿਊਮਰਲ (numeral), ਵਿਰਾਮ ਚਿੰਨ੍ਹ (punctuation mark) ਜਾਂ ਹੋਰ ਕੋਈ ਅਜਿਹਾ ਚਿੰਨ੍ਹ (symbol) ਹੁੰਦਾ ਹੈ। ਸਟਰਿੰਗ (string) ਚਿੰਨ੍ਹਾਂ ਦਾ ਇਕ ਸਾਰ ਹੁੰਦਾ ਹੈ। ਦੋਹਰੇ ਕੁਟੇਸ਼ਨ ਚਿੰਨ੍ਹ ਦੇ ਨਾਲ ਪਰਿਭਾਸ਼ਤ ਕੀਤੇ ਗਏ ਚਿੰਨ੍ਹਾਂ ਦਾ ਸੈੱਟ ਜਾਂ ਕੋਈ ਵੀ ਸਾਰ ਇਕ ਸਥਾਈ ਜਾਂ ਨਿਰੰਤਰ ਚਿੰਨ੍ਹ ਹੁੰਦਾ ਹੈ। ਸਟਰਿੰਗ (string) ਟੈਕਸਟ-ਡਾਟਾ (text data) ਨੂੰ ਕਾਬੂ ਕਰਨ ਲਈ ਵਰਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ਜਿਸ ਵਿਚ ਅੱਖਰ (letter), ਨਿਊਮੈਰਿਕ (numeric), ਵਿਰਾਮ ਚਿੰਨ੍ਹ (punctuation mark) ਅਤੇ ਹੋਰ ਚਿੰਨ੍ਹ (symbol) ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ। ਹੇਠ ਦਿਤੀ ਗਈ ਮਿਸਾਲ ਵਿਚ **'Computer'** ਅੱਖਰ, 8 ਚਿੰਨ੍ਹਾਂ ਦੀ ਸਟਰਿੰਗ (string) ਹੈ ਅਤੇ ਰੈਮ (RAM) ਵਿਚ ਸਟੋਰ ਹੁੰਦਾ ਹੈ। ਜਿੱਥੇ **ch** ਕਰੈਕਟਰ ਦੇ ਐਰੇਜ਼ (arrays of character) ਨੂੰ ਦਰਸਾਉਂਦੀ ਹੈ।

c	o	m	p	u	t	e	r
ch[0]	ch[1]	ch[2]	ch[3]	ch[4]	ch[5]	ch[6]	ch[7]

ਸੀ (C) ਇਕ String Handling Function ਪ੍ਰਦਾਨ ਕਰਦੀ ਹੈ ਜਿਹੜੇ ਹੈਡਰ ਫਾਈਲ (header file) <string.h> ਵਿਚ ਪਰਿਭਾਸ਼ਤ ਕੀਤੇ ਗਏ ਹਨ। ਇਕ ਸਟਰਿੰਗ (string) ਦੇ ਵਿਅਕਤੀਗਤ ਚਿੰਨ੍ਹਾਂ (characters) ਜਾਂ ਸਟਰਿੰਗਸ ਦੇ ਸਮੂਹ (group of strings) ਤੇ ਹੇਠ ਲਿਖੇ ਕੰਮ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ :

- ਸਟਰਿੰਗ ਪੜ੍ਹਨਾ ਸਟਰਿੰਗ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਨਾ — Reading string displaying strings
- ਮਿਲਾਉਣਾ ਜਾਂ ਸਟਰਿੰਗ ਨੂੰ ਲੜੀਬੱਧ ਕਰਨਾ — Combining or concatenating strings
- ਇਕ ਸਟਰਿੰਗ ਨੂੰ ਦੂਸਰੇ ਤੇ ਕਾਪੀ ਕਰਨਾ — Copying one string to another
- ਸਟਰਿੰਗ ਦੀ ਤੁਲਨਾ ਕਰਨਾ ਅਤੇ ਜਾਂਚ ਕਰਨੀ ਕਿ ਉਹ ਬਰਾਬਰ ਹਨ। — Comparing string & checking whether they are equal
- ਸਟਰਿੰਗ ਦੇ ਇਕ ਭਾਗ ਨੂੰ ਕੱਢਣਾ — Extraction of a portion of a string

ਸਟਰਿੰਗ (string) ਮੈਮੋਰੀ ਵਿਚ ਸਟੋਰ ਕੀਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ਕਿਉਂ ਜੋ ASCII ਕੋਡ ਜਿਹੜੇ ਸਟਰਿੰਗ (string) ਨੂੰ '\O' (ASCII value of null) ਦੇ ਨਾਲ ਜੋੜਦੇ ਹਨ। ਆਮ ਤੌਰ ਤੇ ਇਕ ਚਿੰਨ੍ਹ (character) ਇਕ ਬਾਈਟ (byte) ਵਿਚ ਸਟੋਰ ਕੀਤਾ ਜਾਂਦਾ ਹੈ; ਕ੍ਰਮਵਾਰ ਚਿੰਨ੍ਹ (character) ਕ੍ਰਮਵਾਰ ਬਾਈਟਾਂ (bytes) ਵਿਚ ਸਟੋਰ ਕੀਤੇ ਜਾਂਦੇ ਹਨ।

Character	m	y		a	g	e		i	s
ASCII code	77	121	32	97	103	10	32	105	115

Character		2		(	t	w	o	)	\0
ASCII code	32	50	32	40	116	119	41	0	0

ਅੰਤਿਮ ਚਿੰਨ੍ਹ (character) ਖਾਲੀ ਚਿੰਨ੍ਹ (Null character) ਹੁੰਦਾ ਹੈ ਜਿਸਦਾ ASCII ਮੁੱਲ ਜ਼ੀਰੋ (0) ਹੁੰਦਾ ਹੈ। ਕੰਪਿਊਟਰ ਦੀ ਮੈਮੋਰੀ ਸਾਰੇ ਡਾਟਾ ਨੂੰ ਇਕ ਨਿਊਮੈਰਿਕ ਰੂਪ (numeric form) ਵਿਚ ਸਟੋਰ ਕਰਦੀ ਹੈ। ਚਿਨ੍ਹਾਂ (characters) ਨੂੰ ਸਟੋਰ ਕਰਨ ਦਾ ਹੋਰ ਕੋਈ ਸਿੱਧਾ ਰਾਹ ਨਹੀਂ ਹੁੰਦਾ। ਭਾਵੇਂ ਕਿ ਇਕ ਨਿਊਮੈਰਿਕ ਕੋਡ (numeric code) ਹਰੇਕ ਚਿੰਨ੍ਹ (character) ਲਈ ਮੌਜੂਦ ਹੁੰਦਾ ਹੈ। ਇਸ ਨੂੰ ASCII ਕੋਡ ਜਾਂ ASCII ਚਿੰਨ੍ਹ-ਸੈੱਟ (character set) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। (ASCII ਸੂਚਨਾ-ਅੰਤਰ-ਬਦਲ (information interchange) ਲਈ ਅਮਰੀਕਨ ਸਟੈਂਡਰਡ ਕੋਡ ਹੁੰਦਾ ਹੈ।) ਅੱਪਰ ਅਤੇ ਲੋਅਰ ਕੇਸ (case), ਅੱਖਰਾਂ (letters), ਨਿਊਮੈਰਿਕ ਡਿਜਿਟਸ

(numeric digits), ਵਿਰਾਮ ਚਿੰਨ੍ਹਾਂ (punctuation marks) ਅਤੇ ਹੋਰ ਚਿੰਨ੍ਹਾਂ (symbols) ਲਈ ਕੋਡ 0 ਅਤੇ 255 ਦੇ ਦਰਮਿਆਨ ਮੁੱਲ-ਅਸਾਈਨ (value assigns) ਕਰਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ 97 ਜੋ **a** ਲਈ ASCII ਕੋਡ ਹੈ। ਜਦੋਂ ਤੁਸੀਂ ਚਿੰਨ੍ਹ (character) **a** ਨੂੰ ਇਕ ਕੈਰ-ਕਿਸਮ (char type) ਵੇਰੀਏਬਲ (variable) ਵਿਚ ਸਟੋਰ ਕਰਦੇ ਹੋ, ਤੁਸੀਂ ਅਸਲ ਵਿਚ 97 ਮੁੱਲ ਨੂੰ ਸਟੋਰ ਕਰ ਰਹੇ ਹੁੰਦੇ ਹੋ। ਕਿਉਂਜੋ ਕੈਰ-ਕਿਸਮ (char type) ਲਈ ਆਗਿਆਯੋਗ ਨਿਊਮੈਰਿਕ ਰੇਂਜ (numeric range) ਸਟੈਂਡਰਡ ASCII ਚਿੰਨ੍ਹ ਸੈੱਟ (character set) ਨਾਲ ਮੇਲ ਖਾਂਦੀ ਹੈ, ਕੈਰ (char) ਚਿੰਨ੍ਹ (character) ਸਟੋਰ ਕਰਨ ਲਈ ਢੁਕਵਾਂ ਹੁੰਦਾ ਹੈ।

ਡਿਕਲੇਅਰੇਸ਼ਨ ਅਤੇ ਇਨੀਸ਼ਿਯਲਾਈਜ਼ਿੰਗ ਸਟਰਿੰਗ ਵੇਰੀਏਬਲਸ (Declaration & Initializing String Variables)

ਸੀ (C) ਵਿਚ ਅਸੀਂ ਸਟਰਿੰਗ ਵੇਰੀਏਬਲ (string variable) ਲਈ ਸਟਰਿੰਗ (string) ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹਾਂ। ਆਮ ਰੂਪ ਹੈ

```
data-type      string-name [size];
```

ਡਾਟਾ-ਕਿਸਮ ਸਟਰਿੰਗ (string) ਦੀ ਇਕ ਕਿਸਮ ਹੈ ਜੋ ਆਮ ਤੌਰ ਤੇ ਕੈਰ (char) ਹੁੰਦੀ ਹੈ। ਸਾਈਜ (size) ਫੈਸਲਾ ਕਰਦਾ ਹੈ ਕਿ ਸਟਰਿੰਗ (string) ਵਿਚ ਕਿੰਨੇ ਚਿੰਨ੍ਹ (character) ਹੋਣਗੇ । ਅਸੀਂ ਜਦੋਂ ਚਿੰਨ੍ਹ (char) ਸਟਰਿੰਗ (string) ਦਾ ਕੰਪਾਈਲ (compile) ਕਰਦੇ ਹਨ ਤਾਂ ਕੰਪਾਈਲਰ (compiler) ਆਪਣੇ ਆਪ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਵਿਚ EOS (end of string) ਜਾਂ ਖਾਲੀ ਚਿੰਨ੍ਹ (null character) ('\O') ਸਪਲਾਈ ਕਰਦੇ ਹਨ। ਇਸ ਲਈ ਸਟਰਿੰਗ (string) ਦਾ ਸਾਈਜ਼ (size), ਸਟਰਿੰਗ (string) ਵਿਚ ਚਿੰਨ੍ਹਾਂ (char) ਦੀ ਵੱਧ ਤੋਂ ਵੱਧ ਸੰਖਿਆ ਵਿਚ ਇਕ ਅੰਕ ਜਮਾਂ ਕਰਕੇ ਹੁੰਦਾ ਹੈ।

ਨਿਊਮੈਰਿਕ-ਐਰੇ (numeric array) ਦੀ ਤਰ੍ਹਾਂ ਕੈਰੇਕਟਰ ਐਰੇ (character array) ਇਕ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਦੇਖਣ ਤੋਂ ਪਹਿਲਾਂ ਹੀ ਡਿਕਲੇਅਰ (declare) ਕੀਤੀ ਜਾਣੀ ਚਾਹੀਦੀ ਹੈ। ਡਾਟਾ ਕੈਰ (char) ਟਾਈਪ ਵਿਚ ਤਬਦੀਲੀ ਤੋਂ ਇਲਾਵਾ ਵਾਕ-ਰਚਨਾ (syntax) ਵਿਚ ਕੋਈ ਤਬਦੀਲੀ ਨਹੀਂ ਹੁੰਦੀ ਅਤੇ ਕੈਰੇਕਟਰ ਐਰੇ (character array) ਦੇ ਹਰ ਇਕ ਐਲੀਮੈਂਟ (element) ਦੀ ਇਨੀਸ਼ਿਯਲਾਈਜੇਸ਼ਨ (initialization) ਕੈਰ ਕਾਂਸਟੈਂਟ (char constant) ਨਾਲ ਹੁੰਦੀ ਹੈ ਜੋ ਕਿ ਅਸਾਈਨਿੰਗ ਕਰੈਕਟਰ (assigning character) ਨੂੰ ਸਥਾਈ (ਇਕਹਿਰੇ ਕੋਟ ' ' ਨਾਲ ਨੱਥੀ) ਇਕ ਕਰੈਕਟਰ (character) ਦੁਆਰਾ ਬਣਾਇਆ ਗਿਆ ਹੁੰਦਾ ਹੈ।

ਮਿਸਾਲ ਲਈ : ਅਸੀਂ ਇਕ ਕੈਰੇਕਟਰ ਐਰੇ (character array) ਦਾ ਆਰੰਭ (initialize) ਕਰਦੇ ਹਾਂ ਜਦੋਂ ਉਨ੍ਹਾਂ ਨੂੰ ਡਿਕਲੇਅਰ (declare) ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਉਹ ਦੋ ਰੂਪਾਂ ਵਿਚ ਆਰੰਭ (initialize) ਕੀਤੇ ਜਾਂਦੇ ਹਨ :

```
char name [10] = {'R', 'a', 'm', 'a', '\0'};
```

ਇਥੇ ਕੈਰ name[10] ਦਾ ਇਕ ਸਟਰਿੰਗ (string) ਹੈ ਅਤੇ ਆਪਣੇ ਪਹਿਲੇ ਚਾਰ ਐਲੀਮੈਂਟ (element) ਜੋ ਕਿ ਕ੍ਰਮਵਾਰ R, a, m, a ਹੁੰਦੇ ਹਨ ਨੂੰ ਆਰੰਭ (initialize) ਕਰਦਾ ਹੈ ਅਤੇ ਪੰਜਵੇਂ ਐਲੀਮੈਂਟ (element) ਨੂੰ ਖਾਲੀ ਚਿੰਨ੍ਹ (character) ਦੇ ਨਾਲ ਅਸਾਈਨ (assign) ਕੀਤਾ ਗਿਆ ਹੈ ਜਿਸਦੇ ਨਾਲ ਸਟਰਿੰਗ (string) ਖਤਮ ਹੁੰਦੀ ਹੈ।

ਇਕ ਹੋਰ ਮਿਸਾਲ ਤੇ ਵਿਚਾਰ ਕਰੋ, ਜਿੱਥੇ ਚਿੰਨ੍ਹ ਵਿਅਕਤੀਗਤ ਤੌਰ ਤੇ ਆਰੰਭ ਨਹੀਂ ਕੀਤੇ ਜਾਂਦੇ।

```
char days [10] = {"Monday"}
```

ਇਥੇ ਸਟਰਿੰਗ **"Monday"** ਦਾ ਹਰ ਇਕ ਕੈਰੇਕਟਰ ਐਰੇ (character array) **days** ਐਰੇ ਦੇ ਵਿਅਕਤੀਗਤ ਐਲੀਮੈਂਟ (element) ਵਿਚ ਸਟੋਰ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਪਰੰਤੂ ਇਹ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਤੇ null character ਸ਼ਾਮਲ ਕਰਦਾ ਹੈ।

ਅਸੀਂ ਇਕ ਐਰੇ (array) ਨੂੰ ਆਕਾਰ ਨਿਸ਼ਚਿਤ ਕੀਤੇ ਬਿਨਾਂ ਵੀ ਆਰੰਭ (initialize) ਕਰ ਸਕਦੇ ਹਾਂ ਜਿਵੇਂ

```
char name [ ] = {'R', 'a', 'm', 'a', '\0'};
```

ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਅਸੈਸ ਲਈ (Program to Access strings)

```
#include<stdio.h>
#include<string.h>
void main()
{
 int a ,count ;
 char my_string[9];          //initialize each element of the string equal to a
                             // space and then add the end-of-string marker.
 for (a=0;a<8;a++);          //Line1
 {
  my_string[a]=' ';
 }
 my_string[8]='\0';          //Line2
                             // to set the string equal to the word GO
 my_string[0]='G';           //Line3
 my_string[1]='O';           // Line4
 my_string[2]='A';
 my_string[3]='\0';          //Line5
 printf("String = %s\n" ,my_string);
}
```

ਪਰਿਣਾਮ (Output)

String = GOA

ਸਪਸ਼ਟੀਕਰਨ (Explanation)

ਰੇਖਾ 1 ਤੋਂ ਰੇਖਾ 2 ਰਾਹੀਂ ਸਟਰਿੰਗ (string) ਵਿਚ ਹਰ ਇਕ ਐਲੀਮੈਂਟ (element) ਨੂੰ ਇਕ ਸਥਾਨ-ਚਿੰਨ੍ਹ (space character) ਵਿਚ ਆਰੰਭ (initialize) ਕਰਦਾ ਹੈ ਅਤੇ ਫਿਰ end-of-string ਨਿਸ਼ਾਨ ਨੂੰ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਵਿਚ ਰੱਖਦਾ ਹੈ।

ਰੇਖਾ 4 ਰਾਹੀਂ ਰੇਖਾ 3 GO ਨੂੰ ਸਟਰਿੰਗ (string) ਦੇ ਪਹਿਲੇ ਦੋ ਐਲੀਮੈਂਟਸ (elements) GO ਵੱਲ ਭੇਜਦੀ ਹੈ ਅਤੇ end-of-string ਨਿਸ਼ਾਨ ਨੂੰ ਸਟਰਿੰਗ (string) ਦੇ ਐਲੀਮੈਂਟ ਨੰਬਰ 3 ਵਿਚ ਰਖਦੀ ਹੈ ਸਟਰਿੰਗ (string) ਵਿਚ ਹੋਰ ਐਲੀਮੈਂਟ (element) ਬਿਲਕੁਲ ਨਹੀਂ ਬਦਲਦੇ। ਰੇਖਾ 5 string = GoA ਨੂੰ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਦੀ ਹੈ। ਕੇਵਲ 3 ਚਿੰਨ੍ਹ (character) ਹੀ ਪ੍ਰਦਰਸ਼ਿਤ (display) ਹੁੰਦੇ ਹਨ ਕਿਉਂ ਜੋ ਸਟਰਿੰਗ (string) ਤਿੰਨ ਚਿੰਨ੍ਹ (character) ਲੰਬੀ ਹੁੰਦੀ ਹੈ। ਜੇਕਰ '\O' ਚਿੰਨ੍ਹ ਸਟਰਿੰਗ (character string) ਦੇ ਅੰਤ ਤੇ ਨਹੀਂ ਹੁੰਦਾ, printf ਫੰਕਸ਼ਨ ਮੈਮੋਰੀ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਦਾ ਹੈ ਜਿੰਨੀ ਦੇਰ ਕਿਸੇ ਇਕ ਨੂੰ ਲੱਭ ਨਹੀਂ ਲੈਂਦਾ। end of string ਨਿਸ਼ਾਨ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਤੇ ਰੱਖੇ ਜਾਣੇ ਚਾਹੀਦੇ ਹਨ ਤਾਂ ਜੋ ਸਟਰਿੰਗ (string) ਫੰਕਸ਼ਨਜ਼ ਉਚਿੱਤ ਢੰਗ ਨਾਲ ਕੰਮ ਕਰ ਸਕਣ।

ਇਕ ਚਿੰਨ੍ਹ (character) ਜਿਹੜਾ ਸਿੰਗਲ ਕੋਟ (single quote) ਵਿਚ ਹੁੰਦਾ ਹੈ ਉਸਨੂੰ ਚਿੰਨ੍ਹ ਕਾਂਨਸਟੈਂਟ (character constant) ਕਹਿੰਦੇ ਹਨ। ਮਿਸਾਲ ਲਈ : **'A' or 'a'** ਜਾਂ **'B' or 'b'**

ਇਹ ਸਾਰੇ ਕਾਂਸਟੈਂਟ ਹਨ ਜਿਹਨਾਂ ਦੀ ASCII ਚਿੰਨ-ਸੈਟ (character Set) ਵਿਚ ਇਕਲਾ ਮੁੱਲ ਹੁੰਦਾ ਹੈ।

ਜਿਵੇਂ ਕਿ : **X = 'A';** (ਇਸ 'A' ਦਾ ASCII ਚਿੰਨ ਸੈਟ (character set) ਮੁੱਲ 65 ਹੁੰਦਾ ਹੈ।ਇਸ ਲਈ)
X = 65,

ਇਕ ਸਟਰਿੰਗ (string) ਜਿਸਦੇ ਵਿਚ ਇਕ ਤੋਂ ਜ਼ਿਆਦਾ ਚਿੰਨ (character) ਹੁੰਦੇ ਹਨ ਉਨ੍ਹਾਂ ਨੂੰ ਡਬਲ ਕੋਟਸ (double quotes) ਵਿਚ ਲਿਖਿਆ ਜਾਂਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ : "Hello World"

(ਨੋਟ : **ASCII** ਟੇਬਲ ਪੇਜ 108 ਤੇ ਦਿੱਤਾ ਗਿਆ ਹੈ)

ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਦੀ ਪ੍ਰਿੰਟਿੰਗ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਨ ਲਈ (Program to demonstrate printing of a string)

ਮਿਸਾਲ 1: To print a string

```
#include<stdio.h>
#include<string.h>
void main ( )
{
   char name[ ]="RAM";
   int a = 0;
   while (a<=2)
   {
     printf("%c",name[a]);
     a++;
   }
}
```

Here in this program , a character array name [ ] has been initialized , and then printed out the elements of this array within the while loop.

ਪਰਿਣਾਮ (Output) :

RAM

ਮਿਸਾਲ 2: ਇਹ ਪ੍ਰੋਗਰਾਮ ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਪ੍ਰਵਾਨ ਕਰਦਾ ਹੈ ਅਤੇ ਇਸ ਵਿਚ ਚਿੰਨ੍ਹਾਂ (character) ਦੀ ਸੰਖਿਆ ਦੀ ਗਿਣਤੀ ਕਰਦਾ ਹੈ ਅਤੇ ਪ੍ਰਿੰਟ ਕਰਦਾ ਹੈ

```
#include<stdio.h>
void main()
{
int ccont = 0, index;
char letters[20];
printf("Enter I-D character array\n");
scanf("%s", letters);
for (index = 0; letters[index] != '\0'; index++)
{
ccont +=1;
}
printf("%s contains %d character\n", letters, ccont);
}
```

ਪਰਿਣਾਮ (Output)

Enter I-D character array
honest
honest contains 6 characters

ਸਟਰਿੰਗਜ਼ ਪੜ੍ਹਨਾ ਅਤੇ ਲਿਖਣਾ (Reading & Writing Strings)

ਇਨਪੁਟ/ਆਉਟਪੁਟ ਸਟੇਟਮੈਂਟਸ ਦੀ ਵਰਤੋਂ ਨਾਲ ਇਕ ਸਟਰਿੰਗ (String) ਨੂੰ ਲਿਖਿਆ ਜਾਂ ਪੜ੍ਹਿਆ ਜਾ ਸਕਦਾ ਹੈ।

ਸਟਰਿੰਗ ਪੜ੍ਹਨਾ (Reading Strings)

ਨਿਮਨਲਿਖਿਤ ਇਕ ਸਟਰਿੰਗ (String) ਨੂੰ ਪੜ੍ਹਣ ਲਈ ਇਨਪੁਟ ਫੰਕਸ਼ਨ ਨੂੰ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ।

- **scanf ( ) ਦੇ ਨਾਲ %s format specification**
- **gets( ) function**
- **getchar ( ) function**

ਸਟਰਿੰਗ ਲਿਖਣਾ (Writing Strings)

ਨਿਮਨਲਿਖਿਤ ਇਕ ਸਟਰਿੰਗ (String) ਜਾਂ ਸਟਰਿੰਗਜ਼ (Strings) ਦੇ ਸਮੂਹ ਨੂੰ ਆਉਟਪੁਟ ਫੰਕਸ਼ਨਜ਼ (functions) ਦੀ ਵਰਤੋਂ ਨਾਲ ਮੋਨੀਟਰ ਤੇ ਪ੍ਰਦਰਸ਼ਿਤ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

- **printf( ) ਦੇ ਨਾਲ %s as format specification**
- **puts( ) function**
- **putchar( ) function**

ਮੋਨੀਟਰ ਤੋਂ ਸਟਰਿੰਗ ਪੜ੍ਹਣਾ (Reading Strings from the monitor):

ਫੰਕਸ਼ਨ (function) scanf ਦੇ ਨਾਲ %s format specification ਨੂੰ ਮੋਨੀਟਰ (monitor) ਤੋਂ ਕਰੈਕਟਰ ਸਟਰਿੰਗ (character string) ਪੜ੍ਹਣ ਦੀ ਲੋੜ ਹੁੰਦੀ ਹੈ।

ਮਿਸਾਲ ਲਈ :
```
char address [15];
scanf ("%s",address);
```

scanf ਸਟੇਟਮੈਂਟ (statement) ਦੀ ਵਰਤੋਂ ਨਾਲ ਇਨਪੁੱਟ ਕਰਨ ਵਿਚ ਸਮੱਸਿਆ ਇਹ ਹੈ ਕਿ ਇਹ ਸਟੇਟਮੈਂਟ (statement) ਵਿਚ ਇਕ ਖਾਲੀ ਸਥਾਨ ਮਿਲਣ ਤੇ ਇਸ ਨੂੰ ਖਤਮ ਕਰ ਦਿੰਦੀ ਹੈ।

ਮੰਨ ਲਉ ਕਿ ਅਸੀਂ ਸਟਰਿੰਗ (string) **'New York'** ਟਾਈਪ ਕੀਤਾ ਹੈ ਤਾਂ ਕੇਵਲ ਸਟਰਿੰਗ (string) 'New' ਹੀ ਪੜ੍ਹੀ ਜਾਵੇਗੀ ਕਿਉਂ ਜੋ ਸ਼ਬਦ **'New'** ਦੇ ਬਾਅਦ ਖਾਲੀ ਸਥਾਨ (space) ਹੈ ਇਸ ਲਈ ਇਹ ਸਟਰਿੰਗ (string) ਨੂੰ ਖਤਮ ਕਰ ਦੇਵੇਗੀ।

scanf() ਬਹੁ-ਸ਼ਬਦੀ ਸਟਰਿੰਗ (string) ਪ੍ਰਾਪਤ ਕਰਨ ਦੇ ਕਾਬਿਲ ਨਹੀਂ ਹੈ ਇਸ ਲਈ ਨਾਮ ਜਿਵੇਂ ਕਿ 'Ram Gopal' ਗੈਰ-ਪ੍ਰਵਾਨਯੋਗ (unacceptable) ਹੋਵੇਗਾ। ਇਸ ਦੇ ਹੱਲ ਲਈ gets function () ਦੀ ਵਰਤੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। gets function () ਅਤੇ ਇਸਦੇ counter part puts() ਦੀ ਵਰਤੋਂ ਹੇਠਾਂ ਦਰਸਾਈ ਗਈ ਹੈ।

```
#include<stdio.h>
#include<string.h>
void main()
{
 char name[25];
 printf("Enter your full name ");
  gets(name);
  puts("hello!") ;
  puts (name);
 }
```

ਪਰਿਣਾਮ (output)

```
Enter your full name Ram Gopal
hello!
Ram Gopal
```

puts () ਇਕ ਸਮੇਂ ਵਿਚ ਕੇਵਲ ਇਕੋ ਹੀ ਸਟਰਿੰਗ ਪ੍ਰਦਰਸ਼ਿਤ (string display) ਕਰ ਸਕਦੀ ਹੈ ਇਸ ਲਈ ਇਸ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਦੋ puts () ਵਰਤੇ ਗਏ ਹਨ। ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਪ੍ਰਦਰਸ਼ਿਤ (display) ਕਰਨ ਤੇ printf() ਦੇ ਉਲਟ puts() ਕਰਸਰ (cursor) ਨੂੰ ਅਗਲੀ ਲਾਈਨ ਤੇ ਰਖਦਾ ਹੈ। ਭਾਵੇਂ gets() ਇਕ ਸਮੇਂ ਤੇ ਇਕੋ ਸਟਰਿੰਗ (string) ਹੀ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ ਪਰ ਇਹ ਬਹੁ-ਸ਼ਬਦੀ ਸਟਰਿੰਗ (string) gets () ਦੀ ਵਰਤੋਂ ਨਾਲ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ।

ਸਾਧਾਰਣ (simple) I/O getchar, putchar, printf

I/O ਫਾਰਮੇਟ (format) ਲਈ function printf () & scanf () ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

getchar ਅਤੇ putchar ਸੀ (c) ਦੇ ਬੁਨਿਆਦੀ I/O ਲਾਇਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨ (function) ਹਨ।

getchar()

ਇਹ ਸਟੈਂਡਰਡ ਇਨਪੁਟ (standard input) ਜਿਵੇਂ ਕਿ ਕੀ-ਬੋਰਡ ਤੋਂ ਇਕ ਚਿੰਨ੍ਹ (character) ਲੈਂਦਾ ਹੈ, ਜਦੋਂ ਵੀ ਇਸ ਨੂੰ ਕਾਲ (call) ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਇਹ ਉਸੇ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਫੰਕਸ਼ਨ (function) ਦੇ ਮੁੱਲ ਵਜੋਂ ਵਾਪਸ ਭੇਜਦਾ ਹੈ। ਇਹ ਇਸ ਉਪਰੰਤ ਖਾਲੀ ਅੱਖਰ '\0' ਨੂੰ ਵਾਪਸ ਮੋੜਦਾ ਹੈ ਜਦੋਂ ਉਸ ਫਾਈਲ ਦੇ ਅੰਤ ਤੇ ਪੁੱਜ ਜਾਂਦਾ ਹੈ ਜਿਹੜੀ ਇਹ ਪੜ੍ਹਦਾ ਹੈ।

putchar ( )

ਇਹ ਫੰਕਸ਼ਨ (function) ਇਕ ਸਮੇਂ ਤੇ ਇਕ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਸਟੈਂਡਰਡ ਆਊਟਪੁਟ (standard output) ਤੇ ਭੇਜਦਾ ਹੈ ਇਸ ਲਈ ਉਪਰੋਕਤ ਪ੍ਰੋਗਰਾਮ ਇਕ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਪੜ੍ਹਦਾ ਹੈ ਅਤੇ ਇਸ ਨੂੰ ਵਾਪਸ ਲਿਖਦਾ ਹੈ।

ਮਿਸਾਲ ਲਈ ਹੇਠਲੇ ਪ੍ਰੋਗਰਾਮ ਤੇ ਵਿਚਾਰ ਕਰੋ ਜਿਹੜਾ ਫੰਕਸ਼ਨ (function) <ctype.h>; ਵਿਚ ਪਰਿਭਾਸ਼ਤ ਕੀਤਾ ਗਿਆ ਹੈ। ਇਹ ਅਪਰ-ਕੇਸ (upper case) ਅੱਖਰ ਨੂੰ ਲੋਅਰ-ਕੇਸ (lower case) ਵਿਚ ਬਦਲਦਾ ਹੈ ਅਤੇ ਬਾਕੀ ਦੇ ਚਿੰਨ੍ਹਾਂ (character) ਨੂੰ ਬਿਨਾ ਛੂਹਿਆਂ ਹੀ ਮੋੜ ਦਿੰਦਾ ਹੈ ਜਿਵੇਂ ਕਿ <stdio.h> ਵਿਚ getchar ਅਤੇ putchar ਫੰਕਸ਼ਨ (function) ਹੁੰਦੇ ਹਨ ਅਤੇ ਇਸੇ ਤਰ੍ਹਾਂ <ctype.h> ਵਿਚ tolower() function ਹੁੰਦਾ ਹੈ, ਇਹ ਮੈਕਰੋਸ (macros) ਦੀ ਤਰ੍ਹਾਂ ਹੁੰਦੇ ਹਨ ਅਤੇ ਇਸ ਪ੍ਰਕਾਰ ਮਾਈਕਰੋ (micro) ਦੀ ਮਦਦ ਦੇ ਨਾਲ ਇਕ ਚਿੰਨ੍ਹ ਨੂੰ ਪੜ੍ਹਨ ਲਈ ਫੰਕਸ਼ਨ (function) ਸਦਣ ਦੇ ਕਾਰਜ ਨੂੰ ਘਟਾਇਆ ਜਾ ਸਕਦਾ ਹੈ।

```
#include<stdio.h>
#include<ctype.h>
int main()                    //convert input to lower case
{
   int c;
   while((c = getchar()) != EOF)
              putchar(tolower(c));
   return(0);
}
```

[Note: This program will give you a blank screen & if you type and Press ^Z to exit]

RAM

ਪਰਿਣਾਮ (output)

ram

ਇਕਲਾ ਚਿੰਨ੍ਹ ਇਨਪੁਟ / ਆਊਟਪੁਟ (Single character input/output)

ਇਨਪੁਟ /ਆਊਟਪੁਟ (input/output) ਵਿਚ ਕੀਤਾ ਗਿਆ ਬੁਨਿਆਦੀ ਕੰਮ ਸਟੈਂਡਰਡ (standard) ਇਨਪੁੱਟ ਯੰਤਰ (device) ਤੋਂ ਚਿੰਨ੍ਹ (character) ਪੜ੍ਹਨਾ ਹੁੰਦਾ ਹੈ ਜਿਵੇਂ ਕਿ ਕੀ-ਬੋਰਡ ਅਤੇ ਆਊਟਪੁਟ (output) ਲਈ ਇਸ ਨੂੰ ਆਮ ਤੌਰ ਤੇ ਸਕਰੀਨ ਤੇ ਲਿਖਣਾ। ਸਟੈਂਡਰਡ ਇਨਪੁੱਟ ਡਿਵਾਈਸ (device) ਤੋਂ ਚਿੰਨ੍ਹ (character) ਪੜ੍ਹਨ ਲਈ getchar ਫੰਕਸ਼ਨ (function) ਵਰਤੇ ਜਾ ਸਕਦੇ ਹਨ।

The getchar ਰੂਪ ਹੈ   **Variable name= getchar();**

ਵੇਰੀਏਬਲ (variable) ਨਾਮ ਇਕ ਵੈਲਿਡ (valid) ਸੀ 'C' ਵੇਰੀਏਬਲ (variable) ਹੈ ਜੋ ਪਹਿਲਾਂ ਹੀ ਡਿਕਲੇਅਰ ਕੀਤਾ ਜਾ ਚੁੱਕਾ ਹੈ ਅਤੇ ਜਿਸ ਦੀ ਟਾਈਪ ਕੈਰ (char) ਹੁੰਦੀ ਹੈ।

ਮਿਸਾਲ ਪ੍ਰੋਗਰਾਮ :

```
#include<stdio.h>                      // assigns stdio.h header file to your program
void main()                            // indicates the starting point of the program
{
 char C;                               // variable declaration
 printf ("Type one character:");       // message to user
 C = getchar();                        // get a character from key board and stores it in variable C.
 printf ("The character you typed is = %c", C);  // output
}                                      // Statement which displays value of C on standard screen
```

Output

Type one character: D
The Character you typed is = D

putchar ਫੰਕਸ਼ਨ (function) ਜੋ getchar ਫੰਕਸ਼ਨ (function) ਦੀ ਤਰ੍ਹਾਂ ਹੈ, ਇਹ ਇਕ ਵਾਰ ਚਿੰਨ੍ਹਾਂ ਨੂੰ ਮੋਨੀਟਰ (monitor) ਤੇ ਲਿਖ ਸਕਦਾ ਹੈ। ਆਮ ਰੂਪ ਹੈ **putchar (variable name);**

ਜਿਹੜਾ ਵੇਰੀਏਬਲ (variable) ਵੈਲਿਡ ਸੀ (C) ਵੇਰੀਏਬਲ (variable) ਹੈ ਜੋ ਪਹਿਲਾਂ ਹੀ ਡਿਕਲੇਅਰ (declare) ਕੀਤਾ ਗਿਆ ਹੈ।

ਮਿਸਾਲ : **putchar ()**

ਵੇਰੀਏਬਲ (variable) ਸੀ (C) ਵਿਚ ਸਟੋਰ ਕੀਤੇ ਗਏ ਮੁੱਲ ਨੂੰ ਸਟੈਂਡਰਡ ਸਕਰੀਨ ਤੇ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਦਾ ਹੈ। ਪਰਸਪਰ ਪ੍ਰਭਾਵੀ ਵਾਤਾਵਰਣ ਵਿਚ ਪ੍ਰੋਗਰਾਮ ਇਕ getchar ਫੰਕਸ਼ਨ (function) ਨੂੰ ਦਰਸਾਉਂਦਾ ਹੈ।

ਮਿਸਾਲ ਪ੍ਰੋਗਰਾਮ :

```
#include<stdio.h>                          // inserts stdio.h header file into the Program
void main ( )                              // Beginning of main function.
{
 char n1;                                  // character declaration of variable n1.
 printf("Please enter one character  ");   // message to user
 n1 = getchar() ;                          // assign the keyboard input value to n1.
 putchar (n1);                             // output n1 value to standard screen.
}
```

Output

Please enter one character: G
G

ਇਨ੍ਹਾਂ ਦੋ ਫੰਕਸ਼ਨਾਂ (functions) ਦੀ ਵਰਤੋਂ ਨਾਲ ਹੇਠਾਂ ਦਿੱਤਾ ਗਿਆ ਪ੍ਰੋਗਰਾਮ ਸੰਪੂਰਨ ਕਰੋ, ਇਨਪੁਟ (input) ਨੂੰ ਕਾਪੀ ਕਰਨ ਲਈ, ਇਕੋ ਵਾਰ ਇਕ ਚਿੰਨ੍ਹ (character) ਇਨਪੁਟ (input) ਅਤੇ ਆਊਟਪੁਟ (output) ਦੀ ਜਾਂਚ ਕਰੋ।

ਮਿਸਾਲ : /* copy input to output after inputting single word*/

```
#include<stdio.h>
int main()
{
 int c;
  c = getchar();
  while(c!= EOF)
  {
    putchar(c);
    c = getchar();
  }
  return(0);
}
```

This program will give blank sceen after execution of the program , when you enter a word , it will display the same word

Output

SHAM

SHAM

ਸਪਸ਼ਟੀਕਰਨ (Explanation)

ਇਹ ਇਕ ਚਿੰਨ੍ਹ (character) ਪੜ੍ਹੇਗਾ ਅਤੇ ਜੇਕਰ ਇਹ EOF ਕੋਡ ਨਹੀਂ ਹੈ ਤਾਂ while loop ਵਿਚ ਪ੍ਰਵੇਸ਼ ਕਰੇਗਾ. ਇਕ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਛਾਪਦੇ ਅਤੇ ਇਕ ਹੋਰ ਨੂੰ ਪੜ੍ਹਦੇ ਹੋਏ ਜਦ ਤੱਕ character read "EOF" ਨਹੀਂ ਹੋ ਜਾਂਦੀ।

ਹਰ ਇਕ ਚਿੰਨ੍ਹ (character) ਦੀ ਨਵੀ ਰੇਖਾ ਚਿੰਨ੍ਹ(character) \n ਦੇ ਨਾਲ ਤੁਲਨਾ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਅਤੇ ਜੇਕਰ ਬਰਾਬਰ ਨਾ ਹੋਵੇ ਤਾਂ ਸਕਰੀਨ ਤੇ putchar () ਦੇ ਨਾਲ ਪ੍ਰਦਰਸ਼ਿਤ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਜਦੋਂ ਨਵੀ ਰੇਖਾ getchar () ਦੁਆਰਾ ਵਾਪਸ ਮੁੜਦੀ ਹੈ ਤਾਂ while loop ਸਮਾਪਤ ਹੋ ਜਾਂਦਾ ਹੈ।

ਸਮੱਸਿਆ (Problem)

ਸਮੱਸਿਆ ਹੇਠਾਂ ਲਿਖੀ ਗਈ ਹੈ ਜਿਹੜੀ ਟੈਕਸਟ ਦੀ ਇਕ ਲਾਈਨ ਨੂੰ "ABC govt school" ਵਜੋਂ ਪੜ੍ਹਦੀ ਹੈ। ਪ੍ਰੋਗਰਾਮ ਟੈਕਸਟ ਦੀ ਉਸੀ ਲਾਈਨ **"ABC govt. school"** ਨੂੰ ਸਕਰੀਨ ਤੇ ਆਊਟਪੁਟ ਵਿਚ ਦਿਖਾਣਾ ਚਾਹੀਦਾ ਹੈ ਇਸ ਲਈ ਹੇਠਾਂ ਦਿਤੇ ਪ੍ਰੋਗਰਾਮ ਦਾ ਵਿਸ਼ਲੇਸ਼ਣ ਕਰੋ ਅਤੇ ਇਸਨੂੰ ਸਹੀ ਕਰੋ।

ਮਿਸਾਲ : /* To read a line of text *\

```
#include<stdio.h>
  void main()
  {
   char text[40],c ;
   int a = 0 ;
   printf("Enter text of line : \n");
   while((c = getchar())!= '\n')
    {
    text[a] = c;
      a++ ;
    }
   text[a] = '\0';
  printf("\n%s",text);
  }
```

ਪਰਿਣਾਮ (Output) :

Enter the Text of line : ABC govt School

Output: ABC govt School

ਸਪਸ਼ਟੀਕਰਣ (Explanation)

ਰੇਖਾ 7 ਤੇ getchar ਫੰਕਸ਼ਨ (function) ਨੂੰ call ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ stdin (Standard input) (ਕੀ-ਬੋਰਡ ਤੋਂ ਇਕ ਚਿੰਨ੍ਹ (character) ਪ੍ਰਾਪਤ ਕਰਨ ਦੀ ਇੰਤਜ਼ਾਰ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਕਿਉਂ ਜੋ getchar () ਇਕ ਇਨਪੁੱਟ (input) ਫੰਕਸ਼ਨ (function) ਹੁੰਦਾ ਹੈ ਜਦ ਤੱਕ ਅਸੀ Enter key ਨੂੰ ਦਬਾਉਂਦੇ ਨਹੀ ਉਨੀ ਦੇਰ ਕੋਈ ਚਿੰਨ੍ਹ (character) ਪ੍ਰਾਪਤ ਨਹੀ ਹੁੰਦਾ। ਭਾਵੇਂ ਸਾਡੇ ਵਲੋਂ ਦਬਾਈ ਜਾਂਦੀ ਹਰੇਕ ਕੀ (key) ਇਕਦਮ ਸਕਰੀਨ ਤੇ ਦਿਖਾਈ ਦਿੰਦੀ ਹੈ।

While loop getchar() ਤੋ ਚਿੰਨ੍ਹ (character) ਪ੍ਰਵਾਨ ਕਰਦਾ ਹੈ ਜਦ ਤਕ ਨਵੀ ਰੇਖਾ ਚਿੰਨ੍ਹ (character) ਪਹੁੰਚ ਨਹੀ ਜਾਂਦੀ ਜਾਂ 80 ਚਿੰਨ੍ਹ (character) ਪੜ੍ਹੇ ਨਹੀਂ ਜਾਂਦੇ। ਹਰੇਕ ਚਿੰਨ੍ਹ (character) ਇਕ ਐਰੇ (array) ਨੂੰ ਸਮਰਪਿਤ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਜਿਸਨੂੰ ਬਫਰ (buffer) ਕਹਿੰਦੇ ਹਨ। ਜਦੋ ਚਿੰਨ੍ਹਾਂ (characters) ਦੀ ਇਨਪੁੱਟ (input)

ਸਮਾਪਤ ਹੁੰਦੀ ਹੈ. ਰੇਖਾ 15 ਐਰੇ (array) ਦੇ ਅੰਤ ਖਾਲੀ '\O' ਲਿਖਦੀ ਹੈ ਤਾਂ ਜੋ ਰੇਖਾ 17 ਤੇ print ਫੰਕਸ਼ਨ (function) ਪ੍ਰਵੇਸ਼ ਕਰਨ ਵਾਲੀ ਸਟਰਿੰਗ (string) ਨੂੰ ਛਾਪ ਸਕੇ।

ਜਦੋਂ ਅਸੀਂ Enter ਕੀ (key) ਨੂੰ ਦਬਾਉਂਦੇ ਹਾਂ, ਸਮੂਹ ਚਿੰਨ੍ਹ (character) ਜਿਹੜੇ ਅਸੀਂ ਪ੍ਰਵੇਸ਼ ਕੀਤੇ ਹਨ ਨਵੀਂ ਰੇਖਾ ਸਮੇਤ stdin ਨੂੰ ਉਪਰੇਟਿੰਗ ਸਿਸਟਮ ਦੁਆਰਾ ਭੇਜੇ ਜਾਂਦੇ ਹਨ। getchar ਫੰਕਸ਼ਨ (function) ਇਕ ਟਾਈਮ ਤੇ ਇਕੋ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਵਾਪਸ ਮੋੜਦਾ ਹੈ ਅਤੇ ਹਰ ਇਕ ਚਿੰਨ੍ਹ ਨੂੰ ਸੀ 'C' ਵੇਰੀਏਬਲ ਵਿਚ ਸਮਰਪਣ ਕਰਦਾ ਹੈ।

getchar () ਫੰਕਸ਼ਨ (function) ਦੀ ਵਰਤੋਂ ਖਰੜੇ (texts) ਦੀ ਸਮੁੱਚੀ ਲਾਈਨ ਲਈ ਕੀਤੀ ਜਾ ਸਕਦੀ ਹੈ ਜਿਵੇਂ ਦਿਖਾਈ ਗਈ ਹੈ। ਭਾਵੇਂ ਕਿ ਹੋਰ ਇਨਪੁੱਟ (input) ਫੰਕਸ਼ਨ (function) ਵੀ ਇਸ ਕੰਮ ਲਈ ਜ਼ਿਆਦਾ ਢੁਕਵੇਂ ਹਨ।

ਸਟਰਿੰਗ ਇਨਪੁਟ ਅਤੇ ਆਊਟਪੁਟ (String input and output)

gets function, ਸਟਰਿੰਗ (string) ਨੂੰ ਸਟੈਂਡਰਡ ਇਨਪੁਟ-ਡਿਵਾਈਸ (standard input device) ਤੋਂ ਮੁਕਤ (relieves) ਕਰਦੀ ਹੈ ਜਦਕਿ puts function, ਸਟੈਂਡਰਡ ਆਊਟਪੁਟ-ਡਿਵਾਈਸ (standard output device) ਲਈ ਸਟਰਿੰਗ (string) ਦਾ ਆਊਟਪੁੱਟ ਦਿੰਦਾ ਹੈ।

gets function ਵਲੋਂ ਇਕ ਪੈਰਾਮੀਟਰ ਵਜੋਂ ਸਟਰਿੰਗ (string) ਦੇ ਨਾਮ ਨੂੰ ਪ੍ਰਵਾਨ ਕਰ ਲਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਸਟਰਿੰਗ (string) ਨੂੰ ਉਨ੍ਹਾਂ ਚਿੰਨ੍ਹਾਂ (characters) ਦੇ ਨਾਲ ਭਰਿਆ ਜਾਂਦਾ ਹੈ ਜਿਹੜੇ ਨਵੀਂ ਰੇਖਾ ਚਿੰਨ੍ਹ (character) ਤੱਕ ਕੀ-ਬੋਰਡ ਤੋਂ ਇਨਪੁੱਟ (input) ਹੁੰਦੇ ਹਨ। (ਉਹ ਵੀ ਜਦ ਤੱਕ ਅਸੀਂ Enter ਕੀ (key) ਨੂੰ ਦਬਾਉਂਦੇ ਹਨ)।

puts function ਵਲੋਂ ਸਟੈਂਡਰਡ-ਸਕਰੀਨ (standard screen) ਤੇ ਇਸਦੇ ਪੈਰਾਮੀਟਰਾਂ ਦੇ ਨਾਲ ਸਟੋਰ ਕੀਤੀ ਗਈ ਵਿਸ਼ਾ ਸੂਚੀ ਨੂੰ ਪ੍ਰਦਰਸ਼ਿਤ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।

gets function ਦਾ ਸਟੈਂਡਰਡ ਰੂਪ ਹੁੰਦਾ ਹੈ gets (str) ਜਿੱਥੇ str ਇਕ ਸਟਰਿੰਗ-ਵੇਰੀਏਬਲ (string variable) ਹੈ।
puts character ਦਾ ਸਟੈਂਡਰਡ ਰੂਪ ਹੁੰਦਾ ਹੈ puts (str) ਜਿੱਥੇ str ਇਕ ਸਟਰਿੰਗ-ਵੇਰੀਏਬਲ (string variable) ਹੈ।

ਮਿਸਾਲ

```
#include<stdio.h >
void main()
{
char s[80];
printf("Type a string less than 80 characters:\n");
gets(s);
printf("The string typed is:");
puts(s);
}
```

Output

Type a string less than 80 characters:
God is Great
The string types is: God is Great

ਚਿੰਨ੍ਹਾਂ ਦੇ ਅਰਥਮੇਟਿਕ ਔੱਪਰੇਸ਼ਨ (Arithmetic operations on characters)

ਅਸੀਂ ਕਰੈਕਟਰਸ (characters) ਦਾ ਜੋੜ-ਤੋੜ (manupilation) ਵੀ ਕਰ ਸਕਦੇ ਹਾਂ ਜਿਵੇਂ ਕਿ ਸੀ (C) ਭਾਸ਼ਾ ਵਿਚ ਅੰਕਾਂ ਦਾ ਕਰਦੇ ਹਾਂ। ਜਦੋਂ ਵੀ ਸਿਸਟਮ (system) ਕਰੈਕਟਰਸ (characters) ਡਾਟਾ ਦਾ ਸਾਹਮਣਾ ਕਰਦੇ ਹਨ ਤਾਂ ਇਹ ਸਿਸਟਮ (system) ਦੁਆਰਾ ਖੁਦ-ਬ-ਖੁਦ ਇੰਟੀਜ਼ਰ (integer) ਮੁੱਲ ਵਿਚ ਬਦਲ ਜਾਂਦੀ ਹੈ। ਅਸੀਂ ਹੇਠਾਂ ਲਿਖੀ ਗਈ ਵਿਧੀ ਦੇ ਨਾਲ ਇਕ ਇੰਟਰਫੇਸ (interface) ਵਜੋਂ ਚਿੰਨ੍ਹ (character) ਦੀ ਨੁਮਾਇੰਦਗੀ ਕਰ ਸਕਦੇ ਹਾਂ।

```
x='a';
printf("%d\n",x);
```

ਸਕਰੀਨ ਤੇ 97 ਪ੍ਰਦਰਸ਼ਿਤ ਹੋਵੇਗਾ। ਚਿੰਨ੍ਹਾਂ (characters) ਤੇ ਗਣਿਤ ਔਪਰੇਸ਼ਨ ਵੀ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ ਜਿਵੇਂ ਮਿਸਾਲ x='z'-1; ਇਕ ਵੈਲਿਡ-ਸਟੇਟਮੈਂਟ (valid statement) ਹੈ। z ਦਾ ASCII ਮੁੱਲ 122 ਹੈ ਇਸ ਲਈ ਸਟੇਟਮੈਂਟ (statement) 121 ਦਾ ਵੇਰੀਏਬਲ (variable) x ਅਸਾਈਨ (assign) ਕਰੇਗਾ।

ਰਿਲੇਸ਼ਨਲ-ਐਕਸਪ੍ਰੇਸ਼ਨ (relational expression) ਵਿਚ ਕਾਂਸਟੈਂਟ-ਚਿੰਨ੍ਹ (constant character) ਵਰਤਣੇ ਵੀ ਸੰਭਵ ਹਨ। ਮਿਸਾਲ ਲਈ ch>'a' && ch< = 'z' ਜਾਂਚ ਕਰੇਗਾ ਕਿ ਵੇਰੀਏਬਲ (variable) **ch** ਵਿਚ ਸਟੋਰ ਕੀਤੇ ਗਏ ਚਿੰਨ੍ਹ (character) ਲੋਅਰ ਕੇਸ (lower case) ਅੱਖਰ ਹਨ। ਇਕ ਚਿੰਨ੍ਹ (character) ਡਿਜੀਟਸ (digits) ਇਸਦੇ ਬਰਾਬਰ ਦੇ ਇੰਟੀਜ਼ਰ (integer) ਵਿਚ ਬਦਲਿਆ ਜਾ ਸਕਦਾ ਹੈ

ਮੰਨ ਲਓ 44 ਜੋ a=character-'1'; ਦਾ ਐਕਸਪ੍ਰੈਸ਼ਨ (expression) ਹੈ ਜਿਥੇ a ਨੂੰ ਇਕ ਇੰਟੀਜਰ-ਵੇਰੀਏਬਲ (integer variable) ਵਿਚ ਪਰਿਭਾਸ਼ਤ ਕੀਤਾ ਗਿਆ ਹੈ ਅਤੇ ਚਿੰਨ੍ਹ (character) ਵਿਚ ਮੁੱਲ 8 ਤਦ a=ASCII ਮੁੱਲ ਜੋ 8 ASCII value '1'=56-49=7 ਸ਼ਾਮਲ ਹੁੰਦਾ ਹੈ।

ਅਸੀਂ ਅੰਕਾਂ (digits) ਦੀ ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਉਨ੍ਹਾਂ ਦੇ ਬਰਾਬਰ ਇੰਟੀਜ਼ਰ (integer) ਮੁੱਲ ਵਿਚ ਬਦਲਣ ਲਈ ਸੀ (C) ਲਾਇਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨ (function) ਦੀ ਸਹਾਇਤਾ ਵੀ ਪ੍ਰਾਪਤ ਕਰ ਸਕਦੇ ਹਾਂ।

x=atoi(string) ਸਟਰਿੰਗ (string) ਵਿਚ ਫੰਕਸ਼ਨ ਦੀ ਆਮ ਬਣਤਰ : ਇਥੇ **x** ਇਕ ਇੰਟੀਜਰ-ਵੇਰੀਏਬਲ (integer variable) ਹੈ ਅਤੇ ਸਟਰਿੰਗ (string) ਇਕ ਚਿੰਨ੍ਹ ਐਰੇ (character array) ਹੈ ਜਿਸ ਵਿਚ ਅੰਕਾਂ ਦੀ ਸਟਰਿੰਗ (string) ਸ਼ਾਮਲ ਹੁੰਦੀ ਹੈ ਅਤੇ atoi() ਇਕ ਫੰਕਸ਼ਨ ਹੈ ਜੋ ਸਟਰਿੰਗ ਨੂੰ ਇੰਟੀਜ਼ਰ ਵਿਚ ਬਦਲਦਾ ਹੈ।

String operations (string.h)

ਸੀ (C) ਭਾਸ਼ਾ ਸ਼ਨਾਖਤ (recognizes) ਕਰਦੀ ਹੈ ਕਿ ਸਟਰਿੰਗ (string) ਇਕ ਐਰੇ (array) ਦੀ ਵੱਖਰੀ ਸ਼੍ਰੇਣੀ ਹੁੰਦੀ ਹੈ ਜੋ ਐਰੇ (array) ਦੀ I/O ਨੂੰ ਇਕ ਇਕਾਈ ਵਜੋਂ ਵਰਤਦੀ ਹੈ ਅਤੇ ਖਾਲੀ ਚਿੰਨ੍ਹ (character) ਰਾਹੀਂ ਸਮਾਪਤ ਹੋ ਜਾਂਦੀ ਹੈ। ਸੀ (C) ਲਾਇਬ੍ਰੇਰੀ ਵਲੋਂ ਬਹੁਤ ਸਾਰੇ ਸਟਰਿੰਗ (string) ਸੰਭਾਲ ਲਈ ਫਕੰਸ਼ਨ ਵਰਤੇ ਜਾਂਦੇ ਹਨ ਜਿਵੇਂ ਕਿ :

• ਲੰਬਾਈ (length)	(ਸਟਰਿੰਗ (string) ਵਿਚ ਚਿੰਨ੍ਹਾਂ (characters) ਦੀ ਗਿਣਤੀ)
• Concatenation	(ਦੋ ਜਾਂ ਵਧੇਰੇ ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਜੋੜਨਾ)
• Comparing two strings	ਦੋ ਸਟਰਿੰਗਜ਼ (strings) ਦੀ ਤੁਲਨਾ ਕਰਕੇ
• ਸਬਸਟਰਿੰਗ (substring)	(ਦਿਤੀ ਗਈ ਸਟਰਿੰਗ (string) ਵਿਚੋਂ ਛੋਟਾ ਸਟਰਿੰਗ ਕੱਢਣਾ)
• ਕਾਪੀ (Copy)	(ਇਕ ਸਟਰਿੰਗ (string) ਦੂਜੇ ਤੇ ਕਾਪੀ ਕਰੋ)

ਉੱਪਰ ਲਿਖੇ ਗਏ ਸਮੂਹ ਕੰਮ ਕਰਨ ਦੇ ਲਈ ਪ੍ਰੋਗਰਾਮ ਵਿਚ string ਲਾਇਬ੍ਰੇਰੀ ਹੈਡਰ ਫਾਈਲ ਸ਼ਾਮਲ ਕਰਨੀ ਜ਼ਰੂਰੀ ਹੈ।

strlen( ) function: ਇਹ ਫੰਕਸ਼ਨ (function) ਸਟਰਿੰਗ (string) ਵਿਚੋਂ ਚਿੰਨ੍ਹਾਂ (characters) ਦੀ ਗਿਣਤੀ ਕਰਦਾ ਹੈ ਅਤੇ ਵਾਪਸ ਕਰਦਾ ਹੈ। ਲੰਬਾਈ ਵਿਚ ਕੋਈ ਖਾਲੀ ਚਿੰਨ੍ਹ (null character) ਨਹੀਂ ਹੁੰਦਾ।

ਵਾਕ-ਰਚਨਾ (Syntax) **n = strlen(string) ;**

ਜਿਥੇ n ਇਕ ਇੰਟੀਜਰ-ਵੇਰੀਏਬਲ (integer variable) ਹੈ ਉਹ ਸਟਰਿੰਗ (string) ਦੀ ਲੰਬਾਈ ਦਾ ਮੁੱਲ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ।

ਮਿਸਾਲ : **length = strlen("Hollywood");**

ਫੰਕਸ਼ਨ (function) ਦੁਆਰਾ ਇੰਟੀਜਰ-ਵੇਰੀਏਬਲ (integer variable) ਲੰਬਾਈ ਲਈ ਸਟਰਿੰਗ (string) ਵਿਚ 9 ਚਿੰਨ੍ਹਾਂ (characters) ਦੀ ਗਿਣਤੀ ਸਮਰਪਿਤ ਹੋਵੇਗੀ।

ਮਿਸਾਲ : Strlen() ਫੰਕਸ਼ਨ ਵਰਤ ਕੇ ਸਟਰਿੰਗ ਦੀ ਲੰਬਾਈ ਪਤਾ ਕਰਨੀ
/*A program to find the length of the string using strlen() function*/

```
#include<stdio.h>
#include<string.h>
#include <conio.h>
void main()
{
 char name[100];
 int length;
 clrscr();
 printf("Enter the string");
 gets(name);
 length=strlen(name);
 printf("\nNumber of characters in the string is=%d" ,length);
}
```

ਪਰਿਣਾਮ (Output)

Enter the string   ABC
Number of characters in the string is= 3

strcat( ) function

ਜਦੋਂ ਤੁਸੀਂ ਦੋ ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਮਿਲਾਉਂਦੇ ਹੋ ਤਾਂ ਤੁਸੀ ਇਕ ਸਟਰਿੰਗ (string) ਦੇ ਗੁਣ ਕਿਸੇ ਹੋਰ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਵਿਚ ਜੋੜ ਰਹੇ ਹੁੰਦੇ ਹੋ, ਇਸ ਪ੍ਰਕ੍ਰਿਆ ਨੂੰ concatenation ਕਿਹਾ ਜਾਂਦਾ ਹੈ। strcat() function2 ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਇਕਠਿਆਂ ਮਿਲਾਉਂਦਾ ਹੈ।

ਇਹ ਨਿਮਨਲਿਖਤ ਰੂਪ ਧਾਰਦਾ ਹੈ : **strcat (string1, string2)**

string1 ਅਤੇ string2 ਚਿੰਨ੍ਹ-ਐਰੇ (character array) ਹਨ। ਜਦੋਂ function strcat ਲਾਗੂ ਹੁੰਦਾ ਹੈ ਤਾਂ string1 ਦੇ ਪਿਛੇ string2 ਜੁੜ ਜਾਂਦੀ ਹੈ। string2 ਵਿਚ ਕੋਈ ਤਬਦੀਲੀ ਨਹੀਂ ਹੁੰਦੀ।

ਮਿਸਾਲ :

```
strcpy(string1,"sri");
strcpy(string2,"Bhagavan");
printf("%s",strcat(string1,string2);
```

ਉਪਰੋਕਤ ਪ੍ਰੋਗਰਾਮ ਤੋਂ ਸਟਰਿੰਗ1 (string1) ਦਾ ਮੁੱਲ ਵੰਡਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ **Sribhagavan** ਬਣ ਜਾਂਦਾ ਹੈ। string2 "bhagavan" ਵਿਚ ਕੋਈ ਤਬਦੀਲੀ ਨਹੀ ਹੂੰਦੀ।

ਮਿਸਾਲ : ਦੋ ਸਟਰਿੰਗਜ਼ ਨੂੰ ਜੋੜਨ ਲਈ ਪ੍ਰੋਗਰਾਮ

```
//Program accepts two strings and concatenates them
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main( )
{
 char string1[10], string2[10] ;
 clrscr();
 printf("Enter the first string\n  ") ;
 scanf("%s", string1);
 printf("Enter the second string\n  ") ;
 scanf("%s", string2);
 printf ("\n The First string after appending second one is %s\n");
 strcat(string1,string2);
 printf("%s",string1);
}            //End of main ()
```

ਪਰਿਣਾਮ (Output)

```
Enter the first string
Ram
Enter the second string
Gopal
The First string after appending second one is
Ram Gopal
```

strcmp( ) function

ਫੰਕਸ਼ਨ (function) ਦੋ ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਚਿੰਨ੍ਹ /character (ASCII) ਦੇ ਨਾਲ ਤੁਲਨਾ ਕਰਦਾ ਹੈ ਅਤੇ ਤਿੰਨ ਮੁੱਲਾਂ ਨੂੰ ਵਾਪਸ ਕਰਦਾ ਹੈ {-1, 0, 1} strcmp () ਦੀ ਵਾਕ ਰਚਨਾ ਹੈ :

strcmp (string1, string2)

ਜਿਥੇ string1: ਚਿੰਨ੍ਹਾਂ (characters) ਦਾ ਇਕ ਡਾਈਮੈਂਨਸ਼ਨਲ ਐਰੇ (dimensional array), ਸਟਰਿੰਗ2 (string 2) ਵੀ ਚਿੰਨ੍ਹਾਂ (characters) ਦੀ ਇਕ ਡਾਈਮੈਂਨਸ਼ਨਲ ਐਰੇ (dimensional array) ਹੈ।

ਇਹ ਵਾਪਸ ਭੇਜਦੀ ਹੈ -1 ਨੂੰ ਜੇਕਰ ਪਹਿਲੀ ਸਟਰਿੰਗ (string) ਦਾ ਚਿੰਨ੍ਹ (character) ਮੁੱਲ ASCII ਦੂਜੀ ਸਟਰਿੰਗ (string) ਨਾਲੋਂ ਘੱਟ ਹੁੰਦਾ ਹੈ, 0 (ਜ਼ੀਰੋ) ਵਾਪਸ ਭੇਜਦਾ ਹੈ ਬਸ਼ਰਤੇ ਕਿ ਦੋਵੇਂ ਸਟਰਿੰਗ (string) ਬਰਾਬਰ ਹੋਣ ਅਤੇ 1 ਵਾਪਸ ਭੇਜਦਾ ਹੈ ਜੇ ਸਟਰਿੰਗ (string) ਦੇ ਚਿੰਨ੍ਹ (character) ਦਾ ASCII ਮੁੱਲ ਦੂਜੀ ਸਟਰਿੰਗ (string) ਨਾਲੋਂ ਵੱਡਾ ਹੁੰਦਾ ਹੈ। ਇਹ ਫੰਕਸ਼ਨ (function) ਵੱਖ ਵੱਖ ਢੰਗਾਂ ਨਾਲ ਦਿਖਾਏ ਜਾਂਦੇ ਹਨ ਜੋ ਕਿ ਮਿਸਾਲ (ਕ) ਅਤੇ ਮਿਸਾਲ (ਖ) ਵਿਚ ਦਰਸਾਏ ਗਏ ਹਨ:-

ਮਿਸਾਲ (ਕ): **p = strcmp ("ABC, abc")**

ਇਥੇ **A** (ਮਿਸਾਲ 65) ਦਾ ASCII ਮੁੱਲ **a** (ਮਿਸਾਲ 97) ਦੇ ASCII ਮੁੱਲ ਨਾਲੋ ਘੱਟ ਹੈ ਇਸ ਪ੍ਰਕਾਰ ਉਪਰੋਕਤ ਸਟੈਟਮੈਂਟ (statement) ਮੋੜਦਾ ਹੈ -32 (ਮਿਸਾਲ 65-97) ਅਤੇ **p** ਨੂੰ ਸਮਰਪਣ (assigned) ਕਰਦਾ ਹੈ।

ਮਿਸਾਲ (ਖ): strcmp("Newyork","Newyork") ਜ਼ੀਰੋ ਵਾਪਸ ਭੇਜੇਗਾ ਕਿਉਂਕਿ ਦੋਵੇ ਸਟਰਿੰਗ (string) ਬਰਾਬਰ ਹਨ। strcmp ("their","there") 9 ਮੋੜੇਗਾ ਜੋ ਕਿ ASCII **'i'** ਅਤੇ ASCII **'r'** ਦੇ ਦਰਮਿਆਨ ਨਿਉਮੈਰਿਕ ਅੰਤਰ ਹੁੰਦਾ ਹੈ।

strcmp ("The", "the") ਵਾਪਸ ਕਰੇਗਾ 32 ਜੋ ਕਿ ASCII **'T'** & ASCII **'t'** ਦੇ ਦਰਮਿਆਨ ਨਿਉਮੈਰਿਕ ਅੰਤਰ ਹੁੰਦਾ ਹੈ।

ਮਿਸਾਲ: ਇਹ ਪ੍ਰੋਗਰਾਮ ਦੋ ਸਟਰਿੰਗ (string) ਪੜ੍ਹਦਾ ਹੈ ਅਤੇ ਤੁਲਨਾ ਕਰਦਾ ਹੈ ਕਿ ਇਹ ਬਰਾਬਰ ਹਨ ਜਾਂ ਨਹੀਂ
(This program reads two strings & compares them to test whether they are equal or not)

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
char stringl[25],string2[25];
clrscr();
printf ("\n Enter the first string\n    ");
gets (stringl);
printf("\n Enter the second string\n  ");
gets (string2);
if (strcmp(stringl, string2)== 0)
printf("Strings are equal\n");
else
printf("Strings are not equal\n");
}
```

ਪਰਿਣਾਮ (Output):

Enter the first string
good
Enter the second string
bad
Strings are not equal

Remember the facts about strings:

Value Returned	Meaning
Less then Zero (negative)	str1 < str2 , In dictionary order str1 is first than str2.
Zero	str1 is identical to str2
greater than Zero (positive)	str1 > str2 , In dictionary order str1 comes after str2

ਮਿਸਾਲ :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main(void)
{
 char str1[25],str2[25];
 int value;
 clrscr();
 printf ("\nEnter the first string:  ");
 gets (str1);
 printf("\nEnter the second string:  ");
 gets (str2);
 value=strcmp(str1,str2);
 if (value>0)
  printf("string %s comes after %s in dictionary order ",str1,str2);
 else if (value<0)
  printf("string %s comes after %s in dictionary order ",str1,str2);
 else
  printf("Both strings are Same\n");
}
```

ਪਰਿਣਾਮ (Output)

Enter the first string : Ram
Enter the second string : Gopal
Gopal comes before Ram in dictionary order.

strlwr () function

ਇਹ ਫੰਕਸ਼ਨ (function) ਇਕ ਸਟਰਿੰਗ (string) ਦੇ ਸਮੂਹ ਚਿੰਨ੍ਹਾਂ (characters) ਨੂੰ ਅਪਰਕੇਸ ਤੋਂ ਲੋਅਰਕੇਸ (upper case to lower case) ਵਿਚ ਬਦਲਦਾ ਹੈ। ਇਸ ਦੀ ਸਟਰਿੰਗ ਰਚਨਾ ਹੈ **strlwr (string);**

ਮਿਸਾਲ : 1. **strlwr("EXFORSYS");** EXFORSYS ਬਦਲ ਕੇ **exforsys** ਹੋ ਜਾਵੇਗਾ

2. **strlwr("Bangalore");** ਪਰਿਣਾਮੀ-ਸਟਰਿੰਗ (resultant string) ਹੋਵੇਗੀ **bangalore**

ਨਿਮਨਲਿਖਿਤ ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਨੂੰ ਸਵੀਕਾਰ ਅਤੇ ਪ੍ਰਦਰਸ਼ਨ ਕਰਦਾ ਹੈ। ਇਹ ਸਾਰੇ ਅੱਖਰਾਂ ਨੂੰ ਲੋਅਰ-ਕੇਸ ਤੋਂ ਅਪਰ-ਕੇਸ (lower case to upper case) ਵਿਚ ਬਦਲਦਾ ਹੈ ਅਤੇ ਪਰਿਣਾਮੀ ਸਟਰਿੰਗ (resultant string) ਪ੍ਰਦਰਸ਼ਨ ਕਰਦਾ ਹੈ।

ਮਿਸਾਲ :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main()
{
 char str[25];
 clrscr() ;
 printf("Enter the string\n");
 gets (str);
 printf("Input string is %s\n",str);
 printf("Resultant string is \n");
 puts(strlwr(str));
}
```

ਪਰਿਣਾਮ (output):

Enter the string
RAM
Input string is RAM
Resultant string is
ram

strrev( ) function

ਇਹ ਫੰਕਸ਼ਨ (function) ਸਟਰਿੰਗ (string) ਵਿਚ ਅੱਖਰਾਂ ਨੂੰ ਉਲਟਾਉਂਦਾ ਹੈ। ਇਸ ਦੀ ਵਾਕ ਰਚਨਾ ਹੈ :

strrev(string);

ਮਿਸਾਲ : strrev("program") ਇਹ "program" ਨੂੰ ਉਲਟ ਕੇ "margorp" ਬਣਾ ਦਿੰਦਾ ਹੈ।

ਇਹ ਫੰਕਸ਼ਨ (function) ਸਟਰਿੰਗ ਵੇਰੀਏਬਲਜ਼ (string variables) ਨੂੰ ਆਰਗੂਮੈਂਟ (argument) ਦੀ ਤੌਰ ਤੇ ਲਿਆਂਦਾ ਹੈ। ਇਹ ਸਟਰਿੰਗ (string) ਨੂੰ ਉਲਟ ਦੇਂਦਾ ਹੈ ਅਰਥਾਤ ਅੰਤਿਮ ਚਿੰਨ੍ਹ (characters) ਪਹਿਲਾ, ਅਤੇ ਦੂਜਾ ਅੰਤਿਮ, ਦੂਜਾ ਬਣ ਜਾਂਦਾ ਹੈ। ਇਸ ਦੀ ਵਰਤੋਂ ਲਈ ਆਉ ਇਕ ਸ਼ਬਦ ਤੇ ਗੌਰ ਕਰੀਏ ਜਿਹੜਾ ਉਲਟਾਉਣ ਤੇ ਉਹੀ ਸ਼ਬਦ ਬਣ ਜਾਂਦਾ ਹੈ। ਅਰਥਾਤ ਇਸ ਸ਼ਬਦ ਨੂੰ ਪਾਲੀਨਡਰੋਮ (Palindrome) ਕਿਹਾ ਜਾ ਸਕਦਾ ਹੈ ਬਸ਼ਰਤੇ ਕਿ ਇਹ ਉਲਟਾਉਣ ਤੋਂ ਬਾਅਦ ਵੀ ਇਹੀ ਰਹੇ। ਜਿਵੇਂ ਕਿ RADAR ਅਰਥਾਤ ਇਸਦਾ ਪਹਿਲਾ ਚਿੰਨ੍ਹ (character) ਅੰਤਿਮ ਚਿੰਨ੍ਹ (character) ਦੀ ਤਰ੍ਹਾਂ ਉਹੀ ਰਹਿੰਦਾ ਹੈ; ਦੂਜਾ ਚਿੰਨ੍ਹ (character) ਦੂਜੇ ਅੰਤਿਮ ਚਿੰਨ੍ਹ (character) ਦੀ ਤਰ੍ਹਾਂ ਉਹੀ ਰਹਿੰਦਾ ਹੈ। ਹੋਰ ਮਿਸਾਲਾਂ ਹਨ madam, Malayalam

ਮਿਸਾਲ :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main(void)
{
 char str1[25],str2[25];
 clrscr();
 printf ("\nEnter any string of length not more than 20:\n") ;
 gets(str1);
 strcpy(str2,str1);
 strrev(str2);
 if((strcmp (str1,str2)) ==0)
 printf("Word %s is a palindrome\n",str1);
 else
 printf("Word %s is not a palindrome\n",str1);
}
```

ਪਰਿਣਾਮ (Output):

Enter any string of length not more than 20:
adda
Word adda is a palindrome.

strupr( ) function

ਇਹ ਫੰਕਸ਼ਨ ਸਟਰਿੰਗ ਵਿਚ ਸਭ ਚਿੰਨ੍ਹਾਂ (characters) ਨੂੰ ਲੋਅਰ-ਕੇਸ ਤੋਂ ਅਪਰ-ਕੇਸ (lower case to upper case) ਵਿਚ ਬਦਲ ਦਿੰਦਾ ਹੈ। ਇਸ ਦੀ ਵਾਕ ਰਚਨਾ ਹੈ : strupr(string)

ਮਿਸਾਲ ਲਈ : strupr("govind") ਸਟਰਿੰਗ ਨੂੰ GOVIND ਵਿਚ ਬਦਲ ਦੇਵੇਗਾ।

ਮਿਸਾਲ : ਪ੍ਰੋਗਰਾਮ ਛੋਟੇ ਅੱਖਰਾਂ ਨੂੰ ਵੱਡੇ ਅੱਖਰਾਂ ਵਿਚ ਬਦਲਣ ਲਈ

```
/*Program to convert text in small letters to capital*/
#include<stdio.h>
#include<string.h>
void main()
{
 char str[20];
 int x,y;
  printf("Enter a Word- >");
  scanf("%s",str);
   y = strlen(str);
   printf("Before conversion the string is %s\n" , str);
   for (x=0;x<y;x++)
   {
     // In ASCII character set the difference
      // between "a" and "A" is 32, test for small alphabets
    if (str[x]>='a' && str[x]<='z')
    str[x]=str[x]-32;
    }
    printf("After the conversion the string is % s\n",str);
}
```

ਪਰਿਣਾਮ (Output)

Enter the Word -> ramgopal
Before conversion the string is ramgopal
After the conversion the string is RAMGOPAL

ਮਿਸਾਲ : ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ-ਫੰਕਸ਼ਨ (string functions) ਦੀ ਵਰਤੋਂ

/* Example program to use string functions to correct program to get output*/

```
#include<stdio.h>
#include<string.h>
void main()
{
 char s1[20],s2[20],s3[20];
 int x,l1,l2,l3;
 printf("\n Enter the strings constant \n" );
 scanf("%s %s",s1,s2);
 x=strcmp(s1,s2);
 if(x!=0)
 {
  printf("\nStrings are not equal\n");
  strcat(s1,s2);                         /* joining two strings*/
  }
  else
  printf("\nStrings are equal");          /* copying s1 to s3 */
  strcpy(s3,s1);
  l1=strlen(s1);
  l2=strlen(s2);
  l3=strlen(s3);
  printf("\ns1=%s\t length=%d characters\n",s1,l1);
  printf("\ns2=%s\t length=%d characters\n",s2,l2);
  printf("\ns3=%s\t length=%d characters\n",s3,l3);
}
```

ਪਰਿਣਾਮ **(Output):**

```
Enter two string constants
Ram gopal

strings are not equal
s1 = Ramgopal        length = 8 characters
s2 = Ram             length = 3 characters
s3 = Ramgopal        length = 8 characters

Enter two strings constants
Ram   Ram
Strings are equal

s1 = Ram       length  = 6 characters
s2 = Ram       length  = 6 characters
s3 = Ram       length  = 6 characters
```

ਪ੍ਰੋਗਰਾਮ ਪਰਿਭਾਸ਼ਾ 1: ਇਕ ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਫੰਕਸ਼ਨਜ਼ ਦੀ ਵਰਤੋ ਬਿਨਾਂ ਦੋ ਸਟਰਿੰਗ (strings) ਨੂੰ concatenate ਕਰਨ ਲਈ ਅਤੇ ਇਕ ਨਵੀਂ ਸਟਰਿੰਗ ਬਣਾਉਣ ਲਈ।
(A program to concatenate two strings and create a new string without using the string handling functions)

ਸਟਰਿੰਗ String Concatenation ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਦੂਜੀ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਤੇ ਜੋੜਨ ਵਾਲੀ ਪ੍ਰਕਿਆ ਹੁੰਦੀ ਹੈ। (String concatenation is a process of appending one string to the end of another string).

ਵਿਧੀ (Procedure)

1. ਦੋ ਸਟਰਿੰਗ (string) ਪੜ੍ਹੋ ਅਤੇ ਉਨ੍ਹਾਂ ਨੂੰ ਸਟਰਿੰਗ1 ਅਤੇ ਸਟਰਿੰਗ2 ਵਜੋਂ ਬੁਲਾਉ।
2. ਪਹਿਲੀ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਤਕ ਜਾਉ।
3. ਦੂਜੀ ਸਟਰਿੰਗ (string) ਨੂੰ ਪਹਿਲੀ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਨਾਲ ਜੋੜੋ।
4. ਪਰਿਣਾਮੀ ਸਟਰਿੰਗ (resultant string) ਦੇ ਅੰਤਿਮ ਚਿੰਨ੍ਹ (character) ਨੂੰ ਖਾਲੀ ਚਿੰਨ੍ਹ /null character ('\O') ਦੇ ਅੰਤਿਮ ਚਿੰਨ੍ਹ (character) ਨਾਲ ਨਿਰਧਾਰਿਤ ਕਰੋ।

ਮਿਸਾਲ :

```
/* Check and correct the program to get the Output shown below */
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main(void)
{
 char str1[20],str2[20],str3[50];
 int a,b,c;
 clrscr();
 printf("Enter the first string==");
scanf("%s",str1);
printf("Enter the second string==");
scanf("%s",str2);
a=strlen(str1);                    // To find the length of each string
b=strlen(str2);                    //concatenation begins
for(c=0;c<a;c++)
str3[c]=str1[c];
for(c=0;c<b;c++)
str3[a+c] = str2[c];               // end of string mark is placed
str3[a+b] ='\0';
printf("\n\ntwo strings are %s and %s and New string is %s \n",str1,str2,str3);
}
```

ਪਰਿਣਾਮ (Output)

Enter the first string == Ram
Enter the second string == Gopal
Two strings are Ram and Gopal and New string is RamGopal.

ਪ੍ਰੋਗਰਾਮ ਪਰਿਭਾਸ਼ਾ 2: ਇਕ ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਦੇ ਬਿਨਾਂ ਦੇ ਸਟਰਿੰਗਜ਼ ਦੀ ਤੁਲਨਾ ਲਈ (A program to compare two strings without using the string handling functions)

ਤੁੱਲਨਾ (Comparison): ਇਹ ਦੋ ਸਟਰਿੰਗਜ਼ ਦੀ ਤੁਲਨਾ ਦਾ ਪ੍ਰੋਗਰਾਮ ਹੈ ਉਹ ਚਾਹੇ ਬਰਾਬਰ ਹੋਣ ਜਾਂ ਨਾ। (This is a process of comparing two strings whether they are equal or not)

1. ਦੋ ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਪੜ੍ਹੋ ਅਤੇ ਉਨ੍ਹਾਂ ਨੂੰ ਸਟਰਿੰਗ1 ਅਤੇ ਸਟਰਿੰਗ2 ਵਜੋਂ ਬੁਲਾਓ।
2. ਹਰੇਕ ਸਟਰਿੰਗ (string) ਦੀ ਲੰਬਾਈ ਲਭੋ।
3. ਕਾਊਂਟਰ ਨੂੰ ਜ਼ੀਰੋ ਤੇ ਸੈੱਟ ਕਰੋ।
4. ਦੋਹਾਂ ਸਟਰਿੰਗਜ਼ (strings) ਦੇ ਪਹਿਲੇ ਚਿੰਨ੍ਹ (character) ਦੀ ਤੁਲਨਾ ਕਰੋ ਅਤੇ ਜੇਕਰ ਉਹ ਬਰਾਬਰ ਹਨ ਤਾਂ ਕਾਊਂਟਰ ਨੂੰ ਵਧਾਓ ਅਤੇ ਬਾਅਦ ਦੇ ਚਿੰਨ੍ਹਾਂ (characters) ਲਈ ਉਨੀ ਦੇਰ ਜਾਰੀ ਰੱਖੋ ਜਦੋਂ ਤਕ ਪੂਰਨ ਚਿੰਨ੍ਹ (character) ਅਲੱਗ ਨਹੀ ਹੁੰਦੇ ਜਾਂ ਸਟਰਿੰਗ (string) ਦੇ ਅੰਤ ਤਕ ਨਹੀਂ ਪਹੁੰਚਦੇ।
5. 0 ਨੂੰ ਵਾਪਸ ਭੇਜੋ ਜੇਕਰ ਉਹ ਇਕੋ ਜਿਹੇ ਹਨ। 1 ਜੇਕਰ ਸਟਰਿੰਗ1 ਦੂਜੀ ਨਾਲੋਂ ਵੱਡੀ ਹੈ ਅਤੇ -1 ਜੇਕਰ ਸਟਰਿੰਗ1 ਸਟਰਿੰਗ2 ਨਾਲੋਂ ਘੱਟ ਹੈ।

ਸੀ (C) ਵਲੋਂ ਦੋਹਾਂ ਸਟਰਿੰਗਜ ਦੀ ਸਿੱਧੀ ਤੁਲਨਾ ਨਹੀਂ ਕੀਤੀ ਜਾ ਸਕਦੀ। ਇਸ ਲਈ ਇਹ ਸਟੇਟਮੈਂਟ (statement) ਜਿਸ ਤਰ੍ਹਾਂ

if(name1 = =name2) ਜਾਂ if(name = ="ABC")

ਦੀ ਆਗਿਆ ਨਹੀ ਹੁੰਦੀ। ਇਸ ਲਈ ਦੋ ਸਟਰਿੰਗਜ (strings) ਨੂੰ ਜਾਂਚ ਕਰਨ ਲਈ ਉਨ੍ਹਾਂ ਦੀ ਤੁਲਨਾ ਕਰਨੀ ਜ਼ਰੂਰੀ ਹੁੰਦੀ ਹੈ। ਜੋ ਚਿੰਨ੍ਹ (character) ਤੋਂ ਚਿੰਨ੍ਹ (character) ਤੱਕ ਹੁੰਦੀ ਹੈ। ਤੁਲਨਾ ਉਨੀ ਦੇਰ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਜਦ ਤਕ ਅਸਮਾਨਤਾ ਗਾਇਬ ਨਹੀ ਹੁੰਦੀ ਜਾਂ ਇਕ ਸਟਰਿੰਗ (string) ਖਾਲੀ ਚਿੰਨ੍ਹ (character) ਵਿਚ ਖਤਮ ਹੋ ਜਾਂਦੀ ਹੈ ਜੋ ਵੀ ਪਹਿਲਾਂ ਵਾਪਰਦਾ ਹੈ।

ਮਿਸਾਲ :

```
#include<stdio.h>
#include<conio.h>
#include<string.h>
void main(void)
{
char str1[20],str2[20];
int a=0;
clrscr();
printf("Enter the first string ==");
scanf("%s",str1);
printf("Enter the second string ==");
scanf("%s",str2);
while (str1[a] == str2[a] &&  str1[a] !='\0' && str2 [a]!='\0')
a = a+1;
if (str1[a]  == '\0' &&  str2[a] == '\0')
printf("strings are equal\n");
else
printf("strings are not equal\n");
}
```

ਪਰਿਣਾਮ (Output)

```
Enter the first string = =   abcd
Enter the second string = = edf
Strings are not equal
```

ਪ੍ਰੋਗਰਾਮ ਪਰਿਭਾਸ਼ਾ 3: ਇਕ ਪ੍ਰੋਗਰਾਮ ਸਟਰਿੰਗ ਫੰਕਸ਼ਨਜ਼ ਦੀ ਵਰਤੋਂ ਬਿਨਾਂ ਦਿਤੇ ਗਏ text ਨੂੰ ਉਲਟਾਉਣ ਲਈ (A program to reverse a given text of string without using the string handling functions).

C ਗਰੰਟੀ ਦਿੰਦਾ ਹੈ ਕਿ && ਅਤੇ '||' ਦਾ ਮੁਲਾਂਕਣ ਖੱਬੇ ਤੋਂ ਸੱਜੇ ਹੁੰਦਾ ਹੈ। ਸੀ (C) ਦੀ ਇਕ ਖਾਮੀ ਇਹ ਹੈ ਕਿ **if** ਸਟੇਟਮੈਂਟਸ (statements) ਦਾ ਭਾਗ { } ਦੇ ਇਕ ਸੈੱਟ ਦੇ ਨੱਥੀ ਕਰਨ ਦੁਆਰਾ ਗੁੰਝਲਦਾਰ ਬਣ ਜਾਂਦਾ ਹੈ।

ਸਧਾਰਣ ਮਿਸਾਲ : ਮੰਨ ਲਓ ਕਿ ਅਸੀਂ ਇਹ ਯਕੀਨੀ ਬਣਾਉਣਾ ਚਾਹੁੰਦੇ ਹਾਂ ਕਿ **a** ਜੋ ਹੈ ਉਹ **b** ਨਾਲੋਂ ਵੱਡਾ ਹੁੰਦਾ ਹੈ, ਜੋ ਕਿ ਰੂਟੀਨ ਦਾ ਇਕ ਭਾਗ ਹੁੰਦਾ ਹੈ। **a** ਅਤੇ **b** ਦੀ ਅੰਤਰ ਤਬਦੀਲੀ ਲਈ ਸੀ (C) ਵਿਚ ਤਿੰਨ ਸਟੇਟਮੈਂਟ (statement) ਲਏ ਜਾਂਦੇ ਹਨ, ਜੋ ਕਿ { } ਨਾਲ ਦਰਜਾਬੰਦ ਕੀਤੇ ਜਾਂਦੇ ਹਨ।

```
if (a < b) {
t = a;
a = b;
b = t;
}
```

This problem is solved in the following using for loop, hint is given, get the given output

```
#include <stdio.h>
#include<conio.h>
#include<string.h>
#define max 80                      // To store the maximum length of a string
void main()
{
int i,j;
char temp;
char a[max];                        // array holding the string
clrscr();                           // library function which clear the screen
printf("\t Enter a line of text.\n");
gets(a);
for(i=0,j=strlen(a-2);i<=j;++i,--j)
{
    temp = a[i];
   a[i] =a[j];
   a[j] = temp;
}
printf("\n\t The reversed string is %s",a);
puts(a);
}
```

ਪਰਿਣਾਮ (Output)

Enter the line of Text: abc def
The reversed string is fed cba

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1: ਬਹੁਪੱਖੀ ਪਸੰਦ ਦੇ ਪ੍ਰਸ਼ਨ

1.1 ਖਾਲੀ ਸਟਰਿੰਗ (string) ਦੀ ਲੰਬਾਈ ਕੀ ਹੁੰਦੀ ਹੈ ?

a) 1　　b) 0　　c) 2　　d) 4

1.2 ਹਰ ਸਟਰਿੰਗ (string) ਦੀ ਸਮਾਪਤੀ .... ਨਾਲ ਹੋਣੀ ਚਾਹੀਦੀ ਹੈ ?

(ੲ) ਕੈਰਜ ਵਾਪਸੀ ਚਿੰਨ੍ਹ　(ਬੀ) ਨਵੀ ਰੇਖਾ ਚਿੰਨ੍ਹ　(ਸੀ) ਖਾਲੀ ਚਿੰਨ੍ਹ　(ਡੀ) ਲਾਈਨਫੀਡ ਚਿੰਨ੍ਹ।

1.3 ਸਟਰਿੰਗ (string) ਨੂੰ ਉਲਟਣ ਲਈ ਕਿਹੜੇ ਨਿਮਨਲਿਖਤ ਫੰਕਸ਼ਨਾਂ (function) ਦਾ ਇਸਤੇਮਾਲ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ?

a) strcpy　　b) strlen　　c) strcmp　　d) None of these

1.4 ਜੇਕਰ ਅਸੀ ਵਿਚਾਰ ਕਰੀਏ ਕਿ ਸਟਰਿੰਗ (string) ਦਾ ਐਲਾਨ char x[10] ਵਜੋਂ ਹੁੰਦਾ ਹੈ ਤਾਂ ਉਸ ਸਟਰਿੰਗ (string) ਦੀ ਲੰਬਾਈ ਕਿੰਨੀ ਹੁੰਦੀ ਹੈ ਜਿਹੜੀ x ਦੁਆਰਾ ਸਹੀ ਤੌਰ ਤੇ ਪ੍ਰਤੀਨਿਧ ਕੀਤੀ ਜਾ ਸਕਦੀ ਹੈ ?

a) 11　　b) 9　　c) 10　　d) all of these

1.5 ਨਿਮਨਲਿਖਤ ਵਿਚ ਕਿਹੜੇ ਇਨਪੁੱਟ ਫੰਕਸ਼ਨ (function) ਬਹੁਸ਼ਬਦੀ ਸਟਰਿੰਗ (string) ਦੇ ਲਈ ਨਹੀ ਵਰਤੇ ਜਾ ਸਕਦੇ ?

a)getch( )　　b) gets ( )　　c) scanf()　　d) None of these

1.6 ਇਨ੍ਹਾਂ ਵਿਚੋਂ ਕਿਹੜਾ ਸਹੀ ਹੈ, ਜੇਕਰ ਅਸੀ ਲਿਖਦੇ ਹਾਂ strcmp (s1, s2) ਕਿਹੜਾ ਮੁੱਲ ਵਾਪਸ ਹੰਦਾ ਹੈ।

a) <0 when s1<s2　　b) = 0 when s1 =s2　　c) > 0 when s1 > s2　　d) all of these

1.7 ਜੇਕਰ ਫੰਕਸ਼ਨ (function) strcat (s1, s2) ਲਾਗੂ ਹੁੰਦਾ ਹੈ ਤਦ concatenate ਜੋ ਵਾਪਸ ਕਰੇਗਾ।

a) s2 at the end of s2　**b)** s2 at the end of s1　**c)** s1 at the end of s2　**d)** s2 at the start of s1

1.8 ਫੰਕਸ਼ਨ (function) strcmp ('Abcd; ABCD) ਕੀ ਮੋੜੇਗਾ ?

a) -1　　b) 0　　c) 1　　d) None of these

1.9 ਸਟਰਿੰਗ (string) ਫੰਕਸ਼ਨਾਂ (functions) ਦੇ ਨਾਲ ਚਿੰਨ੍ਹ ਸਟਰਿੰਗ (string) ਤੇ ਕਿਹੜੇ ਨਿਮਨਲਿਖਤ ਕੰਮ ਨਹੀ ਕੀਤੇ ਜਾ ਸਕਦੇ ?

a) ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਜੋੜਣਾ　　b) ਇਕ ਸਟਰਿੰਗ (string) ਦੀ ਦੂਜੀ ਨਾਲ ਕਾਪੀ ਕਰਨੀ

c) ਬਰਾਬਰਤਾ ਲਈ ਸਟਰਿੰਗਜ਼ ਦਾ ਮੁਕਾਬਲਾ ਕਰਨਾ　　d) ਇਕ ਸਟਰਿੰਗ ਨੂੰ ਹਟਾਉਣਾ।

1.10 ਸਟਰਿੰਗ ਅੰਕਾਂ ਨੂੰ ਉਨ੍ਹਾਂ ਦੇ ਪੂਰਣ ਮੁੱਲ ਵਿਚ ਬਦਲਣ ਲਈ ਨਿਮਨਲਿਖਿਤ ਚੋਂ ਕਿਹੜਾ ਫੰਕਸ਼ਨ (function) ਸਹੀ ਹੈ ?

a) x =atoi(string)　　b) x = stoi(string)　　c) x = chartoint(string)　　d) all of these

ਪ੍ਰਸ਼ਨ 2: ਦੱਸੋ ਸਹੀ ਹੈ ਜਾਂ ਗਲਤ

2.1 ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਚਿੰਨ੍ਹਾਂ ਦੀ ਐਰੇ ਨਾਲ ਦਰਸਾਇਆ ਜਾ ਸਕਦਾ ਹੈ ?

2.2 ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਆਮ ਐਰੇ ਦੀ ਵਾਕ-ਰਚਨਾਂ ਵਾਂਗ ਹੀ ਕੇਵਲ ਸਟਰਿੰਗ (string) ਚਿੰਨ੍ਹਾਂ ਨਾਲ ਆਰੰਭ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ ?

2.3 strcopy; ਕਾਪੀ ਸਟਰਿੰਗਜ਼ (strings) ਵਿਚ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ?

2.4 ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਉਲਟਣ ਲਈ ਸਾਨੂੰ ਦੋ ਸਾਂਭ ਸੰਭਾਲ ਦੇ ਫੰਕਸ਼ਨ (function) ਜਿਵੇਂ strcpy( ) ਅਤੇ strcmp( ) ਚਾਹੀਦੇ ਹੁੰਦੇ ਹਨ ?

2.5 ਸਟਰਿੰਗ (string) ਲਾਈਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨ (function) ਦੀ ਵਰਤੋ ਲਈ ਆਪਣੇ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਹੈਡਰ ਫਾਈਲ string.h ਦੀ ਵਰਤੋ ਕਰਦੇ ਹਾਂ ?

ਪ੍ਰਸ਼ਨ 3: ਨਿਮਨਲਿਖਤ ਲਈ ਉਚਿਤ ਸ਼ਬਦ ਜਾਂ ਪਰਿਣਾਮ ਲਿਖੋ

3.1 ਸੀ (C) ਸਟਰਿੰਗ (string) ਡਾਟਾ ਟਾਈਪ ਦੀ ਸਹਾਇਤਾ ਨਹੀ ਕਰਦਾ ਭਾਵੇ ਕਿ ਸਟਰਿੰਗਜ਼ (strings) ਨੂੰ ਐਰੇ ਦੇ ਵਜੋਂ ਸੰਭਾਲਿਆ ਜਾ ਸਕਦਾ ਹੈ ?

3.2 strlen("India is a great") ਸਟਰਿੰਗ (string) ਦੀ ਲੰਬਾਈ ਕਿੰਨੀ ਵਾਪਸ ਕਰੇਗਾ ?

3.3 ਇਸ ਸਟਰਿੰਗ (string) ਆਰੰਭੀਕਰਨ (initiliazation) ਵਿਚ ਕੀ ਗਾਇਬ ਹੈ char name[8] = {'r', 'a', 'e', 't'}

3.4 ਇਕ ਖਾਲੀ ਸਟਰਿੰਗ (string) ਨੂੰ ਡਿਕਲੇਅਰ ਸਟੋਰ ਅਤੇ ਇਨੀਸ਼ਿਜਲਾਈਜੇਸ਼ਨ (initiliazation) ਲਈ 10 ਅੰਸ਼ ਚਿੰਨ੍ਹ ਨਾਮਕ ਐਰੇ x ਲਈ ਸਾਡੇ ਕੋਲ char x [10]=' '; ਹੁੰਦੇ ਹਨ। ਇਸ ਲਈ ਹੋਰ ਵਿਧੀ ਕੀ ਹੋਵੇਗੀ ?

3.5 ਇਕੋ ਸਮੇ ਵਿਚ ਸੰਪੂਰਨ ਸਟਰਿੰਗ (string) ਨੂੰ ਪੜ੍ਹਣ ਲਈ ਕਿਹੜਾ ਫੰਕਸ਼ਨ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ?

3.6 ਇਕੋ ਸਮੇ ਇਕ ਚਿੰਨ੍ਹ ਪੜ੍ਹਣ ਲਈ ਕਿਹੜਾ ਫੰਕਸ਼ਨ (function) ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ?

ਪ੍ਰਸ਼ਨ 4: ਸੰਖੇਪ ਉੱਤਰ ਪ੍ਰਕਾਰ ਦੇ ਪ੍ਰਸ਼ਨ

4.1 ਐਰੇ (array) char ਬਾਰੇ ਵਿਸ਼ੇਸ਼ ਕੀ ਹੁੰਦਾ ਹੈ ?
4.2 ਸਟਰਿੰਗ (string) ਕੀ ਹੁੰਦਾ ਹੈ ?
4.3 ਖਾਲੀ ਚਿੰਨ੍ਹ (Null character) ਦਾ ਕੀ ਮੰਤਵ ਹੁੰਦਾ ਹੈ ?
4.4 ਵਿਦਿਆਰਥੀ ਕਹੀ ਜਾਣ ਵਾਲੀ ਸਟਰਿੰਗ ਜਿਸ ਵਿਚ 50 ਚਿੰਨ੍ਹ (character) ਹੋਣ ਦੀ ਘੋਸ਼ਣਾ ਕਰੋ? ਇਸ ਘੋਸ਼ਣਾ ਵਿਚ ਸਟਰਿੰਗ (string) ਇਨੀਸ਼ਿਜਲਾਈਜੇਸ਼ਨ (iniliazation) ਦੇ ਦੋ ਢੰਗ ਦੱਸੋ ?
4.5 ਕੈਰ (char) ਐਰੇ (array) ਦੀ ਕੀ ਸੀਮਾ ਹੁੰਦੀ ਹੈ ?

ਪ੍ਰਸ਼ਨ 5: ਪ੍ਰਸ਼ਨਾਂ ਦੇ ਉੱਤਰ ਦਿਓ

5.1 ਕੀ ਸਟਰਿੰਗ (string) ਪੜ੍ਹਣ ਲਈ ਅਸੀ scanf() ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਦੇ ਹਾਂ ? ਜੇਕਰ ਨਹੀਂ ਤਾਂ ਕਾਰਣ ਦਸੋ ਜਦੋਂ scanf () ਨਾਲ ਇਕ ਸਟਰਿੰਗ (string) ਨੂੰ ਪੜ੍ਹਿਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਚਿੰਨ੍ਹ (character) '&' ਦੀ ਵਰਤੋਂ ਗਲਤ ਕਿਉਂ ਹੁੰਦੀ ਹੈ ?
5.2 **str** ਵਿਚ ਕੀ ਸਟੋਰ ਹੋਵੇਗਾ ਜੇਕਰ ਨਿਮਨਲਿਖਤ ਵਿਚੋਂ ਇਕ ਮੁੱਲ ਨੂੰ ਵੱਖਰੇ ਤੌਰ ਤੇ scanf() ਦੇ ਇਕ ਹੀ ਸੱਦੇ ਨਾਲ ਪੜ੍ਹਿਆ ਜਾਵੇ ਤਾਂ ? char str[10];
(a) cityscape (b) New Delhi (c) one or more blanks
5.3 **"A"** ਅਤੇ **'A'** ਵਿਚ ਕੀ ਅੰਤਰ ਹੁੰਦਾ ਹੈ ?
5.4 putchar ( ) ਅਤੇ puts ( ) ਵਿਚ ਕੀ ਅੰਤਰ ਹੁੰਦਾ ਹੈ ?
5.5 ਸਟਰਿੰਗ (string) ਡਾਟਾ ਦੇ I/O ਦੀ ਅਦਾਇਗੀ ਵਿਚ ਵਰਤੇ ਜਾਣ ਵਾਲੇ ਵੱਖ ਵੱਖ ਫੰਕਸ਼ਨਾਂ (functions) ਦੀ ਸੂਚੀ ਦਿਓ ?
5.6 ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਸਹੀ ਕਰਕੇ ਨਿਮਨਲਿਖਤ ਦਾ ਪਰਿਣਾਮ ਲਭੋ ?

```
#include "stdio.h"
#include "string.h"
main ( )
{
  char str[80] = " I like c" ;
  strcpy (str, "hello") ;
  printf(str) ; }
```

Ans: hello

5.7 ਨਿਮਨ ਲਿਖਤ ਨੂੰ ਲਾਗੂ ਕਰਕੇ x ਦਾ ਮੁੱਲ ਕੀ ਹੋਵੇਗਾ?

```
int x ,y =10 ;
char z = 'a' ;
x = y + z ;
```

Ans: 107

Answer to objective questions

Q	.1	.2	.3	.4	.5	.6	.7	.8	.9	.10
Q1	b	b	d	b	c	d	b	b	d	a
Q2	T	F	F	F	T					
Q3	character	16	'\0'	char x[10]={"0"} ;	gets, getc or getchar	getchar				

ਅਧਿਆਇ

3

ਯੂਜ਼ਰ ਡਿਫਾਇੰਡ ਫੰਕਸ਼ਨਜ਼
USER DEFINED FUNCTIONS

ਭੂਮਿਕਾ (Introduction)

ਜਿਵੇਂ ਅਸੀ ਆਪਣੇ ਰੋਜ਼ਾਨਾ ਜੀਵਨ ਵਿਚ ਦੇਖਦੇ ਹਾਂ ਸਾਨੂੰ ਵੱਖ-ਵੱਖ ਕੰਮਾਂ ਲਈ ਹੋਰਾਂ ਵਿਅਕਤੀਆਂ ਦੀ ਮਦਦ ਦੀ ਜ਼ਰੂਰਤ ਹੁੰਦੀ ਹੈ ਕਿਉਂ ਜੋ ਇਕ ਵਿਅਕਤੀ ਸਾਰੇ ਕੰਮ ਨਹੀਂ ਕਰ ਸਕਦਾ। ਇਸ ਲਈ ਉਸਨੂੰ ਵੱਖ-ਵੱਖ ਕੰਮਾਂ ਲਈ ਵੱਖ-ਵੱਖ ਵਿਅਕਤੀਆਂ ਦੀ ਲੋੜ ਹੁੰਦੀ ਹੈ। ਕੰਪਿਊਟਰ ਪ੍ਰੋਗਰਾਮ ਵੀ ਅਜਿਹੀ ਸਥਿਤੀ ਵਿਚ ਹੀ ਹੁੰਦਾ ਹੈ। ਇਹ ਵੀ ਸਾਰੇ ਕੰਮ ਆਪਣੇ ਆਪ ਨਹੀਂ ਸੰਭਾਲ ਸਕਦਾ। ਇਸ ਦੀ ਬਜਾਏ ਇਹ ਵਜੂਦ (entities) ਜਿਸ ਨੂੰ ਫੰਕਸ਼ਨ (function) ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਲਈ ਕੰਮ ਕਰਨ ਦੀ ਬੇਨਤੀ ਕਰਦਾ ਹੈ। ਸੀ (C) ਵਿਚ, ਫੰਕਸ਼ਨ, ਪ੍ਰੋਗਰਾਮ ਸਟੇਟਮੈਂਟਸ ਦੇ ਸਮੂਹ ਦੀ ਇਕਾਈ (unit) ਦਾ ਨਾਮ ਹੈ। ਇਕ ਇਕਾਈ ਮੇਨ ਪ੍ਰੋਗਰਾਮ ਦੇ ਹੋਰਾਂ ਭਾਗਾਂ ਤੋਂ ਮੰਗ ਕਰਦੀ ਹੈ। ਸੀ (C) ਦੇ ਨਿਰਮਿਤ ਬਲਾਕ ਅਤੇ ਉਹ ਸਥਾਨ ਜਿੱਥੇ ਸਮੂਹ ਪ੍ਰੋਗਰਾਮਾਂ ਦੀਆਂ ਗਤੀਵਿਧੀਆਂ ਵਾਪਰਦੀਆਂ ਹਨ, ਫੰਕਸ਼ਨਜ਼ ਹੁੰਦੇ ਹਨ।

ਇਹ ਫੰਕਸ਼ਨਜ਼, ਲਾਇਬ੍ਰੇਰੀ-ਫੰਕਸ਼ਨ (library function) ਜਾਂ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨ (user defind function) ਹੋ ਸਕਦੇ ਹਨ। ਇਹ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨ ਵੱਡੇ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਅਨੇਕਾਂ ਛੋਟੇ ਛੋਟੇ ਮਾਡਯੂਲਸ (modules) ਵਿਚ ਵੰਡਣ ਦੀ ਆਗਿਆ ਦਿੰਦੇ ਹਨ, ਇਨ੍ਹਾਂ ਵਿਚੋਂ ਹਰੇਕ ਮਾਡਯੂਲ (module) ਦਾ ਕੋਈ ਵਿਸ਼ੇਸ਼ ਕੰਮ ਹੁੰਦਾ ਹੈ। ਸੀ (C) ਵਿਚ ਫੰਕਸ਼ਨਜ਼ ਨੂੰ ਦੋ ਸ਼੍ਰੇਣੀਆਂ ਵਿਚ ਸ਼੍ਰੇਣੀਬੱਧ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

ਬਿਲਟ-ਇੰਨ/ਲਾਇਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨਜ਼ (Built in or Library functions)

ਕੁਝ ਅੱਪਰੇਸ਼ਨ ਜਿਵੇਂ ਇਕ ਅੰਕ ਦਾ ਵਰਗ-ਮੁੱਲ (square root), ਇਕ ਨੰਬਰ ਦਾ ਸਾਈਨ-ਮੁੱਲ (sine value) ਬਾਰ-ਬਾਰ (frequently) ਬਹੁਤ ਸਾਰੇ ਪ੍ਰੋਗਰਾਮਰਾਂ ਵੱਲੋਂ ਆਪਣੇ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ। ਇਕ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਇਕ ਅੰਕ ਦਾ ਵਰਗ-ਮੁੱਲ (square root) ਲੱਭਣ ਜਾਂ ਨੰਬਰ ਦੇ ਸਾਈਨ-ਮੁੱਲ ਲਈ ਪ੍ਰੋਗਰਾਮ ਹਰ ਵਾਰ ਲਿਖਣਾ ਗੈਰ ਜ਼ਰੂਰੀ ਅਤੇ ਥਕਾਵਟ ਵਾਲਾ ਕੰਮ ਹੁੰਦਾ ਹੈ। ਅਜਿਹੇ ਫੰਕਸ਼ਨ (function) ਪ੍ਰੋਗਰਾਮ ਅਧੀਨ ਹੁੰਦੇ ਹਨ ਅਤੇ ਸੀ (C) ਲਾਇਬ੍ਰੇਰੀ ਵਿਚ ਸੇਵ ਕੀਤੇ ਜਾਂਦੇ ਹਨ ਤਾਂ ਜੋ ਉਹ ਕਿਸੇ ਵੀ ਪ੍ਰੋਗਰਾਮ ਰਾਹੀਂ ਬੁਲਾਏ ਜਾ ਸਕਣ। ਮਿਸਾਲ ਲਈ: scanf(), printf() ਅਤੇ ਸਮੂਹ ਸਟਰਿੰਗ (string) ਵਰਤੋਂ ਵਾਲੇ ਫੰਕਸ਼ਨ (function), ਲਾਇਬ੍ਰੇਰੀ-ਫੰਕਸ਼ਨ (library function) ਹੁੰਦੇ ਹਨ ਅਤੇ ਇਨ੍ਹਾਂ ਨੂੰ ਲਾਇਬ੍ਰੇਰੀ-ਫੰਕਸ਼ਨ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਸੀ (C) ਵਿਚ ਅਰਥਮੈਟਿਕਲ ਫੰਕਸ਼ਨਜ਼ (airthmatical functions) ਨੂੰ ਬੁਲਾਉਣ ਲਈ ਪ੍ਰੀਪ੍ਰੋਸੈਸਰ ਡਾਇਰੈਕਟਿਵ # include<math.h> ਦੀ ਵਰਤੋਂ ਨਾਲ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ। ਅਤੇ ਕੁਝ ਵਿਸ਼ੇਸ਼ I/O ਫੰਕਸ਼ਨ (function) ਜਿਵੇਂ #include<stdio.h> ਨੂੰ ਬੁਲਾ ਕੇ ਹੁੰਦੇ ਹਨ।

ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨਜ਼ (User-defined functions)

ਲਾਇਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨ ਜਿਹੜੇ ਸੀ (C) ਵਿਚ ਨਿਰਮਿਤ (built) ਹੁੰਦੇ ਹਨ ਤੋਂ ਇਲਾਵਾ ਯੂਜ਼ਰ ਆਪਣੇ ਪ੍ਰੋਗਰਾਮਾਂ ਨਾਲ ਸੰਬੰਧਿਤ ਕੰਮ ਨੂੰ ਕਰਨ ਲਈ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨਜ਼ (user defined functions) ਵੀ ਇਸਤੇਮਾਲ ਕਰ ਸਕਦੇ ਹਨ। ਇਹ ਪ੍ਰੋਗਰਾਮ, ਯੂਜ਼ਰ ਵੱਲੋਂ ਲਿਖੇ ਹੋਣੇ ਚਾਹੀਦੇ ਹਨ ਤਾਂ ਜੋ ਕੋਈ ਵੀ ਬੁਲਾਵਾ ਇਸਦਾ ਹਵਾਲਾ ਦੇ ਸਕੇ। ਇਨ੍ਹਾਂ ਫੰਕਸ਼ਨਜ਼ ਨੂੰ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨਜ਼ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਵਿਚ main().

ਇਨ੍ਹਾਂ ਦੋ ਪ੍ਰਕਾਰ ਦੇ ਫੰਕਸ਼ਨਜ਼ ਵਿਚ ਮੇਨ ਅੰਤਰ ਇਹ ਹੈ ਕਿ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨ ਪ੍ਰੋਗਰਾਮ ਲਿਖਣ ਦੇ ਸਮੇਂ ਯੂਜ਼ਰ ਵੱਲੋਂ ਵਿਕਸਤ ਕੀਤੇ ਜਾਂਦੇ ਹਨ ਜਦਕਿ ਲਾਇਬ੍ਰੇਰੀ-ਫੰਕਸ਼ਨ ਸਾਡੇ ਵੱਲੋਂ ਲਿਖੇ ਜਾਣ ਦੀ ਲੋੜ ਨਹੀਂ ਹੁੰਦੀ। ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨ ਬਾਅਦ ਵਿਚ ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਫੰਕਸ਼ਨ ਦਾ ਭਾਗ ਬਣ ਜਾਂਦੇ ਹਨ।

ਫੰਕਸ਼ਨਜ਼ ਦੀ ਜ਼ਰੂਰਤ (Need of functions)

ਅਸੀਂ ਜਾਣਦੇ ਹਾਂ ਕਿ ਹਰ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਇਕ ਮੇਨ-ਫੰਕਸ਼ਨ (main function) ਹੁੰਦਾ ਹੈ ਜੋ ਇਹ ਸੰਕੇਤ ਦਿੰਦਾ ਹੈ ਕਿ ਪ੍ਰੋਗਰਾਮ ਨੇ ਆਪਣੇ ਪ੍ਰੋਗਰਾਮ ਦਾ ਲਾਗੂਕਰਨ (execution) ਕਿਥੋਂ ਸ਼ੁਰੂ ਕੀਤਾ ਹੈ। ਜਦ ਕਿ ਮੇਨ-ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਨਾਲ ਕਿਸੇ ਵੀ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਕੋਡ ਕਰਨਾ ਸੰਭਵ ਹੁੰਦਾ ਹੈ। ਜੇਕਰ ਸਾਰਾ ਕੋਡ ਮੇਨ ਫੰਕਸ਼ਨ (main function) ਵਿਚ ਲਿਖਿਆ ਜਾਵੇ ਤਾਂ ਬਹੁਤ ਸਾਰੀਆਂ ਮੁਸ਼ਕਿਲਾਂ ਪੈਦਾ ਹੋ ਜਾਂਦੀਆਂ ਹਨ ਜਿਵੇਂ ਕਿ ਪ੍ਰੋਗਰਾਮ ਬਹੁਤ ਵੱਡਾ ਅਤੇ ਪੇਚੀਦਾ ਹੋ ਸਕਦਾ ਹੈ ਅਤੇ ਸਿੱਟੇ ਵਜੋਂ ਡਿਬਗਿੰਗ (debugging), ਟੈਸਟਿੰਗ (testing) ਅਤੇ ਸਾਂਭ ਸੰਭਾਲ ਦਾ ਕੰਮ ਬਹੁਤ ਮੁਸ਼ਕਿਲ ਹੋ ਜਾਂਦਾ ਹੈ। ਜੇਕਰ ਪ੍ਰੋਗਰਾਮ ਉਪਭਾਗਾਂ ਵਿਚ ਵੰਡਿਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਹਰੇਕ ਭਾਗ ਸੁਤੰਤਰਤਾ ਦੇ ਨਾਲ ਕੋਡ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ। ਅਤੇ ਬਾਅਦ ਵਿਚ ਇਕ ਇਕਹਿਰੀ ਇਕਾਈ ਵਿਚ ਮਿਲਾਇਆ ਜਾ ਸਕਦਾ ਹੈ। ਇਹ ਉਪ ਪ੍ਰੋਗਰਾਮ ਜਿਨ੍ਹਾਂ ਨੂੰ 'ਫੰਕਸ਼ਨ' (function) ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਸਮਝਣ, ਡਿਬੱਗ (debug) ਅਤੇ ਪਰਖ ਕਰਨ ਵਿਚ ਬਹੁਤ ਆਸਾਨ ਹੁੰਦੇ ਹਨ। ਵੱਡੇ ਮਾਡਯੂਲ (module) ਦਾ ਉਪ-ਵਰਗੀਕਰਨ ਨਿਮਨਲਿਖਿਤ ਲਾਭ ਪਹੁੰਚਾਉਂਦਾ ਹੈ ।

1. ਇਹ **ਟੌਪ-ਡਾਊਨ ਮਾਡਯੂਲਰ ਪ੍ਰੋਗਰਾਮਿੰਗ (top-down modular programming)** ਨੂੰ ਉਤਸ਼ਾਹਿਤ ਕਰਦਾ ਹੈ। ਜਿਥੇ ਸਮੁੱਚੀ ਸਮੱਸਿਆ ਦੀ ਹਾਈ ਲੈਵਲ (high level) ਦੇ ਲੌਜਿਕ (logic) ਦੀ ਸਮੱਸਿਆ ਪਹਿਲਾਂ ਹੱਲ ਹੁੰਦੀ ਹੈ ਜਦ ਕਿ ਹਰ ਇਕ ਲੋਅਰ ਲੈਵਲ (lower level) ਫੰਕਸ਼ਨ ਦੇ ਵੇਰਵੇ ਬਾਅਦ ਵਿਚ ਹੱਲ ਹੁੰਦੇ ਹਨ।
2. ਇਹ **ਪੁਨਰ-ਵਰਤੋਂ (reusability)** ਨੂੰ ਵੀ ਪ੍ਰੋਤਸਾਹਿਤ ਕਰਦੀ ਹੈ। ਕੁਝ ਭਾਗਾਂ ਨੂੰ ਜਦੋਂ ਲੋੜ ਹੋਵੇ ਬੁਲਾਇਆ ਜਾ ਸਕਦਾ ਹੈ।
3. **ਡਿਬਗਿੰਗ ਆਸਾਨ ਹੁੰਦੀ ਹੈ (Debugging is easier),** ਕਿਉਂ ਜੋ ਹਰੇਕ ਮਾਡਯੂਲ (module) ਛੋਟਾ ਅਤੇ ਸਪੱਸ਼ਟ ਹੁੰਦਾ ਹੈ, ਯੂਜ਼ਰ ਵੱਲੋਂ ਗਲਤੀਆਂ ਲੱਭਣੀਆਂ ਅਤੇ ਉਨ੍ਹਾਂ ਨੂੰ ਸਹੀ ਕਰਨਾ ਆਸਾਨ ਹੋ ਜਾਂਦਾ ਹੈ।
4. **ਬਿਲਡ ਲਾਇਬ੍ਰੇਰੀ (Build library):** ਇਹ ਬੇਹੱਦ ਸਾਂਝੀ ਵਰਤੋਂ ਵਾਲੇ ਉਪ-ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਇਕ ਲਾਇਬ੍ਰੇਰੀ ਬਣਾਉਣ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ। ਇਹ ਟਾਈਮ ਅਤੇ ਸਥਾਨ ਦੀ ਪੇਚੀਦਗੀ ਘਟਾਉਂਦਾ ਹੈ।

ਜਿਵੇਂ ਪਹਿਲਾਂ ਕਿਹਾ ਜਾ ਚੁਕਿਆ ਹੈ ਕਿ ਕੁਝ ਫੰਕਸ਼ਨ ਇਕ ਪ੍ਰੋਗਰਾਮ ਰਾਹੀਂ ਬਹੁਤ ਸਾਰੇ ਬਿੰਦੂਆਂ ਤੇ ਦੋਹਰਾਏ ਜਾਂਦੇ ਹਨ। ਇਥੇ ਇਹ ਫੰਕਸ਼ਨ ਦੁਬਾਰਾ ਬਿਲਡ (build) ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ ਜਿਹੜੇ ਲੋੜ ਅਨੁਸਾਰ ਬੁਲਾਏ ਜਾ ਸਕਦੇ ਹਨ। ਇਕ ਵਾਰ ਜਦੋਂ ਫੰਕਸ਼ਨ ਆਪਣੀ ਲੋੜੀਂਦੀ ਕਾਰਵਾਈ ਕਰ ਲੈਂਦਾ ਹੈ ਤਾਂ ਕੰਟਰੋਲ ਉਸ ਪੁਆਇੰਟ ਤੇ ਵਾਪਸ ਮੁੜ ਜਾਵੇਗਾ, ਜਿਥੋਂ ਫੰਕਸ਼ਨ ਲਾਗੂ ਕੀਤਾ ਗਿਆ ਸੀ।

ਆਮ ਤੌਰ ਤੇ ਫੰਕਸ਼ਨ called ਪ੍ਰੋਗਰਾਮ ਵੱਲੋਂ ਭੇਜੀ ਗਈ ਸੂਚਨਾ ਦੀ ਪ੍ਰੋਸੈਸਿੰਗ ਕਰਦਾ ਹੈ ਅਤੇ ਇਕਹਿਰੇ ਮੁੱਲ ਨੂੰ ਵਾਪਿਸ ਕਰਦਾ ਹੈ।

ਇਕ ਫੰਕਸ਼ਨ (function) ਦੀ ਵਰਤੋਂ ਲਈ ਪ੍ਰੋਗਰਾਮਰ ਨੂੰ ਨਿਮਨਲਿਖਤ ਕੰਮ ਕਰਨੇ ਚਾਹੀਦੇ ਹਨ:

- **ਫੰਕਸ਼ਨ ਡਿਫਾਈਨ ਕਰਨਾ** **(Define the function)**
- **ਫੰਕਸ਼ਨ ਡਿਕਲੇਅਰ ਕਰਨਾ** **(Declare the function)**
- **ਮੇਨ-ਕੋਡ ਵਿਚ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ** **(Use the function in the main code)**

ਆਰਗੂਮੈਂਟਸ ਅਤੇ ਪੈਰਾਮੀਟਰਸ (Arguments and Parameters)

ਇਹ ਵੇਰੀਏਬਲਸ ਹਨ ਜੋ ਫੰਕਸ਼ਨ ਵਿਚ ਵਰਤੇ ਜਾਂਦੇ ਹਨ। ਫੰਕਸ਼ਨ ਹਵਾਲਿਆਂ (reference) ਵਿਚ ਵੇਰਿਏਬਲ ਦੀ ਵਰਤੋਂ ਨੂੰ ਆਰਗੂਮੈਂਟਸ (arguments) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਇਨ੍ਹਾਂ ਨੂੰ ਪੈਰੇਨਥਿਸਿਸ (parenthesis) ਵਿਚ ਲਿਖਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਫੰਕਸ਼ਨ ਦੇ ਨਾਮ ਦੇ ਪਿੱਛੇ ਹੋਣੇ ਚਾਹੀਦੇ ਹਨ। ਇਨ੍ਹਾਂ ਨੂੰ ਅਸਲ (actual) ਪੈਰਾਮੀਟਰਸ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਕਿਉਂ ਜੋ ਇਹ ਮੇਨ ਫੰਕਸ਼ਨ ਜਾਂ calling ਫੰਕਸ਼ਨ ਵਿਚ ਪ੍ਰਵਾਨ ਕੀਤੇ ਜਾਂਦੇ ਹਨ।

ਵੇਰੀਏਬਲਸ ਜੋ ਫੰਕਸ਼ਨ ਪਰਿਭਾਸ਼ਾ (function definition) ਵਿਚ ਵਰਤੇ ਜਾਂਦੇ ਹਨ ਨੂੰ ਪੈਰਾਮੀਟਰਸ (parameters) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਇਨ੍ਹਾਂ ਨੂੰ ਰਸਮੀ-ਪੈਰਾਮੀਟਰਸ (formal parameters) ਵੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਕਿਉਂ ਜੋ ਉਹ ਪ੍ਰਵਾਨਤ ਮੁੱਲ (accepted value) ਨਹੀਂ ਹੁੰਦੇ। ਇਹ calling ਫੰਕਸ਼ਨ ਤੋਂ ਮੁੱਲ ਪ੍ਰਾਪਤ ਕਰਦੇ ਹਨ। ਫੰਕਸ਼ਨ-ਪਰਿਭਾਸ਼ਾ (function definition) ਦੇ ਵਿਚ, ਪੈਰਾਮੀਟਰਸ ਪੈਰੇਨਥਿਸਿਸ ਵਿਚ ਲਿਖੇ ਜਾਂਦੇ ਹਨ ਅਤੇ ਫੰਕਸ਼ਨ ਦੇ ਨਾਮ ਦੇ ਪਿੱਛੇ ਹੋਣੇ ਚਾਹੀਦੇ ਹਨ।

ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਹਮੇਸ਼ਾ ਆਪਣਾ ਲਾਗੂਕਰਨ (execution) ਮੇਨ ਫੰਕਸ਼ਨ ਵੱਲੋਂ ਸ਼ੁਰੂ ਕਰਦੇ ਹਨ ਇਸ ਲਈ ਅਸੀਂ ਵੀ ਉਥੋਂ ਹੀ ਸ਼ੁਰੂ ਕਰਾਂਗੇ ।

ਅਸੀਂ ਦੇਖ ਸਕਦੇ ਹਾਂ ਕਿ ਕਿਵੇਂ ਮੇਨ ਫੰਕਸ਼ਨ, ਵੇਰੀਏਬਲ **z** (ਜਿਹੜਾ int ਟਾਈਪ ਹੈ) ਦੀ ਡਿਕਲੇਰੇਸ਼ਨ (declaration) ਨਾਲ ਸ਼ੁਰੂ ਹੁੰਦਾ ਹੈ। ਇਸ ਉਪਰੰਤ ਅਸੀਂ **"addition"** ਨਾਮ ਦੇ ਫੰਕਸ਼ਨ ਨੂੰ call ਕਰਾਂਗੇ। ਅਸੀਂ ਧਿਆਨ ਨਾਲ ਵੇਖੀਏ ਤਾਂ ਇਹ ਦੇਖਣ ਦੇ ਕਾਬਿਲ ਹੋ ਜਾਵਾਂਗੇ ਕਿ ਫੰਕਸ਼ਨ ਦੀ call ਦੀ ਸਟਰਕਚਰ (structure) ਅਤੇ ਫੰਕਸ਼ਨ ਦੀ ਡਿਕਲੇਰੇਸ਼ਨ (declaration) ਦੀਆਂ ਕਈ ਕੋਡ ਲਾਈਨਾਂ ਵਿਚ ਕਿੰਨੀ ਸਮਾਨਤਾ ਹੁੰਦੀ ਹੈ।

ਪੈਰਾਮੀਟਰਸ ਅਤੇ ਆਰਗੂਮੈਂਟਸ (Parameters & arguments) ਵਿਚ ਸਪਸ਼ਟ ਅਦਾਨ-ਪ੍ਰਦਾਨ ਹੈ। ਮੇਨ ਫੰਕਸ਼ਨ ਦੇ ਵਿਚ ਅਸੀਂ ਦੋ ਮੁੱਲਾਂ (5 ਅਤੇ 3) ਨੂੰ ਭੇਜਣ ਲਈ addition function ਨੂੰ ਬੁਲਾਉਂਦੇ ਹਾਂ ਜਿਹੜੇ **int a** ਅਤੇ **int b ਪੈਰਾਮੀਟਰਸ** ਟਾਈਪ ਦੇ ਹਨ ਅਤੇ ਫੰਕਸ਼ਨ "addition" ਲਈ ਡਿਕਲੇਅਰ (declare) ਕੀਤੇ ਗਏ ਹਨ।

```
int addition (int a, int b)
                  ↑      ↑
z = addition (    5  ,   3  );
```

ਮੇਨ ਫੰਕਸ਼ਨ ਵਿਚ ਉਹ ਸਥਾਨ ਜਿੱਥੇ ਫੰਕਸ਼ਨ ਨੂੰ call ਕੀਤਾ ਗਿਆ ਹੈ, ਕੰਟਰੋਲ ਮੇਨ ਫੰਕਸ਼ਨ ਤੋਂ ਹਟਾ ਕੇ ਫੰਕਸ਼ਨ addition ਨੂੰ ਭੇਜਿਆਂ ਜਾਂਦਾ ਹੈ। ਫੰਕਸ਼ਨ call ਵਿਚ ਭੇਜਿਆਂ ਗਈਆਂ ਆਰਗੂਮੈਂਟਸ (arguments) 5 ਅਤੇ 3 ਲੋਕਲ ਵੇਰਿਏਬਲਸ int a ਅਤੇ int b ਵਿਚ ਕਾਪੀ ਹੋਣਗੀਆਂ।

ਮਿਸਾਲ :

```
//function example
#include <stdio.h>

int addition (int a, int b)
{
  int r;
  r = a + b;
  return (r);
}

int main ()
{
  int z;
  z = addition (5,3);
  printf("The result is %d", z);
  return 0;
}
```

Output:
The result is 8

ਫੰਕਸ਼ਨ addition ਇਕ ਹੋਰ ਲੋਕਲ ਵੇਰਿਏਬਲਸ **(int r)** ਨੂੰ ਡਿਕਲੇਅਰ (declare) ਕਰਦਾ ਹੈ ਅਤੇ ਪ੍ਰਗਟਾਓ (expression) **r = a + b** ਦੇ ਸਾਧਨਾਂ ਰਾਹੀਂ **a** ਜਮ੍ਹਾਂ **b** ਦੇ ਪਰਿਣਾਮ **r** ਨੂੰ ਅਸਾਇੰਡ (assigned) ਕਰਦਾ ਹੈ। ਕਿਉਂ ਜੋ ਅਸਲ ਪੈਰਾਮੀਟਰਸ ਜੋ **a** ਅਤੇ **b** ਲਈ ਕ੍ਰਮਵਾਰ 5 ਅਤੇ 3 ਹਨ, ਪਰਿਣਾਮ 8 ਹੈ।

```
int addition (int a, int b)
  ↓
z = addition (  5  ,  3  );
```

ਕੋਡ ਲਾਈਨ return(r)

ਇਹ ਫੰਕਸ਼ਨ addition ਨੂੰ ਮੁਕੰਮਲ ਕਰਦਾ ਹੈ ਅਤੇ ਪਹਿਲਾਂ ਬੁਲਾਏ ਗਏ ਕੰਟਰੋਲ ਨੂੰ ਫੰਕਸ਼ਨ ਵੱਲ ਵਾਪਸ ਭੇਜਦਾ ਹੈ (ਇਸ ਕੇਸ ਵਿਚ, main)।

ਇਸ ਮੌਕੇ ਤੇ ਪ੍ਰੋਗਰਾਮ ਪਹਿਲਾਂ ਵਾਲੀ ਸਥਿਤੀ ਤੇ ਪਹੁੰਚ ਜਾਂਦਾ ਹੈ ਜਿਥੋਂ ਇਹ ਫੰਕਸ਼ਨ "addition" ਦੀ call ਹੋਣ ਤੇ ਰੋਕਿਆ ਗਿਆ ਸੀ।

ਪਰੰਤੂ ਇਸ ਤੋਂ ਇਲਾਵਾ, ਕਿਉਂ ਜੋ ਫੰਕਸ਼ਨ addition ਵਿਚ return ਸਟੇਟਮੈਂਟ ਇਕ ਮੁੱਲ ਨੂੰ ਨਿਸ਼ਚਿਤ ਕਰਦਾ ਹੈ, ਵੇਰੀਏਬਲਸ z ਦੀ ਵਿਸ਼ਾ ਸੂਚੀ r **[return (r)]** ਜਿਸ ਦਾ ਇਸ ਮੌਕੇ ਤੇ ਮੁੱਲ 8 ਹੈ, ਇਹ ਮੁੱਲ, ਫੰਕਸ਼ਨ call ਤੇ ਮੁਲਾਂਕਣ ਦਾ ਮੁੱਲ ਬਣ ਜਾਂਦਾ ਹੈ। ਇਸ ਲਈ ਵੇਰੀਏਬਲ **z** ਦਾ ਮੁੱਲ ਫੰਕਸ਼ਨ addition (5, 3) ਦੁਆਰਾ ਵਾਪਸ ਕੀਤੀ ਗਈ ਕੀਮਤ ਹੋਵੇਗੀ ਜੋ ਕਿ 8 ਹੈ

ਮਲਟੀ-ਫੰਕਸ਼ਨ ਪ੍ਰੋਗਰਾਮ (Multi-Function Program)

ਇਕ ਫੰਕਸ਼ਨ (function) ਨੂੰ ਸਧਾਰਣ ਸ਼ਬਦਾਂ ਵਿਚ ਇਕ "Black box" ਵਜੋਂ ਦੇਖਿਆ ਜਾ ਸਕਦਾ ਹੈ ਜਿਹੜਾ ਕੁਝ ਮੁੱਲ (ਜੇਕਰ ਚਾਹੀਦੇ ਹੋਣ) ਲੈਂਦਾ ਹੈ ਅਤੇ ਕੁਝ ਆਊਟਪੁਟ ਦਿੰਦਾ ਹੈ। ਫੰਕਸ਼ਨ ਦੇ ਅੰਦਰੂਨੀ ਵੇਰਵੇ ਬਾਕੀ ਦੇ ਪ੍ਰੋਗਰਾਮ ਤੋਂ ਛੁਪੇ ਹੋਏ ਹੁੰਦੇ ਹਨ। ਇਨ੍ਹਾਂ ਇਕੱਠੇ ਕੀਤੇ "Black boxes" ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਸਾਰੇ ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਡਿਜ਼ਾਈਨ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ।

(Note: ਪ੍ਰੋਗਰਾਮ ਦੀ ਅੰਦਰੂਨੀ ਛੁਪੀ ਹੋਈ ਪ੍ਰੋਸੈਸਿੰਗ ਨੂੰ "Black box" ਕਹਿੰਦੇ ਹਨ)

ਹੇਠਾਂ ਦਿੱਤੀ ਮਿਸਾਲ ਤੇ ਵਿਚਾਰ ਕਰੋ :

```
displayname()
{
printf("\nResult of displayname function call");
}
```

Above function can be used as follows:

```
void main()
{
printf("\nThis demonstrates user-defined functions");
displayname();
}
```

Output of this program is:

This demonstrates user-defined functions
Result of display name function call

ਮੇਨ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਪਹਿਲਾਂ printf ਸਟੇਟਮੈਂਟ ਲਾਗੂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਇਸ ਤੋਂ ਬਾਅਦ ਯੂਜ਼ਰ-ਡਿਫਾਈਂਡ-ਫੰਕਸ਼ਨ displayname() ਨੂੰ call ਕੀਤਾ ਜਾਂਦਾ ਹੈ, ਜਿਸਦੇ ਪਰਿਣਾਮ, ਫੰਕਸ਼ਨ ਦੀ ਬਾਡੀ ਦੇ ਵਿਚ ਦੀ ਸਟੇਟਮੈਂਟਸ ਲਾਗੂ ਹੁੰਦੀ ਹੈ।

ਫੰਕਸ਼ਨ ਦੀ ਬਣਤਰ (form of function)

ਸੀ (C) ਭਾਸ਼ਾ ਵਿਚ ਆਮ ਤੌਰ ਤੇ ਫੰਕਸ਼ਨ ਦੀ ਬਣਤਰ ਨਿਮਨਲਿਖਤ ਅਨੁਸਾਰ ਹੁੰਦੀ ਹੈ :

```
function-name(argument list)
argument declaration;
{
 local variable declarations;
 executable statement1;
 executable statement2;
 ----------
 ----------
 return (expression);
}
```

ਫੰਕਸ਼ਨ ਦੇ ਸਾਰੇ ਭਾਗ (parts) ਜ਼ਰੂਰੀ ਨਹੀਂ ਹਨ ਅਰਥਾਤ ਕੁਝ ਗੈਰ-ਹਾਜ਼ਿਰ ਵੀ ਹੋ ਸਕਦੇ ਹਨ। ਮਿਸਾਲ ਲਈ, ਆਰਗੂਮੈਂਟ (argument) ਸੂਚੀ ਅਤੇ ਇਸਦੀ ਸੰਬੰਧਿਤ ਆਰਗੂਮੈਂਟ-ਡਿਕਲੇਅਰੇਸ਼ਨ ਭਾਗ (argument declaration parts) ਵਿਕਲਪੀ ਹੁੰਦੇ ਹਨ। ਲੋਕਲ ਵੇਰੀਏਬਲਸ ਦੀ ਡਿਕਲੇਅਰੇਸ਼ਨ ਦੀ ਲੋੜ ਕੇਵਲ ਉਦੋਂ ਹੀ ਹੁੰਦੀ ਹੈ ਜਦੋਂ ਇਹ ਫੰਕਸ਼ਨ ਵਿਚ ਵਰਤੇ ਜਾਂਦੇ ਹਨ।

ਇਕ ਫੰਕਸ਼ਨ ਦੇ ਅਨੇਕ ਲਾਗੂ-ਕਰਨ (executable) ਵਾਲੇ ਸਟੇਟਮੈਂਟਸ ਹੋ ਸਕਦੇ ਹਨ।

ਮਿਸਾਲ ਲਈ :

```
do_nothing()
{
  ..........
}
```

return ਸਟੇਟਮੈਂਟ ਇਕ mechanism ਹੈ ਜੋ calling ਫੰਕਸ਼ਨ ਨੂੰ ਮੁੱਲ ਵਾਪਿਸ ਭੇਜਦਾ ਹੈ। ਇਹ ਇਕ ਵਿਕਲਪੀ ਸਟੇਟਮੈਂਟ (optional statement) ਵੀ ਹੈ। ਜੇਕਰ ਇਹ ਨਹੀਂ ਲਿਖਿਆ ਗਿਆ ਤਾਂ ਇਸ ਦਾ ਮਤਲਬ ਹੈ ਕਿ ਫੰਕਸ਼ਨ ਨੂੰ ਕੋਈ ਮੁੱਲ ਵਾਪਿਸ ਨਹੀ ਭੇਜਿਆ ਗਿਆ।

ਮਿਸਾਲ : ਇਸ ਪ੍ਰੋਗਰਾਮ ਤੇ ਵਿਚਾਰ ਕਰੀਏ ਜਿੱਥੇ ਫੰਕਸ਼ਨ ਬੁਲਾਇਆ ਜਾਂਦਾ ਹੈ ਜਾਂ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

```
#include"stdio.h"
void main( )
{
 void printline();
 printline( );
 printf("This illustrates the use of C functions\n");
 printline();
}

void printline( )
 {
  int a;
  for(a=1;a<46;a++)
  printf("-");
  printf("\n");
 }
```

the output will be:

```
---------------------------------------------
This illustrates the use of C functions
---------------------------------------------
```

ਸਟੇਟਮੈਂਟ ਦੇ ਇਕ ਸੈੱਟ ਤੇ ਵਿਚਾਰ ਕਰੋ ਜਿਹੜਾ ਫੰਕਸ਼ਨ ਪ੍ਰਿੰਟ-ਲਾਈਨ ਦੀ ਪਰਿਭਾਸ਼ਾ (definition) ਦਿੰਦਾ ਹੋਵੇ।

```
printline()
{
int a;
for (a =1;a<46;a++)
printf("-");
printf("\n");
}
```

the output will be:

```
---------------------------------------------

---------------------------------------------
```

ਉਪਰੋਕਤ ਮਿਸਾਲ ਵਿਚ ਦੋ ਯੂਜ਼ਰ-ਡਿਫਾਇੰਡ-ਫੰਕਸ਼ਨ ਹਨ : 1. main() 2. printline()

ਸੀ (C) ਵਿਚ ਪ੍ਰੋਗਰਾਮ ਲਾਗੂ ਕਰਨ, ਮੇਨ ਫੰਕਸ਼ਨ ਦੇ ਲਾਗੂ ਕਰਨ ਦੇ ਨਾਲ ਸ਼ੁਰੂ ਹੁੰਦਾ ਹੈ।ਪਹਿਲਾ ਸਟੇਟਮੈਂਟ ਪ੍ਰਿੰਟ-ਲਾਈਨ ਫੰਕਸ਼ਨ (printline function) ਮਿਲਦਾ ਹੈ।

ਇਹ ਇਸ ਗੱਲ ਦਾ ਪ੍ਰਤੀਕ ਹੈ ਕਿ ਫੰਕਸ਼ਨ ਪ੍ਰਿੰਟ-ਲਾਈਨ **(printline)** ਲਾਗੂ ਕੀਤੀ ਜਾਣੀ ਹੈ। ਇਸ ਜਗ੍ਹਾ ਤੇ ਪ੍ਰੋਗਰਾਮ ਕੰਟਰੋਲ ਫੰਕਸ਼ਨ ਪ੍ਰਿੰਟ-ਲਾਈਨ ਕੋਲ ਚਲਾ ਜਾਂਦਾ ਹੈ। ਫੰਕਸ਼ਨ ਪ੍ਰਿੰਟ-ਲਾਈਨ **(function printline)** ਨੂੰ ਲਾਗੂ ਕਰਨ ਤੋਂ ਬਾਅਦ 46 ਚਿੰਨ੍ਹਾਂ ਦੀ ਲਾਈਨ ਦਾ ਆਊਟਪੁੱਟ ਆਉਂਦਾ ਹੈ। printf() ਸਟੇਟਮੈਂਟ ਨੂੰ ਲਾਗੂ ਕਰਨ ਤੋਂ ਬਾਅਦ ਕੰਟਰੋਲ ਫਿਰ ਪ੍ਰਿੰਟ-ਲਾਈਨ ਫੰਕਸ਼ਨ ਤੇ ਲਾਈਨ (46 dots) ਨੂੰ ਪ੍ਰਿੰਟ ਕਰਨ ਵਾਸਤੇ ਚਲਾ ਜਾਂਦਾ ਹੈ।

ਮੇਨ ਫੰਕਸ਼ਨ, user defined functions ਨੂੰ ਦੋ ਵਾਰ ਅਤੇ library function printf ਨੂੰ ਬਾਰ-ਬਾਰ ਬੁਲਾਉਂਦਾ ਹੈ। ਕੋਈ ਵੀ ਫੰਕਸ਼ਨ ਕਿਸੇ ਵੀ ਫੰਕਸ਼ਨ ਨੂੰ ਸੱਦ ਸਕਦਾ ਹੈ। ਫੰਕਸ਼ਨ ਆਪਣੇ ਆਪ ਨੂੰ ਵੀ ਸੱਦ ਸਕਦਾ ਹੈ। called ਫੰਕਸ਼ਨ ਕਿਸੇ ਹੋਰ ਫੰਕਸ਼ਨ ਨੂੰ ਵੀ ਸੱਦ ਸਕਦਾ ਹੈ। ਇਕ ਫੰਕਸ਼ਨ ਨੂੰ ਇਕ ਤੋਂ ਵੱਧ ਵਾਰ ਵੀ ਸੱਦਿਆ ਜਾ ਸਕਦਾ ਹੈ।

ਆਰਗੂਮੈਂਟ ਲਿਸਟ (Argument List)

ਆਰਗੂਮੈਂਟ ਲਿਸਟ ਵਿਚ ਕੌਮਿਆਂ ਦੁਆਰਾ ਵੱਖ ਕੀਤੇ ਗਏ ਇਕ ਤੋਂ ਵੱਧ ਵੇਰੀਏਬਲ ਦੇ ਨਾਮ ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ। ਆਰਗੂਮੈਂਟਸ, ਵੇਰੀਏਬਲਸ called ਫੰਕਸ਼ਨ ਤੋਂ ਮੁੱਲ ਪ੍ਰਾਪਤ ਕਰਦੇ ਹਨ। ਇਸ ਪ੍ਰਕਾਰ ਫੰਕਸ਼ਨ ਤੋਂ calling function ਡਾਟਾ ਸੰਚਾਰ ਲਈ ਸਾਧਨ ਪ੍ਰਦਾਨ ਕਰਦੇ ਹਨ।

ਆਰਗੂਮੈਂਟਸ ਨਾਲ ਫੰਕਸ਼ਨਸ ਦੀਆਂ ਕੁਝ ਮਿਸਾਲਾਂ ਹਨ : (1) Ram(a, b, c) **(2)** gopal(s, d, f)

ਸਾਰੇ ਆਰਗੂਮੈਂਟ, ਵੇਰੀਏਬਲਸ ਨੂੰ ਫੰਕਸ਼ਨ header ਦੇ ਬਾਅਦ ਅਤੇ ਫੰਕਸ਼ਨ ਬਾਡੀ ਦੇ ਸ਼ੁਰੂ ਹੋਣ ਤੋਂ ਪਹਿਲਾਂ ਉਨ੍ਹਾਂ ਦੀਆਂ ਕਿਸਮਾਂ ਨਾਲ ਡਿਕਲੇਅਰ (declare) ਕੀਤਾ ਜਾਣਾ ਚਾਹੀਦਾ ਹੈ।

ਮਿਸਾਲ :

```
sham(x,y)
float x;
int y;
{
--------
--------
}
```

ਰਿਟਰਨ ਮੁੱਲ ਅਤੇ ਉਨ੍ਹਾਂ ਦੇ ਟਾਈਪਸ (Return values and their types)

ਇਕ ਫੰਕਸ਼ਨ ਬੁਲਾਏ ਜਾ ਰਹੇ ਫੰਕਸ਼ਨ ਨੂੰ ਕੋਈ ਮੁੱਲ ਵਾਪਸ ਭੇਜ ਵੀ ਸਕਦਾ ਹੈ ਤੇ ਨਹੀ ਵੀ। ਜੇਕਰ ਇਹ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਤਾਂ ਇਹ Return statement ਦੁਆਰਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਜਦਕਿ ਇਹ ਸੰਭਵ ਹੈ ਕਿ Calling function ਨੂੰ ਕਿੰਨੇ ਵੀ ਮੁੱਲ ਭੇਜੇ ਜਾ ਸਕਦੇ ਹਨ। Called ਫੰਕਸ਼ਨ ਇਕ ਕਾਲ (per call) ਲਈ ਇਕ ਮੁੱਲ ਹੀ ਵਾਪਿਸ ਕਰ ਸਕਦਾ ਹੈ।

```
return ;
or
return(expression) ;
```

ਪਹਿਲੀ return statement ਵੱਲੋਂ ਕੋਈ ਮੁੱਲ ਵਾਪਸ ਨਹੀਂ ਕੀਤਾ ਜਾਂਦਾ। ਇਹ ਫੰਕਸ਼ਨ ਦੇ ਅੰਤਿਮ ਬਰੇਸ (brace) ਵਜੋਂ ਕੰਮ ਕਰਦੇ ਹਨ। ਜਦੋਂ ਇਕ return statement ਦਾ ਸਾਹਮਣਾ ਹੁੰਦਾ ਹੈ ਤਾਂ ਕੰਟਰੋਲ ਜਲਦੀ ਹੀ Called ਫੰਕਸ਼ਨ ਵਿਚ ਭੇਜ ਦਿੱਤਾ ਜਾਂਦਾ ਹੈ।

ਸਿੰਪਲ ਰਿਟਰਨ (simple return) ਦੀ ਵਰਤੋਂ ਦੀ ਮਿਸਾਲ ਨਿਮਨ ਅਨੁਸਾਰ ਹੈ :

```
If(error)
return;
```

return ਦੇ ਦੂਜੇ ਰੂਪ ਵਿਚ expression returns ਨਾਲ ਮੁੱਲ ਦਾ expression

ਮੁਤਾਬਿਕ ਤੌਰ ਤੇ ਇਕ ਫੰਕਸ਼ਨ int ਟਾਈਪ ਡਾਟਾ ਵਾਪਿਸ ਕਰਦੇ ਹਨ। ਜੇਕਰ ਇਕ ਫੰਕਸ਼ਨ ਕਿਸੇ ਹੋਰ ਕਿਸਮ ਦਾ ਮੁੱਲ (ਇਹ string, floating, character ਜਾਂ structure ਹੋ ਸਕਦਾ ਹੈ) ਵਾਪਿਸ ਭੇਜਣਾ ਚਾਹੁੰਦੇ ਹੋਣ ਤਾਂ ਇਹ ਭਾਵੇਂ ਯੂਜ਼ਰ ਡਿਫਾਇੰਡ ਹੋਵੇ ਜਾਂ ਨਿਰਮਿਤ ਡਾਟਾ ਟਾਈਪ ਦਾ ਹੋਵੇ ਤਾਂ ਇਸ ਮੁੱਲ ਨੂੰ ਫੰਕਸ਼ਨ ਹੈਡਰ ਵਿਚ "ਟਾਈਪ-ਸਪੇਸੀਫਾਇਰ" (type specifier) ਦੇ ਕੇ ਪ੍ਰਾਪਤ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

ਮਿਸਾਲ :

```
double sum(w, r)
char sham(c)
```

ਜਦੋਂ ਮੁੱਲ ਵਾਪਿਸ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਤਾਂ ਇਹ ਫੰਕਸ਼ਨ ਦੀ ਟਾਈਪ ਨੂੰ ਪ੍ਰਭਾਵਿਤ ਕਰਦਾ ਹੈ। ਫੰਕਸ਼ਨ ਵਿਚ ਇਹ double ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਗਣਨਾ ਕਰਦੇ ਹਨ ਅਤੇ ਫਿਰ ਵੀ int ਹੀ ਵਾਪਸ ਕਰਦੇ ਹਨ।

ਮਿਸਾਲ A ਵਿਚ ਅਸੀਂ ਪਹਿਲੀ ਲਾਈਨ ਵਿਚ ਡਾਟਾ ਟਾਈਪ ਦੀ ਡਿਕਲੇਅਰੇਸ਼ਨ ਕੀਤੀ ਹੈ ਜੋ ਫੰਕਸ਼ਨ ਦੁਆਰਾ ਵਾਪਿਸ ਹੁੰਦੀ ਹੈ ਕਿਉਂਕਿ ਵਾਪਸ ਕੀਤਾ ਗਿਆ ਮੁੱਲ long int type ਦਾ ਹੈ। ਜੇਕਰ ਅਸੀਂ long int ਨੂੰ ਨਿਸ਼ਚਿਤ ਨਹੀਂ ਕਰਦੇ ਤਾਂ ਫੰਕਸ਼ਨ ਸਿਰਫ integer ਮੁੱਲ ਨੂੰ ਹੀ ਵਾਪਿਸ ਕਰਦਾ ਹੈ।

ਮਿਸਾਲ (A) : Calculating the factorial of a number(sample program)

```
long int factorial(a)
{
int a, b;
long int prod =1;
if (a>1)
for (b =2;b<=a; ++b)
prod * = a;
return(prod);
}
```

ਮਿਸਾਲ : (B)

```
mul(x,y)
int x,y;
{
 int p;
 p = x*y;
 return(p);
}
```

ਅਸੀਂ ਅੰਤਿਮ ਦੋ ਬਿਆਨਾਂ ਨੂੰ ਮਿਲਾ ਵੀ ਸਕਦੇ ਹਾਂ ਜਿਵੇਂ

```
return (x*y);
```

ਫੰਕਸ਼ਨ ਨੂੰ ਕਾਲ ਕਰਨਾ (Calling of a function)

ਮਿਸਾਲ: ਇਕ ਸਟੇਟਮੈਂਟ ਵਿਚ ਫੰਕਸ਼ਨ ਦੇ ਨਾਮ ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹੋਏ ਕੋਈ ਵੀ ਫੰਕਸ਼ਨ ਬੁਲਾਇਆ ਜਾ ਸਕਦਾ ਹੈ।

- ਇਕ ਫੰਕਸ਼ਨ ਜਿਹੜਾ ਕੋਈ ਮੁੱਲ ਵਾਪਸ ਨਹੀਂ ਕਰਦਾ ਐਕਸਪ੍ਰੈਸ਼ਨ (expression) ਵਿਚ ਵਰਤਿਆ ਜਾਵੇ ਜਾਂ ਨਾ) ਪਰੰਤੂ ਇਸ ਨੂੰ ਫੰਕਸ਼ਨ ਵਿਚ ਕੁਝ ਵਿਸ਼ੇਸ਼ ਕੰਮ ਕਰਨ ਲਈ ਬੁਲਾਇਆ ਜਾ ਸਕਦਾ ਹੈ। ਪਹਿਲਾਂ ਦਿਖਾਇਆ ਫੰਕਸ਼ਨ, ਫੰਕਸ਼ਨ ਪ੍ਰਿੰਟਲਾਈਨ() ਜਿਸਦਾ ਵਰਣਨ ਉਪਰੋਕਤ ਕੀਤਾ ਗਿਆ ਹੈ, ਦਾ ਸੰਬੰਧ ਇਸ ਸ਼੍ਰੇਣੀ ਨਾਲ ਹੈ। ਅਜਿਹੇ ਫੰਕਸ਼ਨ ਕੇਵਲ ਸੁਤੰਤਰ ਸਟੇਟਮੈਂਟਸ ਵਜੋਂ ਉਨ੍ਹਾਂ ਦੇ ਨਾਮ ਬਿਆਨ ਕੇ ਬੁਲਾਏ ਜਾ ਸਕਦੇ ਹਨ।

ਮਿਸਾਲ :

```
void  main()
{
printline( );
}
```

ਮਿਸਾਲ :

```
void main()
{
int p;
p = mul(5,3);
printf("%d\n" , p);
```

ਜਦੋਂ ਇਕ ਕੰਪਾਈਲਰ ਵੱਲੋਂ ਫੰਕਸ਼ਨ-ਕਾਲ ਦਾ ਸਾਹਮਣਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਤਾਂ ਕੰਟਰੋਲ ਫੰਕਸ਼ਨ mul (x, y) ਨੂੰ ਦਿੱਤਾ ਜਾਂਦਾ ਹੈ। ਇਹ ਫੰਕਸ਼ਨ ਫਿਰ ਉਪਰੋਕਤ ਬਿਆਨੇ ਗਏ ਅਨੁਸਾਰ ਲਾਗੂ (execute) ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਮੁੱਲ ਉਦੋਂ ਵਾਪਸ ਜਾਂਦਾ ਹੈ ਜਦੋਂ return statement ਦਾ ਸਾਹਮਣਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਇਹ ਮੁੱਲ p ਨੂੰ ਅਸਾਈਨ (assign) ਕੀਤਾ ਜਾਂਦਾ ਹੈ।

ਇਕ ਫੰਕਸ਼ਨ ਜਿਹੜਾ ਮੁੱਲ ਵਾਪਸ ਕਰਦਾ ਹੈ ਕਿਸੇ ਹੋਰ ਵੇਰੀਏਬਲ (variable) ਦੀ ਤਰ੍ਹਾਂ ਪ੍ਰਗਟਾਓ (expression) ਵਿਚ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ।

```
printf("%d\n",mul(p,q));
y = mul(p,q)/(p+q);
```

ਮਿਸਾਲ : ਫੰਕਸ਼ਨ "Calsum()"ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਜਮਾਂ ਕਰਨ ਦਾ ਪ੍ਰੋਗਰਾਮ

```
#include<stdio.h>
void main()
{
 int calsum(int x ,int y ,int z);
 int a,b,c,sum;
 printf("\n Enter the three numbers:");
 scanf("%d %d %d" ,&a, &b, &c);
 sum =calsum(a,b,c);
 printf("\n sum = %d" ,sum);
}
                    // defining the function
int calsum(int x ,int y ,int z)
{
 int d;
 d = x+y+z ;
 return(d) ;
}
```

ਪਰਿਣਾਮ (Output)

```
Enter the three numbers 20 30 50
sum = 100
```

ਇਸ ਮਿਸਾਲ ਵਿਚ main() ਤੋ a,b,c ਦੇ ਮੁੱਲ ਫੰਕਸ਼ਨ Calsum ਨੂੰ ਦਿੱਤੇ ਗਏ ਹਨ।

a,b,c ਵੇਰੀਏਬਲਸ ਨੂੰ actual ਆਗਗੂਮੈਂਟਸ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ਅਤੇ x,y,z ਨੂੰ formal ਆਗਗੂਮੈਂਟਸ ਕਿਹਾ ਜਾਂਦਾ ਹੈ

ਇਨ੍ਹਾਂ ਦੋਹਾਂ ਆਰਗੂਮੈਂਟਸ ਦੇ ਟਾਈਪ ਅਤੇ ਕ੍ਰਮ ਇਕੋ ਜਿਹੇ ਹੋਣੇ ਚਾਹੀਦੇ ਹਨ।

ਮਿਸਾਲ: ਆਉ ਦੇਖੀਏ ਫੰਕਸ਼ਨ ਦੇ ਆਰਗੂਮੈਂਟਸ ਜੋ calling ਫੰਕਸ਼ਨ ਨੂੰ ਮੁੱਲ ਵਾਪਸ ਕਿਵੇਂ ਕਰਦੇ ਹਨ

```
#include<stdio.h>
void main()
{
 int length,width;
 printf("Enter the length of the rectangle:");
 length = get_an_int();
 printf("Enter the width: ");
 width = get_an_int();
 printf("\n The area is %d. \n",length * width);
}
 // the following function inputs an integer and returns its value

get_an_int( )
{
 int a ;
 scanf("%d" ,&a);
 return (a);
}
```

ਪਰਿਣਾਮ (Output)

```
Enter the length of the rectangle    15
Enter the width                       5
The area is :                        75
```

ਪੂਰਣ-ਅੰਕ ਨੂੰ ਫੰਕਸ਼ਨ get_an_int ਪੜ੍ਹੇਗਾ ਅਤੇ return statement ਦੁਆਰਾ ਮੇਨ-ਫੰਕਸ਼ਨ ਨੂੰ ਇਹ ਮੁੱਲ ਵਾਪਸ ਕਰੇਗਾ। ਵੇਰੀਏਬਲ (variable) **a** ਦੀ ਵਰਤੋਂ ਮੁੱਲ ਪੜ੍ਹਣ ਵਿਚ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਜੋ ਮੁੱਲ ਦੇ ਮੇਨ-ਫੰਕਸ਼ਨ ਵਿਚ ਵਾਪਿਸ ਤੋਂ ਪਹਿਲਾਂ ਦਿੱਤੀ ਜਾਂਦੀ ਹੈ। ਇਸ ਨੇ ਉਸ ਵੇਰੀਏਬਲ ਦੇ ਨਾਲ ਕੁਝ ਨਹੀਂ ਕਰਨਾ ਹੁੰਦਾ ਜੋ ਅੰਤ ਵਿਚ ਮੁੱਲ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ। ਫੰਕਸ਼ਨ ਨੂੰ ਦੋ ਵਾਰ ਵਾਪਸ ਬੁਲਾਇਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਜੋ ਮੇਨ-ਫੰਕਸ਼ਨ ਵੱਲੋਂ ਦੋ ਪੂਰਣ-ਅੰਕ, ਇਕ ਲੰਬਾਈ ਤੇ ਚੌੜਾਈ ਪ੍ਰਾਪਤ ਕੀਤੀ ਜਾ ਸਕੇ। ਮੇਨ-ਫੰਕਸ਼ਨ ਵੱਲੋਂ ਫਿਰ ਇਸ ਲੰਬਾਈ ਅਤੇ ਚੌੜਾਈ ਦੇ ਨਾਲ ਆਇਤਾਕਾਰ ਦੇ ਖੇਤਰਫਲ ਦੀ ਗਣਨਾ ਅਤੇ ਪ੍ਰਿੰਟਿੰਗ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।

ਜਦੋਂ actual argument ਰਾਹੀਂ ਇਕ ਫੰਕਸ਼ਨ ਨੂੰ ਇਕੱਲਾ ਮੁੱਲ ਭੇਜਿਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਇਸ ਮੁੱਲ ਨੂੰ ਫੰਕਸ਼ਨ ਵਿਚ ਕਾਪੀ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਉੱਤਰ-ਵਰਤੀ formal argument ਦਾ ਮੁੱਲ ਫੰਕਸ਼ਨ ਦੇ ਵਿਚ ਬਦਲ ਜਾਂਦਾ ਹੈ ਪਰੰਤੂ calling ਫੰਕਸ਼ਨ ਦੇ ਨਾਲ actual argument ਦਾ ਮੁੱਲ ਨਹੀਂ ਬਦਲੇਗਾ। ਇਕ argument ਦੇ ਮੁੱਲ ਨੂੰ ਫੰਕਸ਼ਨ ਵਿਚ ਭੇਜਣ ਦੀ ਵਿਧੀ ਨੂੰ passing by value ਵਜੋਂ ਜਾਣਿਆ ਜਾਂਦਾ ਹੈ।

ਉਪਰੋਕਤ ਸਾਰੀਆਂ ਮਿਸਾਲਾਂ ਅਤੇ ਹੇਠਲੀ ਮਿਸਾਲ ਇਸ ਸੰਕਲਪ ਨੂੰ ਦਰਸਾਉਂਦੀਆਂ ਹਨ।

ਮਿਸਾਲ :

```
#include<stdio.h>
void main()
{
  int a =2;
  int modify(int);
  printf("\n a =%d(before calling the function in main",a);
  modify(a);
  printf("\n\n a =%d(after calling the function in main)",a);
}

int modify(int a)
{
   a *= 3;
printf("\n\n a =%d(from the function after being modified )" ,a);
   return(0);
}
```

ਪਰਿਣਾਮ (Output)

a = 2 (before calling the function in main)
a = 6 (from the function after being modified)
a = 2 (after calling the function in main)

ਇਹ ਪਰਿਣਾਮ ਦਰਸਾਉਂਦੇ ਹਨ ਕਿ **a** ਦਾ ਮੁੱਲ main ਵਿਚ ਨਹੀਂ ਬਦਲਦਾ ਇਥੋਂ ਤਕ ਕਿ **a** ਦਾ ਮੁੱਲ ਫੰਕਸ਼ਨ modify ਦੇ ਵਿਚ ਬਦਲ ਜਾਂਦਾ ਹੈ। ਇਸ ਲਈ ਭੇਜਿਆ ਜਾਣ ਵਾਲਾ ਮੁੱਲ ਸੂਚਨਾ ਦੇ ਇਕ ਤਰਫਾ ਤਬਦੀਲੀ ਤਕ ਹੀ ਸੀਮਿਤ ਹੈ ਕਿਉਂ ਜੋ ਇਹ ਸੂਚਨਾ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਦੇ ਬੁਲਾਏ ਜਾਣ ਵਾਲੇ ਭਾਗ ਤੋਂ ਵਾਪਸ ਕੀਤੇ ਜਾਣ ਨੂੰ ਰੋਕਦਾ ਹੈ।

ਮਿਸਾਲ : ਆਰਗੂਮੈਂਟਸ (argument) ਪ੍ਰਦਾਨ ਕਰਨ ਵਾਲਾ ਫੰਕਸ਼ਨ ਜਿਹੜਾ ਇਕ ਮੁੱਲ ਵਾਪਸ ਕਰਦਾ ਹੈ। ਉਹ ਮੁੱਲ called ਫੰਕਸ਼ਨ ਵਿਚ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ਜਿੱਥੇ ਇਸ ਫੰਕਸ਼ਨ ਦਾ ਹਵਾਲਾ ਦਿੱਤਾ ਗਿਆ ਹੈ।

```
#include<stdio.h>
void main()
{
 float a,b,c;
 float avg(float,float);
 a = 10.6;
 b = 20.2;
 c =avg(a,b);
 printf("average =%f\n" ,c);
}

float avg(float x,float y)
{
  float z;
  z =(x+y)/2;
  return(z);
}
```

ਪਰਿਣਾਮ (Output):

Average = 15.400001

main() ਵਿਚ **a** ਅਤੇ **b** ਪੈਰਾਮੀਟਰਸ (parameters) ਵਾਂਗ ਹਨ ਜੋ ਫੰਕਸ਼ਨ avg ਨੂੰ ਭੇਜੇ ਜਾਂਦੇ ਹਨ ਜਿਹੜਾ **c** ਨੂੰ ਸਮਰਪਿਤ ਕੀਤਾ ਗਿਆ, ਔਸਤਨ ਅੰਕ ਵਾਪਿਸ ਭੇਜਦਾ ਹੈ। **a** ਦਾ ਮੁੱਲ **x** ਅਤੇ **b** ਨੂੰ **y** ਵਿਚ ਸਮਰਪਣ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। return statement ਇਕ ਮੁੱਲ ਨੂੰ ਵਾਪਿਸ ਕਰਨ ਵਿਚ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ, ਜੋ ਕਿ, ਦੋ ਅੰਕਾਂ ਦੀ ਔਸਤ ਵਿਚ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

ਫੰਕਸ਼ਨ ਨਾਲ ਆਰਗੂਮੈਂਟਸ, ਬਗੈਰ ਰਿਟਰਨ ਮੁੱਲ (Function with arguments but no return value)

ਇਥੇ called ਫੰਕਸ਼ਨਜ਼, calling ਫੰਕਸ਼ਨ ਤੋਂ ਡਾਟਾ ਪ੍ਰਾਪਤ ਕਰਦੇ ਹਨ। ਆਰਗੂਮੈਂਟਸ (arguments) ਅਤੇ ਪੈਰਾਮੀਟਰਸ (parameters) ਦੇ ਅੰਕ, ਡਾਟਾ ਟਾਈਪ ਅਤੇ ਕ੍ਰਮ ਜਰੂਰ ਮੇਲ ਖਾਣੇ ਚਾਹੀਦੇ ਹਨ। ਪਰੰਤੂ called ਫੰਕਸ਼ਨ, calling ਫੰਕਸ਼ਨ (function) ਨੂੰ ਕੋਈ ਮੁੱਲ ਵਾਪਸ ਨਹੀਂ ਭੇਜਦਾ। ਇਸਦੀ ਬਜਾਏ ਇਹ ਕੇਵਲ ਆਪਣੀ ਗੁੰਜਾਇਸ਼ ਵਿਚ ਡਾਟਾ ਪ੍ਰਿੰਟ ਕਰਦਾ ਹੈ। ਇਹ calling ਫੰਕਸ਼ਨ ਅਤੇ called ਫੰਕਸ਼ਨ ਦੇ ਦਰਮਿਆਨ ਇਕ ਤਰਫਾ ਡਾਟਾ ਸੰਚਾਰ ਹੁੰਦਾ ਹੈ।

ਮਿਸਾਲ : **Function with arguments but no return value**

```
#include <stdio.h>
void main()
{
 void read_value();
 read_value();
}

void read_value()                               //no return value
{
char name[10];
printf("\nEnter the ycur name: ");
scanf("%s" ,name);
printf("Your name is %s" ,name);
}
```

ਪਰਿਣਾਮ (Output):

Enter your name : Ram
Your name is Ram

ਮਿਸਾਲ :

```
#include <stdio.h>
void main()
{
  int x,y;
  int maximum(int,int);
  printf("Enter the value x and y\n");
  scanf("%d %d",&x,&y);
  maximum(x,y);
}
// function to compute the maximum
  maximum (int p,int q)
  {
  if (p>q)
     printf("Maximum = %d\n",p);
  else
     printf("Maximum = %d\n",q);
  return(0) ;
}
```

ਪਰਿਣਾਮ (Output)

Enter the value of x and y
14
7
Maximum = 14

ਦਿੱਤੀ ਗਈ ਮਿਸਾਲ ਵਿਚ called ਫੰਕਸ਼ਨ ਦੇ ਦੋ ਆਰਗੂਮੈਂਟਸ (arguments) x ਅਤੇ y ਨੂੰ ਮੁੱਲ ਦਿੱਤਾ ਜਾਂਦਾ ਹੈ, ਜੋ ਕਿ calling ਫੰਕਸ਼ਨ ਦੇ ਦੋ ਆਰਗੂਮੈਂਟਸ (arguments) p ਅਤੇ q ਨਾਲ ਮੈਚ (match) ਕੀਤੇ ਜਾਂਦੇ ਹਨ।

ਸਟੋਰੇਜ਼ ਸ਼੍ਰੇਣੀਆਂ (Storage Classes)

ਵੇਰੀਏਬਲ ਦੀ ਵਿਸ਼ੇਸ਼ਤਾ ਨੂੰ ਡਾਟਾ-ਟਾਈਪ ਅਤੇ ਸਟੋਰੇਜ-ਸ਼੍ਰੇਣੀ (Storage Class) ਦੇ ਆਧਾਰ ਤੇ ਦੋ ਭਾਗਾਂ ਵਿਚ ਵੰਡਿਆ ਜਾ ਸਕਦਾ ਹੈ। ਸਟੋਰੇਜ਼-ਸ਼੍ਰੇਣੀ (Storage Class) ਵੇਰੀਏਬਲ ਦੀ ਪ੍ਰਾਥਮਿਕਤਾ ਦਾ ਹਵਾਲਾ ਦਿੰਦੀ ਹੈ ਅਤੇ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਇਸਦੀ ਗੁੰਜਾਇਸ਼ ਬਿਆਨ ਕਰਦੀ ਹੈ। ਸੀ (C) ਵਿਚ ਚਾਰ ਵੱਖ-ਵੱਖ ਕਿਸਮ ਦੀਆਂ ਸਟੋਰੇਜ-ਸ਼੍ਰੇਣੀਆਂ (Storage ਹੁੰਦੀਆਂ ਹਨ ।

ਆਟੋਮੈਟਿਕ (Automatic), ਐਕਸਟਰਨਲ (External), ਸਟੈਟਿਕ (Static) ਅਤੇ ਰਜਿਸਟਰ (Register). ਇਥੇ ਅਸੀ ਸਟੈਟਿਕ (Static) ਅਤੇ ਐਕਸਟਰਨਲ (External) ਬਾਰੇ ਹੀ ਵਿਚਾਰ ਵਟਾਂਦਰਾ ਕਰ ਰਹੇ ਹਾਂ।

ਸਟੈਟਿਕ ਵੇਰੀਏਬਲ (static variable)

ਇਹ ਮੇਮੋਰੀ ਵਿਚ ਸਟੋਰ ਕੀਤੇ ਜਾਂਦੇ ਹਨ। ਉਨ੍ਹਾਂ ਦਾ ਪਹਿਲਾ ਮੁੱਲ ਜ਼ੀਰੋ ਹੁੰਦਾ ਹੈ। ਉਹ ਉਸ ਬਲਾਕ ਵਿਚ ਲੋਕਲ ਹੁੰਦੇ ਹਨ ਜਿਸ ਵਿਚ ਵੇਰੀਏਬਲ ਨੂੰ ਪਰਿਭਾਸ਼ਤ (define) ਕੀਤਾ ਗਿਆ ਹੈ।

ਐਕਸਟਰਨਲ ਵੇਰਿਏਬਲ (external variable)

ਇਹ ਕਿਸੇ ਵੀ ਫੰਕਸ਼ਨ ਦੇ ਬਾਹਰ ਪਰਿਭਾਸ਼ਤ (define) ਕੀਤੇ ਜਾਂਦੇ ਹਨ ਅਤੇ ਇਸ ਪ੍ਰਕਾਰ ਬਹੁਤ ਸਾਰੇ ਫੰਕਸ਼ਨਸ ਲਈ ਉੱਪਲਬਧ ਹੁੰਦੇ ਹਨ। ਇਹ ਮੇਮੋਰੀ ਵਿਚ ਸੇਵ ਕੀਤੇ ਜਾਂਦੇ ਹਨ ਅਤੇ ਇਨ੍ਹਾਂ ਦਾ ਪਹਿਲਾ ਮੁੱਲ ਜ਼ੀਰੋ ਹੁੰਦਾ ਹੈ। ਇਨ੍ਹਾਂ ਦਾ ਸਕੋਪ (scope) ਪ੍ਰੋਗਰਾਮ ਦੇ ਲਾਗੂ ਹੋਣ ਤੋਂ ਖ਼ਤਮ ਹੋਣ ਦੇ ਸਮੇਂ ਦੇ ਵਿਚ ਹੁੰਦਾ ਹੈ। ਐਕਸਟਰਨਲ ਵੇਰਿਏਬਲ (external variable) ਹੋਰ ਵੇਰਿਏਬਲਸ ਤੋਂ ਭਿੰਨ ਹੁੰਦੇ ਹਨ ਕਿਉਂ ਜੋ ਉਨ੍ਹਾਂ ਦੀ ਗੁੰਜਾਇਸ਼ ਗਲੋਬਲ ਹੁੰਦੀ ਹੈ ਨਾ ਕਿ ਲੋਕਲ।

ਮਿਸਾਲ : ਸਟੈਟਿਕ ਵੇਰੀਏਬਲ (static variable)

```
#include<stdio.h>
void main()
{
 void increase();
 increase( ) ;
 increase( ) ;
 increase( ) ;
}
void increase( )
{
 static int a = 1;
 printf("%d\n",a);
 a = a + 1;
}
```

ਪਰਿਣਾਮ (Output)

```
1
2
3
```

ਸਟੈਟਿਕ ਵੇਰੀਏਬਲ ਉਸ ਸਮੇਂ ਵੀ ਅਲੋਪ ਨਹੀਂ ਹੁੰਦਾ ਜਦੋਂ ਫੰਕਸ਼ਨ ਸਰਗਰਮ ਨਹੀਂ ਵੀ ਰਹਿੰਦਾ। ਇਨ੍ਹਾਂ ਦੇ ਮੁੱਲ ਕਾਇਮ ਰਹਿੰਦੇ ਹਨ। ਜੇਕਰ ਕੰਟਰੋਲ ਮੁੜ ਉਸੇ ਫੰਕਸ਼ਨ ਵਿਚ ਵਾਪਸ ਆ ਜਾਂਦਾ ਹੈ ਤਾਂ **ਸਟੈਟਿਕ ਵੇਰੀਏਬਲ** ਦਾ ਉਹੀ ਮੁੱਲ ਰਹਿੰਦਾ ਹੈ ਜਿਹੜਾ ਉਸ ਵਿਚ ਪਿਛਲੇ ਸਮੇਂ ਤੇ ਸੀ।

ਮਿਸਾਲ : ਐਕਸਟਰਨਲ ਵੇਰੀਏਬਲ (external variable)

```
#include <stdio.h>
int a;
void main()
{
void inc();
void dec();
 printf("\n a= %d" ,a);
 inc( );
 inc( );
 dec( );
 dec( );
 }
void inc()
   {
    a= a + 1;
    printf("\n on increasing  a = %d" ,a);
   }
void dec()
 {
        a = a - 1 ;
        printf("\n on decreasing  a = %d" ,a);
 }
```

ਪਰਿਣਾਮ (Output)

```
a = 0
on increasing =  1
on increasing =  2
on decreasing = 1
on decreasing = 0
```

ਇਹ ਸਪਸ਼ਟ ਹੈ ਕਿ **a** ਦਾ ਮੁੱਲ ਦੋਹਾਂ ਫੰਕਸ਼ਨ inc ਅਤੇ dec ਕੋਲ ਉਪਲਬਧ ਹੈ। ਜਿਵੇਂ ਕਿ **a** ਨੂੰ ਸਮੁੱਚੇ ਫੰਕਸ਼ਨ ਦੇ ਬਾਹਰ ਡਿਕਲੇਅਰ (declare) ਕੀਤਾ ਗਿਆ ਹੈ।

ਫੰਕਸ਼ਨ ਬਗੈਰ ਆਰਗੂਮੈਂਟਸ, ਰਿਟਰਨ ਮੁੱਲ ਨਾਲ (Function with no arguments but return value)

ਇਥੇ called ਫੰਕਸ਼ਨ , calling ਫੰਕਸ਼ਨ ਤੋਂ ਕੋਈ ਡਾਟਾ ਪ੍ਰਾਪਤ ਨਹੀਂ ਕਰਦਾ। ਇਹ ਵਿਸ਼ੇਸ਼ ਫੰਕਸ਼ਨ ਕਰਨ ਲਈ ਲੋਕਲ ਡਾਟਾ ਦਾ ਹੀ ਪ੍ਰਬੰਧ ਕਰਦਾ ਹੈ। ਨਿਸ਼ਚਿਤ ਪ੍ਰੋਸੈਸ ਦੇ ਬਾਅਦ called ਫੰਕਸ਼ਨ ਗਣਨਾ ਕੀਤੇ ਗਏ ਡਾਟੇ ਨੂੰ calling ਫੰਕਸ਼ਨ ਵੱਲ ਵਾਪਸ ਭੇਜ ਦਿੰਦਾ ਹੈ। ਇਹ called ਫੰਕਸ਼ਨ (function) ਅਤੇ calling ਫੰਕਸ਼ਨ ਦੇ ਵਿਚਕਾਰ ਇਕ ਤਰਫਾ ਡਾਟਾ ਸੰਚਾਰ ਹੁੰਦਾ ਹੈ।

ਮਿਸਾਲ :

```
#include <stdio.h>
void main()
{
 float sum ;
 float total ( );
 sum =total ( );
 printf("sum = %f\n",sum);
}
float total ( )
{
 float x,y;
 x = 20.0 ;
 y = 10.0;
 return (x+y);
}
```

ਪਰਿਣਾਮ (Output):

Sum = 30.000000

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1: ਬਹੁਪੱਖੀ ਪਸੰਦ ਦੇ ਪ੍ਰਸ਼ਨ

1.1 **ਸੀ ਵਿਚ ਪ੍ਰੋਗਰਾਮ ਇਨ੍ਹਾਂ ਵਿਚੋਂ ਕਿਸ ਨਾਲ ਸ਼ੁਰੂ ਹੁੰਦਾ ਹੈ ?**

a) ਫੰਕਸ਼ਨ (function) ਜਿਹੜਾ ਪਹਿਲਾਂ ਪਰਿਭਾਸ਼ਤ ਕੀਤਾ ਗਿਆ b) main() ਫੰਕਸ਼ਨ (function)
c) ਫੰਕਸ਼ਨ (function) ਜਿਹੜਾ ਅੰਤ ਵਿਚ ਪਰਿਭਾਸ਼ਤ ਕੀਤਾ ਗਿਆ d) main () ਤੋਂ ਇਲਾਵਾ ਫੰਕਸ਼ਨ

1.2 **ਇਕ ਫੰਕਸ਼ਨ (function) ਜਿਸਦੀ ਕੋਈ ਕਾਰਵਾਈ ਨਹੀਂ ਹੁੰਦੀ ?**

a) ਇਕ ਅਵੈਧ ਫੰਕਸ਼ਨ (function) ਹੁੰਦਾ ਹੈ। b) ਵਾਕ-ਰਚਨਾ-ਗਲਤੀ (syntax error) ਪੈਦਾ ਕਰਦਾ ਹੈ
c) ਆਗਿਆ ਦਿੱਤੀ ਜਾਂਦੀ ਹੈ ਅਤੇ ਇਸਨੂੰ ਡੰਮੀ ਫੰਕਸ਼ਨ ਕਿਹਾ ਜਾਂਦਾ ਹੈ d) ਜ਼ੀਰੋ ਵਾਪਸ ਕਰਦਾ ਹੈ।

1.3 **ਫੰਕਸ਼ਨ (function) ਵਿਚ ਮੁੱਢਲੀ ਡਾਟਾ ਟਾਈਪ ਹੁੰਦੀ ਹੈ ?**

a) void b) int c) float d) char

1.4 **ਸੀ (C) ਵਿਚ ਪੈਰਾਮੀਟਰ ਪਾਸਿੰਗ (passing) ਯੰਤਰੀਕਰਨ (mechanism) ਕੀ ਹੁੰਦਾ ਹੈ ?**

a) call by name b) call by value c) name call d) None of these

1.5 **ਇਨ੍ਹਾਂ ਵਿਚੋਂ ਮੇਨ ਫੰਕਸ਼ਨ ਕਿਹੜਾ ਹੈ ?**

a) a built-in function b) a user define function c) optional d) all of these

ਪ੍ਰਸ਼ਨ 2: ਦੱਸੋ ਸਹੀ ਹੈ ਜਾਂ ਗਲਤ

2.1 ਕੀ ਹਰ ਫੰਕਸ਼ਨ ਦੇ ਵਿਚ ਇਕ ਰਿਟਰਨ ਸਟੇਟਮੈਂਟ ਹੋਣੀ ਜਰੂਰੀ ਹੁੰਦੀ ਹੈ ?
2.2 ਇਕ ਫੰਕਸ਼ਨ ਦੇ ਵਿਚ ਇਕ ਤੋਂ ਵੱਧ ਰਿਟਰਨ ਸਟੇਟਮੈਂਟ ਹੋ ਸਕਦੀਆਂ ਹਨ ?
2.3 ਕੀ ਅਸੀਂ ਕਾਂਸਟੈਂਟ ਨੂੰ ਫੰਕਸ਼ਨ ਵਿਚ ਪਾਸਡ (passed) ਕਰ ਸਕਦੇ ਹਾਂ ?
2.4 ਫੰਕਸ਼ਨ ਇਕ ਤੋਂ ਵਧੇਰੇ ਮੁੱਲ ਵਾਪਸ ਭੇਜਦਾ ਹੈ ਜੋ ਕਿ ਵਾਪਸ ਵਿਵਰਣ ਦੁਆਰਾ ਹੁੰਦਾ ਹੈ ?
2.5 ਇਕ ਫੰਕਸ਼ਨ ਤੋਂ ਵਾਪਸ ਮੁੜਨ ਲਈ ਤੁਹਾਨੂੰ ਇਕ ਫੰਕਸ਼ਨ ਵਿਚ ਵਾਪਸ ਵਿਵਰਣ ਵਰਤਣਾ ਚਾਹੀਦਾ ਹੈ?

ਪ੍ਰਸ਼ਨ 3: ਨਿਮਨਲਿਖਤ ਲਈ ਉਚਿੱਤ ਸ਼ਬਦ ਜਾਂ ਪਰਿਣਾਮ ਲਿਖੋ

3.1 ਫੰਕਸ਼ਨ ਪੂਰਵ ਪਰਿਭਾਸ਼ਤ ਹੁੰਦੇ ਹਨ ਅਤੇ ਸੰਗ੍ਰਹਿ ਦੇ ਨਾਲ ਦਿੱਤੇ ਜਾਂਦੇ ਹਨ ?
3.2 ਫੰਕਸ਼ਨ ਜਿਹੜੇ ਹਮੇਸ਼ਾਂ ਹੋਂਦ ਵਿਚ ਹੁੰਦੇ ਹਨ ਅਤੇ called ਪਹਿਲੇ ਫੰਕਸ਼ਨ ਨਹੀਂ ਹੁੰਦੇ ?
3.3 ਫੰਕਸ਼ਨ ਡਿਕਲੇਅਰੇਸ਼ਨ _______ ਦੇ ਨਾਲ ਖਤਮ ਹੁੰਦੀ ਹੈ ?
3.4 ਇਕ ਫੰਕਸ਼ਨ ਜਿਹੜਾ ਕੁਝ ਵੀ ਵਾਪਸ ਨਹੀਂ ਕਰਦਾ ਉਸ ਦੀ return ਟਾਈਪ ਕੀ ਹੁੰਦੀ ਹੈ ?
3.5 ਪੈਰਾਮੀਟਰ ਭੇਜਣ ਲਈ ਇਕ ਫੰਕਸ਼ਨ ਦੁਆਰਾ ਕਿਹੜਾ ਸਾਧਨ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ?

ਪ੍ਰਸ਼ਨ 4: ਪ੍ਰਸ਼ਨਾਂ ਦੇ ਉੱਤਰ ਦਿਓ

4.1 ਆਰਗੂਮੈਂਟ (argument) ਅਤੇ ਪੈਰਾਮੀਟਰ ਦੇ ਵਿਚ ਫਰਕ ਦੱਸੋ ?
4.2 ਫੰਕਸ਼ਨ (function) ਡਿਕਲੇਅਰੇਸ਼ਨ ਦੀ ਕੀ ਲੋੜ ਹੁੰਦੀ ਹੈ ?
4.3 ਫੰਕਸ਼ਨ (function) ਪਰਿਭਾਸ਼ਾ ਫੰਕਸ਼ਨ (function) ਡਿਕਲੇਅਰੇਸ਼ਨ ਤੋਂ ਕਿਵੇਂ ਭਿੰਨ ਹੁੰਦੀ ਹੈ ?
4.4 return ਸਟੇਟਮੈਂਟ ਕਿਹੜੀ ਭੂਮਿਕਾ ਨਿਭਾਉਂਦਾ ਹੈ ?
4.5 ਫੰਕਸ਼ਨ (function) ਦੀ execution ਕਦੋਂ ਸਮਾਪਤ ਹੁੰਦੀ ਹੈ ?
4.6 ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਦੇ ਲਾਭ ਦੱਸੋ ?
4.7 ਨਿਮਨਲਿਖਤ ਫੰਕਸ਼ਨ ਡਿਕਲੇਅਰੇਸ਼ਨ ਦੀ ਵਾਕ ਰਚਨਾ ਵਿਚ ਕਿਹੜੀ ਗਲਤੀ ਹੈ ?

```
void fun1(int parm1 ,float parm2);
{.......
.......
...}
```

4.8 ਨਿਮਨਲਿਖਿਤ ਫੰਕਸ਼ਨ ਪਰਿਭਾਸ਼ਾ ਵਿਚ ਕੀ ਗਲਤ ਹੈ

```
test_function (int k)
{
 float temp = 5.34
 temp = k / 2.0 ;
 return temp;
}
```

ਪ੍ਰਸ਼ਨ 5: ਨਿਮਨਲਿਖਤ ਕੋਡ ਫੰਕਸ਼ਨ ਦੇ ਸ਼ੁਰੂ ਵਿਚ ਦਿੱਤਾ ਹੈ। ਜੇਕਰ ਕੋਡ ਵਿਚ ਕੋਈ ਗਲਤੀਆਂ ਹਨ ਤਾਂ ਉਨ੍ਹਾਂ ਦੀ ਸ਼ਨਾਖਤ ਕਰੋ ?

```
static int count [ ] = {10 ,15, 20, 30;
float value;
```

ਪ੍ਰਸ਼ਨ 6 : ਨਿਮਨਲਿਖਤ ਵਿਚੋਂ ਗਲਤੀਆਂ ਲੱਭੋ

a)
```
int g (void)
{
printf("Inside this book" C \n" );
int h (void)
{  printf("Inside function h\n");}
}
```

b)
```
int sum(int x ,int y)
{
 int result ;
 result =x + y;
}
```

c)
```
int sum(int n)
{ If ( n ==0 )
  return 0 ;
 else
 n + sum(n-1); }
```

d)
```
void f (float a) ;
{
 float a;
 print f(%f" ,a);
}
```

e)
```
void product (void)
{
int a, b, c ,result;
printf( "Enter three integers : ")
scanf("%d %d %d" ,&a ,&b ,&c );
result = a*b*c ;
printf( "Result is %d" , result);
return result ;
}
```

f)
```
int sum (int  n)
{ if (n = = 0)
  return 0;
 else
   return n +sum (n); }
```

ਪ੍ਰਸ਼ਨ 7: ਨਿਮਨਲਿਖਤ ਫੰਕਸ਼ਨਜ਼ ਵਿਚੋਂ ਹਰੇਕ ਦੇ ਫੰਕਸ਼ਨ (function) ਹੈਡਰ ਦਿਓ

7.1 ਫੰਕਸ਼ਨ (function) average_marks ਜੋ ਦੋਹਰੀ ਸ਼ੁਧਤਾ ਵਾਲੇ ਫਲੋਟਿੰਗ (floating) ਅੰਕ marks1,marks2 ਨੂੰ input ਕਰਦੇ ਹਨ ਅਤੇ ਇਕ ਦੋਹਰੀ ਸ਼ੁਧਤਾ ਵਾਲਾ ਫਲੋਟਿੰਗ (floating) ਅੰਕ ਵਾਪਸ ਕਰਦੇ ਹਨ ?

Ans: double average_marks(double marks1 ,marks2)

7.2 ਫੰਕਸ਼ਨ (function) smallest ਜਿਹੜਾ ਤਿੰਨ ਪੂਰਨ ਅੰਕ ਲੈਂਦਾ ਹੈ **x, y, z** ਅਤੇ ਪਰਿਣਾਮ ਵਾਪਸ ਮੋੜਦਾ ਹੈ ?

Ans int smallest (int x ,int y ,int z)

7.3 ਫੰਕਸ਼ਨ (function) instructions ਜਿਹੜਾ ਕੋਈ ਆਰਗੂਮੈਂਟ ਪ੍ਰਾਪਤ ਨਹੀ ਕਰਦਾ ਅਤੇ ਨਾ ਹੀ ਮੁੱਲ ਮੋੜਦਾ ਹੈ (ਅਜਿਹਾ ਇਕ ਫੰਕਸ਼ਨ ਆਮ ਤੌਰ ਤੇ ਇਕ ਯੂਜ਼ਰ ਨੂੰ ਹਦਾਇਤਾਂ ਦਾ ਪ੍ਰਦਰਸ਼ਨ ਕਰਦਾ ਹੈ) ?

Ans void instructions (void)

7.4 ਨਿਮਨਲਿਖਤ ਵਿਚੋਂ ਹਰੇਕ ਲਈ ਇਕ ਡਿਕਲੇਅਰੇਸ਼ਨ ਲਿਖੋ ?
ਪੂਰਨ-ਅੰਕ (integral) count ਜਿਹੜੀ ਇਕ ਰਜਿਸਟਰ (register) ਵਿਚ ਸੰਭਾਲੀ ਜਾਣੀ ਚਾਹੀਦੀ ਹੈ, count ਨੂੰ 0 ਤੋਂ ਆਰੰਭ ਕਰਦੀ ਹੈ ?

Ans Register int count = 0 ;

Answer to objective questions

Q	.1	.2	.3	.4	.5
Q1	b	C	B	b	b
Q2	F	T	T	F	T
Q3	built-in- function	main ( )	Semicolon	void	call by value

ਅਧਿਆਇ

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ (ਭਾਗ – 1)

Windows Movie Maker (Part – 1)

4

ਭੂਮਿਕਾ

ਤੁਸੀ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ (windows movie maker) ਦੀ ਵਰਤੋ ਨਾਲ ਕੈਮਰੇ ਦੁਆਰਾ ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ (audio & video) ਜਾਂ ਤਸਵੀਰਾਂ ਨੂੰ ਵੀ ਆਪਣੀ ਮੂਵੀ ਬਨਾਉਣ ਲਈ ਕੰਪਿਊਟਰ ਵਿਚ ਭਰ ਸਕਦੇ ਹੋ। ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ ਨੂੰ ਐਡਿਟ, ਜਿਵੇਂ ਕਿ ਟਾਈਟਲ ਲਿਖਣਾ, ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਜਾਂ ਇਫੈਕਟ (effect) ਭਰਨ ਮਗਰੋਂ ਤੁਸੀਂ ਆਪਣੀ ਮੂਵੀ ਨੂੰ ਸੇਵ ਕਰ ਸਕਦੇ ਹੋ।

ਤੁਸੀ ਤਿਆਰ ਕੀਤੀ ਮੂਵੀ ਨੂੰ ਆਪਣੇ ਕੰਪਿਊਟਰ ਵਿਚ ਜਾਂ ਸੀ ਡੀ ਵਿਚ ਵੀ ਰੱਖ ਸਕਦੇ ਹੋ। ਕੰਪਿਊਟਰ ਨਾਲ ਜੋੜਨ ਲਈ ਜੇਕਰ ਤੁਹਾਡੇ ਕੋਲ ਡਿਜ਼ੀਟਲ ਵੀਡੀਓ (Digital Video) ਕੈਮਰਾ ਹੈ ਤਾਂ ਤੁਸੀ ਆਪਣੀ ਮੂਵੀ ਨੂੰ ਟੇਪ ਤੇ ਰਿਕਾਰਡ ਕਰ ਸਕਦੇ ਹੋ ਅਤੇ ਮਗਰੋ ਟੈਲੀਵਿਜ਼ਨ ਜਾਂ ਡਿਜ਼ੀਟਲ ਵੀਡੀਓ ਕੈਮਰੇ ਨਾਲ ਦੇਖ ਸਕਦੇ ਹੋ।

ਫਾਈਲ ਕਿਸਮਾਂ

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਭਰੇ ਗਏ ਵਿਸ਼ਿਆਂ ਨੂੰ ਵਿੰਡੋਜ਼ ਮੀਡਿਆ ਫਾਰਮੇਟ ਵਿਚ ਸੇਵ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।

ਤੁਸੀ ਹੇਠਾਂ ਲਿਖੀਆਂ ਫਾਈਲਾਂ ਦੀ ਕਿਸਮਾਂ ਨੂੰ ਮੂਵੀ ਮੇਕਰ ਦੁਆਰਾ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਵਰਤ ਸਕਦੇ ਹੋ :

- **ਆਡੀਓ ਫਾਈਲਜ਼ (Audio files):** .aif, .aifc, .aiff, .asf, .au, .mp2, .mp3, .mpa, .snd, .wav, and .wma
- **ਪਿਕਚਰ ਫਾਈਲਜ਼ (Picture files):** .bmp, .dib, .emf, .gif, .jfif, .jpe, .jpeg, .jpg, .png, .tif, .tiff, and .wmf
- **ਵੀਡੀਓ ਫਾਈਲਜ਼ (Video files):** .asf, .avi, .m1v, .mp2, .mp2v, .mpe, .mpeg, .mpg, .mpv2, .wm, & .wmv

ਕੋਲੈਕਸ਼ਨਜ਼, ਪ੍ਰੋਜੈਕਟ ਅਤੇ ਮੂਵੀਜ਼ ਨੂੰ ਜਾਨਣਾ (knowing Collections, Projects & Movies) :

ਯੂਜ਼ਰ ਇੰਟਰਫੇਸ (interface) ਅਤੇ ਹੈਲਪ (help) ਵਿਚ ਕੋਲੈਕਸ਼ਨਜ਼, ਪ੍ਰੋਜੈਕਟਸ ਅਤੇ ਮੂਵੀ ਦੀ ਵਰਤੋਂ ਭਰਪੂਰ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਹੇਠਾਂ ਲਿਖੇ ਵਿਸ਼ੇ ਇਨ੍ਹਾਂ ਵੱਲ ਚਾਨਣਾ ਪਾਉਂਦੇ ਹਨ:

ਕੋਲੈਕਸ਼ਨ (Collection)

ਕੋਲੈਕਸ਼ਨਜ਼ ਵਿਚ ਆਡੀਓ ਕਲਿਪਸ (audio clips), ਵੀਡੀਓ ਕਲਿਪਸ (video clips) ਜਾਂ ਪਿਕਚਰ (picture) ਆਦਿ ਸ਼ਾਮਿਲ ਹੁੰਦੇ ਹਨ। ਕੋਲੈਕਸ਼ਨਜ਼ ਇਕ ਸਮੂਹ ਭੰਡਾਰ ਦਾ ਕੰਮ ਕਰਦਾ ਹੈ ਜਿਸ ਵਿਚ ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ ਦੇ ਛੋਟੇ-2 ਕਲਿਪਸ ਸ਼ਾਮਿਲ ਹੁੰਦੇ ਹਨ। ਕੋਲੈਕਸ਼ਨਜ਼ ਨੂੰ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੀ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ ਦੇਖਿਆ ਜਾ ਸਕਦਾ ਹੈ।

ਪ੍ਰੋਜੈਕਟ (Project)

ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ ਕਲਿਪਸ ਦੇ ਸਮੇਂ (time) ਸੂਚਨਾ ਦੀ ਵਿਵਸਥਾ ਹੁੰਦੀ ਹੈ। ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੁਆਰਾ ਸੇਵ ਕੀਤੀ ਗਈ ਪ੍ਰੋਜੈਕਟ ਫਾਈਲ ਦੀ ਐਕਸਟੈਨਸ਼ਨ (extension) .msv mm ਹੁੰਦੀ ਹੈ। ਆਪਣੀ ਫਾਈਲ ਨੂੰ ਸੇਵ ਕਰਨ ਨਾਲ ਤੁਸੀ ਇਸ ਫਾਈਲ ਨੂੰ ਮੁੜ ਕੰਮ ਕਰਨ ਲਈ ਖੋਲ੍ਹ ਸਕਦੇ ਹੋ।

ਮੂਵੀ (Movie)

ਮੂਵੀ ਨੂੰ ਇਕ ਅੰਤਿਮ ਸੇਵ ਕੀਤੀ ਗਈ ਪ੍ਰੋਜੈਕਟ ਫਾਈਲ ਕਿਹਾ ਜਾ ਸਕਦਾ ਹੈ। ਤੁਸੀਂ ਆਪਣੀ ਮੂਵੀ ਨੂੰ ਕੰਪਿਊਟਰ, ਜਾਂ ਸੀ ਡੀ ਵਿਚ ਸੇਵ ਕਰ ਸਕਦੇ ਹੋ ਅਤੇ ਈ-ਮੇਲ ਰਾਹੀ ਵੀ ਭੇਜ ਸਕਦੇ ਹੋ। ਸੇਵ ਕੀਤੀ ਗਈ ਮੂਵੀ ਨੂੰ ਦੇਖਣ ਲਈ ਵਿੰਡੋਜ਼ ਮੀਡੀਆ ਪਲੇਅਰ (Player) ਜਾਂ ਵੈੱਬ ਬਰੋਜ਼ਰ ਦੀ ਵਰਤੋਂ ਕੀਤੀ ਜਾ ਸਕਦੀ ਹੈ।

ਸੋਰਸ ਫਾਈਲ (Source File)

ਇਨ੍ਹਾਂ ਨੂੰ ਡਿਜ਼ੀਟਲ ਮੀਡੀਆ ਫਾਈਲ (ਜਿਵੇ ਕੀ ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ ਫਾਈਲ) ਜਾਂ ਪਿਕਚਰਜ਼ ਫਾਈਲ ਵੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਜਦੋਂ ਤੁਸੀ ਇਕ ਵੀਡੀਓ, ਆਡੀਓ ਜਾਂ ਪਿਕਚਰ ਫਾਈਲ ਨੂੰ ਇਮਪੋਰਟ (import) ਕਰਦੇ ਹੋ, ਤਾਂ ਇਹ ਫਾਈਲ ਆਪਣੀ ਅਸਲੀ ਸਥਾਨ ਤੇ ਸੁਰਖਿਅਤ ਰਹਿੰਦੀ ਹੈ। ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਸ਼ਾਮਿਲ ਕੀਤੀ ਫਾਈਲ, ਅਸਲੀ ਫਾਈਲ ਦਾ ਵਰਣਨ (representation) ਹੁੰਦੀ ਹੈ। ਇਹ ਫਾਈਲ ਸੋਰਸ ਫਾਈਲ ਦੀ ਕਾਪੀ ਨਹੀ ਹੁੰਦੀ। ਦੂਸਰੇ ਸ਼ਬਦਾਂ ਵਿਚ, ਜਦੋਂ ਸੋਰਸ ਫਾਈਲ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਸ਼ਾਮਲ ਕੀਤੀ ਜਾ ਚੁੱਕੀ ਹੋਵੇ ਅਤੇ ਜੇਕਰ ਤੁਸੀ ਸੋਰਸ ਫਾਈਲ ਨੂੰ ਕਿਸੇ ਹੋਰ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਐਡਿਟ ਕਰਦੇ ਹੋ ਤਾਂ ਤੁਹਾਡੇ ਵੱਲੋ ਫਾਈਲ ਵਿਚ ਕੀਤੇ ਗਏ ਬਦਲਾਉ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਆਪਣੇ - ਆਪ ਹੀ ਨਜ਼ਰ ਆਉਣਗੇ। ਅਤੇ ਜੇਕਰ ਤੁਸੀ thumbnail ਜਾਂ ਕਲਿਪ ਨੂੰ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚੋਂ ਡਿਲੀਟ ਕਰਦੇ ਹੋ, ਤਾਂ ਸੋਰਸ ਫਾਈਲ ਆਪਣੇ ਅਗਲੇ ਸਥਾਨ ਤੇ ਸੁਰੱਖਿਅਤ ਰਹੇਗੀ।

ਵੀਡੀਓ ਕੈਪਚਰ ਡਿਵਾਇਸਿਜ਼ (Video capture devices)

ਵੀਡੀਓ caputre ਡਿਵਾਇਸਿਜ਼ ਦੁਆਰਾ ਤੁਸੀ live ਜਾਂ ਸੁਰਖਿਅਤ ਵੀਡੀਓ ਨੂੰ ਆਪਣੇ ਕੰਪਿਊਟਰ ਵਿਚ ਉਤਾਰ ਸਕਦੇ ਹੋ।

- ਐਨਾਲਾਗ ਵੀਡੀਓ ਸੋਰਸ ਜਿਵੇਂ ਕੀ ਐਨਾਲਾਗ ਕੈਮਰਾ ਜਾਂ Video Cassette Recorder (VCR) ਜਿਹੜੇ ਕਿ ਐਨਾਲਾਗ capture ਕਾਰਡ ਨਾਲ ਜੁੜੇ ਹੋਣ।
- ਵੈੱਬ ਕੈਮਰਾ (Web camera)
- ਟੀ ਵੀ ਟਿਊਨਰ ਕਾਰਡ (TV Tuner Card)

ਆਡੀਓ ਕੈਪਚਰ ਡਿਵਾਇਸਿਜ਼ (Audio capture devices)

ਇਨ੍ਹਾਂ ਡਿਵਾਈਸਿਜ਼ ਦੁਆਰਾ ਤੁਸੀ ਆਡੀਓ ਨੂੰ ਆਪਣੇ ਕੰਪਊਟਰ ਵਿਚ ਭਰ ਸਕਦੇ ਹੋ। ਮਾਈਕਰੋਫੋਨ ਨੂੰ ਆਡੀਓ ਕੈਪਚਰ ਡਿਵਾਈਸ ਦੀ ਮੁੱਖ ਮਿਸਾਲ ਕਿਹਾ ਜਾ ਸਕਦਾ ਹੈ। ਤੁਸੀ ਹੇਠ ਲਿਖੇ ਪ੍ਰਕਾਰ ਦੇ ਆਡੀਓ ਕੈਪਚਰ ਡਿਵਾਈਸ ਦੀ ਵਰਤੋ ਕਰ ਸਕਦੇ ਹੋ।

- ਆਡੀਓ ਕਾਰਡ
- ਮਾਈਕਰੋਫੋਨ
- ਐਨਾਲਾਗ ਜਾ ਵੈਬ ਕੈਮਰਾ ਵਿਚ ਸ਼ਾਮਲ ਮਾਇਕਰੋਫੋਨ

ਕੈਪਚਰ (capture) ਡਿਵਾਈਸ ਨੂੰ ਜੋੜਨ ਲਈ

ਕੈਪਚਰ (capture) ਡਿਵਾਈਸ ਦੀ ਸੂਚੀ ਹੇਠ ਲਿਖੇ ਅਨੁਸਾਰ ਹੈ। ਕਿਸੇ ਵੀ ਕੈਪਚਰ (capture) ਡਿਵਾਈਸ ਬਾਰੇ ਪੂਰੀ ਜਾਣਕਾਰੀ ਹਾਸਲ ਕਰਨ ਲਈ ਤੁਸੀ ਸਬੰਧਿਤ ਹਾਰਡਵੇਅਰ ਦੀ ਡਾਕੂਮੈਂਟੇਸ਼ਨ ਪੜ੍ਹ ਸਕਦੇ ਹੋ।

- **ਵੈੱਬ ਕੈਮਰਾ** ਦੀ ਕਿਸਮ ਉਪਰ ਅਧਾਰਿਤ ਤੁਸੀ ਇਸਨੂੰ ਯੂ ਐਸ ਬੀ (USB) ਪੋਰਟ ਜਾਂ ਐਨਾਲਾਗ capture ਕਾਰਡ ਨਾਲ ਜੋੜ ਸਕਦੇ ਹੋ।

- **ਐਨਾਲਾਗ ਕੈਮਰਾ ਜਾਂ ਵੀ ਸੀ ਆਰ (VCR) ਦਾ ਐਨਾਲਾਗ capture ਕਾਰਡ ਨਾਲ ਜੋੜ** : ਇਸ ਤਰ੍ਹਾਂ ਦੇ ਮੇਲ ਵਿਚ ਤੁਸੀ ਕੈਮਰਾ ਜਾਂ ਵੀ ਸੀ ਆਰ ਨੂੰ ਐਨਾਲਾਗ capture ਨਾਲ ਜੋੜ ਸਕਦੇ ਹੋ। ਮਿਸਾਲ ਲਈ, ਤੁਸੀ ਕੈਮਰਾ ਦੀ video ਲਾਈਨ ਆਊਟ (line-out) ਨੂੰ capture ਕਾਰਡ ਦੀ video ਲਾਈਨ ਇਨ (line-in) ਨਾਲ ਜੋੜ ਸਕਦੇ ਹੋ ।

- **ਡਿਜੀਟਲ ਵੀਡੀਓ ਕੈਮਰਾ ਜਾਂ ਵੀ ਸੀ ਆਰ ਦਾ IEEE1394 ਪੋਰਟ ਨਾਲ ਜੋੜ** : ਜਦੋਂ ਡਿਜੀਟਲ ਵੀਡੀਓ ਕੈਮਰਾ ਨੂੰ IEEE1394 ਪੋਰਟ ਨਾਲ ਜੋੜਿਆ ਜਾਂਦਾ ਹੈ ਤਾਂ ਤੁਸੀ ਉੱਤਮ ਕੁਆਲਿਟੀ ਨੂੰ ਪਾਉਂਦੇ ਹੋ ।

ਕਿਉਂਕਿ ਡਾਟਾ ਪਹਿਲਾਂ ਤੋਂ ਹੀ ਡਿਜ਼ੀਟਲ ਕਿਸਮ ਵਿਚ ਹੁੰਦਾ ਹੈ। ਅਤੇ ਇਸਨੂੰ IEEE1394 ਪੋਰਟ ਦੁਆਰਾ ਕੰਪਿਊਟਰ ਵਿਚ ਪ੍ਰਵੇਸ਼ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।

- **ਮਾਈਕਰੋਫੋਨ ਦਾ ਸਾਊਂਡ ਕਾਰਡ ਜਾਂ ਯੂ ਐਸ ਬੀ ਪੋਰਟ ਨਾਲ ਜੋੜ :** ਮਾਈਕਰੋਫੋਨ ਤੋਂ ਆਡੀਓ ਨੂੰ ਕੰਪਿਊਟਰ ਤੇ ਲਿਆਉਣ ਲਈ, ਤੁਹਾਨੂੰ ਮਾਈਕਰੋਫੋਨ ਨੂੰ ਕੰਪਿਊਟਰ ਦੀ line-in input ਨਾਲ ਜੋੜਣਾ ਪਵੇਗਾ। ਕੁਝ ਮਾਈਕਰੋਫੋਨ ਯੂ ਐਸ ਬੀ ਪੋਰਟ ਨਾਲ ਵੀ ਜੁੜ ਜਾਂਦੇ ਹਨ।

- ਟੀ ਵੀ ਟਿਊਨਰ ਕਾਰਡ ਟੀ ਵੀ ਤੋਂ ਵੀਡੀਓ ਹਾਸਿਲ ਕਰਨ ਲਈ ਤੁਹਾਨੂੰ ਟੀ ਵੀ ਟਿਊਨਰ ਕਾਰਡ ਨੂੰ ਕੰਪਿਊਟਰ ਨਾਲ ਜੋੜਨ ਦੀ ਜ਼ਰੂਰਤ ਪਵੇਗੀ।

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਇੰਟਰਫੇਸ (Windows Movie Maker Interface)

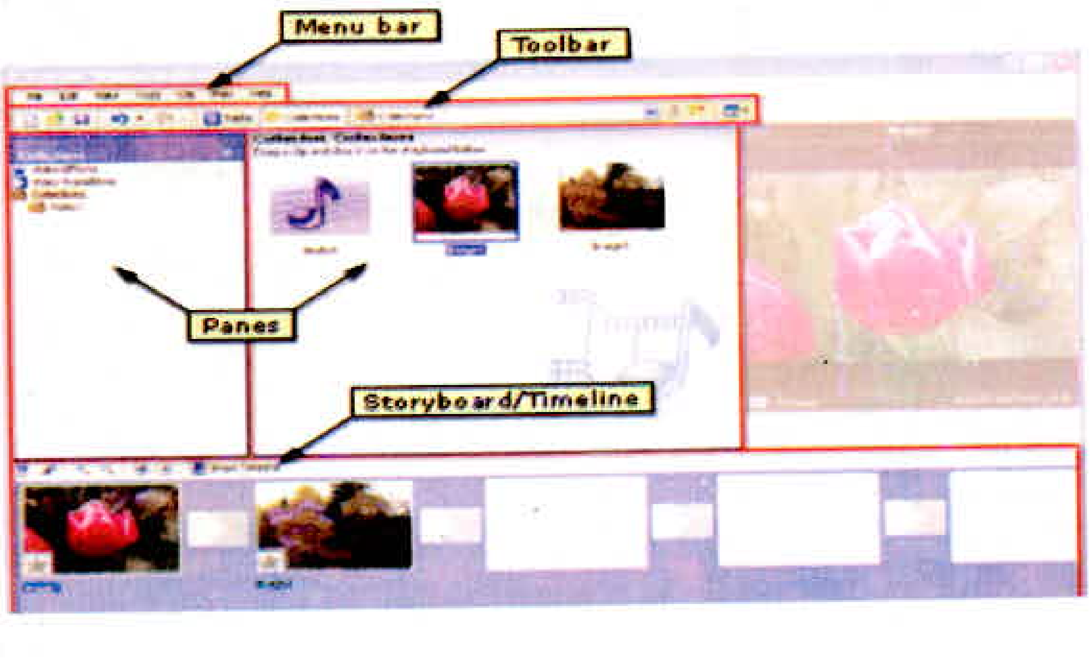
ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੇ ਇੰਟਰਫੇਸ ਨੂੰ ਤਿੰਨ ਭਾਗਾਂ ਵਿਚ ਵੰਡਿਆ ਗਿਆ ਹੈ। ਮੀਨੂੰ-ਬਾਰ (menubar) ਅਤੇ ਟੂਲ-ਬਾਰ (toolbar), ਪੇਨਸ (panes) ਅਤੇ ਸਟੋਰੀ-ਬੋਰਡ (story board) ਜਾਂ ਟਾਈਮ-ਲਾਈਨ (timeline)

- **ਮੀਨੂੰ-ਬਾਰ ਅਤੇ ਟੂਲ ਬਾਰ:** ਇਹ ਤੁਹਾਨੂੰ ਮੀਨੂੰ-ਬਾਰ ਅਤੇ ਟੂਲ ਬਾਰ ਨੂੰ ਵਰਤਣ ਦੀ ਜਾਣਕਾਰੀ ਦਿੰਦਾ ਹੈ।

- **ਪੇਨਸ :** ਇਹ ਤੁਹਾਨੂੰ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਬਾਰੇ ਜਾਣਕਾਰੀ ਦਿੰਦਾ ਹੈ। ਇਹ ਤੁਹਾਨੂੰ ਪ੍ਰੋਜੈਕਟ ਐਡਿਟ ਅਤੇ ਮੂਵੀ ਬਣਾਉਣ ਵਰਗੇ ਕੰਮਾਂ ਦੀ ਵੀ ਜਾਣਕਾਰੀ ਦਿੰਦਾ ਹੈ।

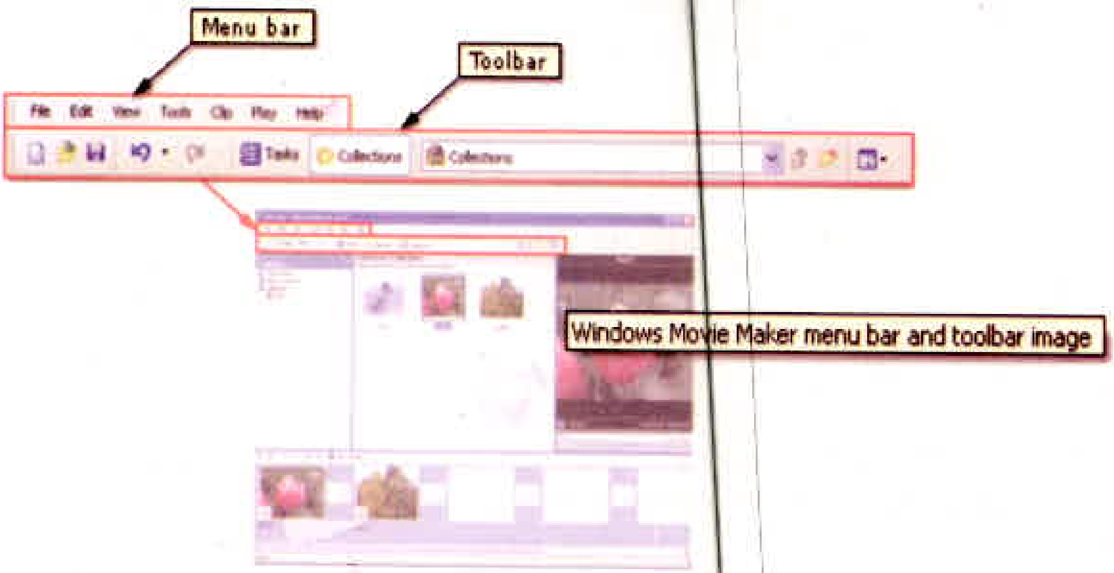
- **ਸਟੋਰੀ-ਬੋਰਡ ਜਾਂ ਟਾਈਮ-ਲਾਈਨ (Story Board/Time Line):** ਇਹ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਕਲਿਪਸ ਨਾਲ ਕੰਮ ਕਰਨ ਦੀ ਸੂਚਨਾ ਪ੍ਰਦਾਨ ਕਰਦੀ ਹੈ।

- ਹੇਠਾਂ ਦਿਤੀ ਮਿਸਾਲ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੇ ਯੂਜ਼ਰ ਇੰਟਰਫੇਸ ਨੂੰ ਵਿਸਥਾਰ ਨਾਲ ਦਰਸਾਉਂਦੀ ਹੈ।



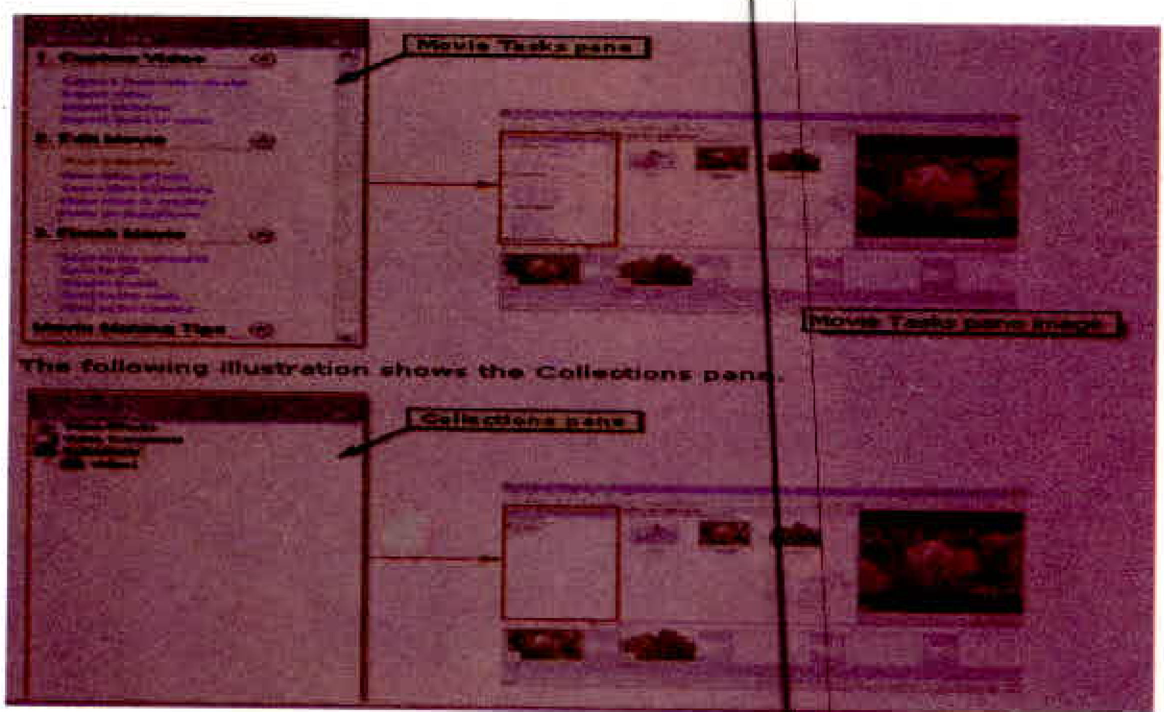
ਮੀਨੂੰ-ਬਾਰ ਅਤੇ ਟੂਲ-ਬਾਰ (Menu bar and Toolbar)

ਤੁਸੀ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਨਾਲ ਕੰਮ ਕਰਨ ਲਈ, ਮੀਨੂੰ-ਬਾਰ ਵਿਚ ਦਰਸਾਈਆਂ ਕਮਾਂਡਜ਼ ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਦੇ ਹੋ। ਟੂਲ-ਬਾਰ ਤੁਹਾਨੂੰ ਮੀਨੂੰ ਤੋ ਕਮਾਂਡ ਸਿਲੈਕਟ ਕਰਨ ਦਾ ਸਹੀ ਤਰੀਕਾ ਪ੍ਰਦਾਨ ਕਰਦਾ ਹੈ। ਤੁਸੀਂ ਟੂਲ-ਬਾਰ ਦੀ ਮੱਦਦ ਨਾਲ ਆਮ ਕੰਮਾਂ ਨੂੰ ਤੇਜ਼ੀ ਨਾਲ ਕਰ ਸਕਦੇ ਹੋ। ਟੂਲ-ਬਾਰ ਨੂੰ ਦਿੱਖ ਜਾਂ ਅਦਿੱਖ ਕਰਨ ਲਈ, ਵਿਊ ਮੀਨੂੰ ਤੇ ਟੂਲ-ਬਾਰ ਕਲਿੱਕ ਕਰੋ। ਹੇਠਾਂ ਦਿਤੀ ਮਿਸਾਲ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੀ ਮੀਨੂੰ ਬਾਰ ਅਤੇ ਟੂਲ-ਬਾਰ ਨੂੰ ਦਰਸਾਉਂਦੀ ਹੈ।



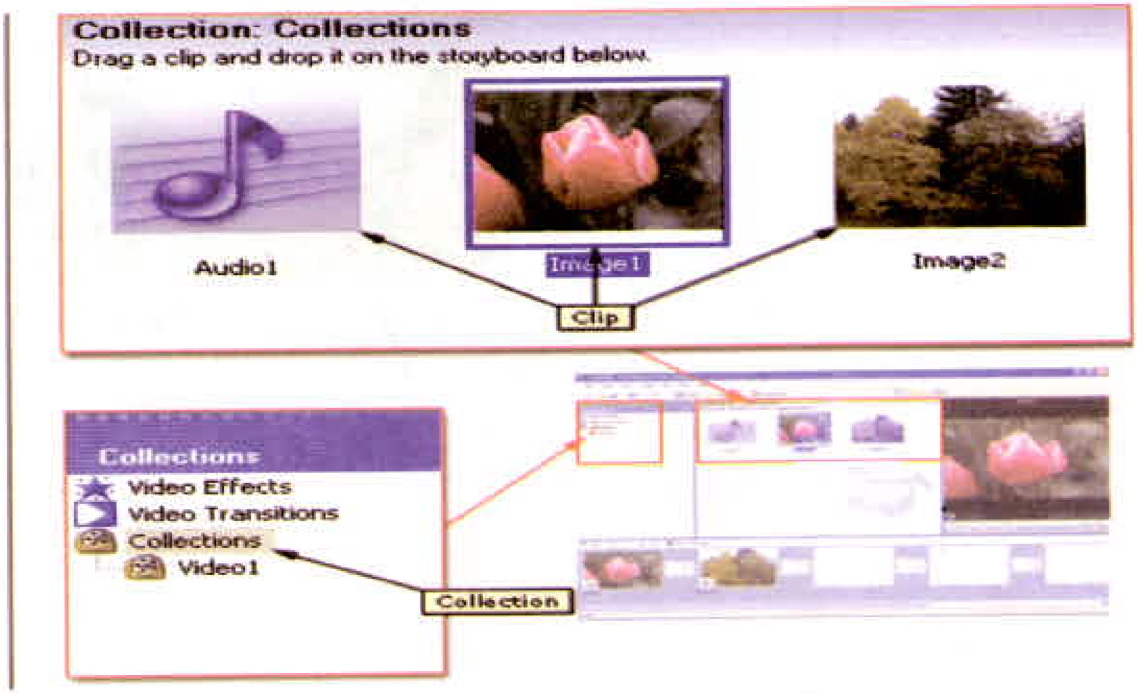
ਪੇਨਜ਼ (The panes)

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੇ ਮੁੱਖ ਤੱਤ ਅਲੱਗ ਅਲੱਗ ਪੇਨਜ਼ ਵਿਚ ਸਥਿਤ ਹੁੰਦੇ ਹਨ। ਹੇਠਾਂ ਦਿੱਤੀ ਮਿਸਾਲ ਮੂਵੀ ਟਾਸਕ ਪੇਨ **(movie task pane)** ਨੂੰ ਦਰਸਾਉਂਦੀ ਹੈ



ਕੋਲੈਕਸ਼ਨ ਪੇਨ (collection pane)

ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ ਤੁਸੀਂ ਕਲਿਪ ਦੇ ਸਮੂਹ ਦੇਖ ਸਕਦੇ ਹੋ। ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਦੇ ਖੱਬੇ ਪਾਸੇ ਤੁਸੀਂ ਕੋਲੈਕਸ਼ਨਜ਼ ਦੀ ਮੂਵੀ ਦੇਖ ਸਕਦੇ ਹੋ ਅਤੇ ਚੁਣੇ ਗਏ ਕਲਿਪਸ ਤੁਸੀਂ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਦੇ ਸੱਜੇ ਪਾਸੇ ਦੇਖ ਸਕਦੇ ਹੋ ਜਿਵੇਂ ਕਿ ਹੇਠਾਂ ਦਿਖਾਇਆ ਗਿਆ ਹੈ।



ਤੁਸੀਂ ਕਲਿਪ ਨੂੰ ਕਾਨਟੈਂਟ ਪੇਨ ਜਾਂ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਤੋਂ ਚੁਣਕੇ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਦੀ ਸਟੋਰੀ ਬੋਰਡ / ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਰੱਖ ਸਕਦੇ ਹੋ। ਕਲਿਪ ਨੂੰ ਮੋਨੀਟਰ ਤੇ ਦੇਖਣ ਲਈ ਸਿੱਧਾ ਮੋਨੀਟਰ ਵਿਚ ਲੈ ਜਾ ਸਕਦੇ ਹੋ। ਤੁਹਾਡੀ ਕਲਿਪ ਅਸਲ ਸੋਰਸ ਫਾਈਲ ਨੂੰ ਹੀ ਦਰਸਾਉਂਦੀ ਹੈ। ਜੇਕਰ ਤੁਸੀਂ ਕਲਿਪ ਵਿਚ ਕੋਈ ਬਦਲਾਉ ਕਰਦੇ ਹੋ, ਇਹ ਬਦਲਾਉ ਚਲ ਰਹੇ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਹੀ ਨਜ਼ਰ ਆਉਣਗੇ, ਇਸ ਨਾਲ ਸੋਰਸ ਫਾਈਲ ਵਿਚ ਕੋਈ ਅੰਤਰ ਨਹੀਂ ਆਏਗਾ।

ਕਾਨਟੈਂਟਸ ਪੇਨ (Contents pane)

ਕਾਨਟੈਂਟ ਪੇਨ ਸਿਰਫ਼ ਉਹੀ ਕਲਿਪਸ ਹੀ ਦਰਸਾਉਂਦੀਆਂ ਹਨ ਜੋ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਦੁਆਰਾ ਚੁਣੇ ਗਏ ਹੋਣ। ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ ਤੁਸੀਂ ਸਾਰੇ ਵੀਡੀਓ, ਆਡੀਓ, ਪਿਕਚਰ ਆਦਿ ਦੇਖ ਸਕਦੇ ਹੋ ਜਿਹੜੇ ਸਟੋਰੀ ਬੋਰਡ / ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਰੱਖਣੇ ਹੋਣ।

ਡੀਟੇਲਜ਼ (Details)

ਇਹ ਕਾਨਟੈਂਟਸ ਵਿਚ ਦਰਸਾਏ ਗਏ ਕਲਿਪ ਬਾਰੇ ਜਾਣਕਾਰੀ ਨੂੰ ਦਰਸਾਉਂਦਾ ਹੈ। ਵਿਊ ਵਿਚ ਤੁਸੀਂ item ਨੂੰ sort ਕਰਨ ਲਈ ਕਾਲਮ ਤੇ ਕਲਿਕ ਕਰ ਸਕਦੇ ਹੋ।

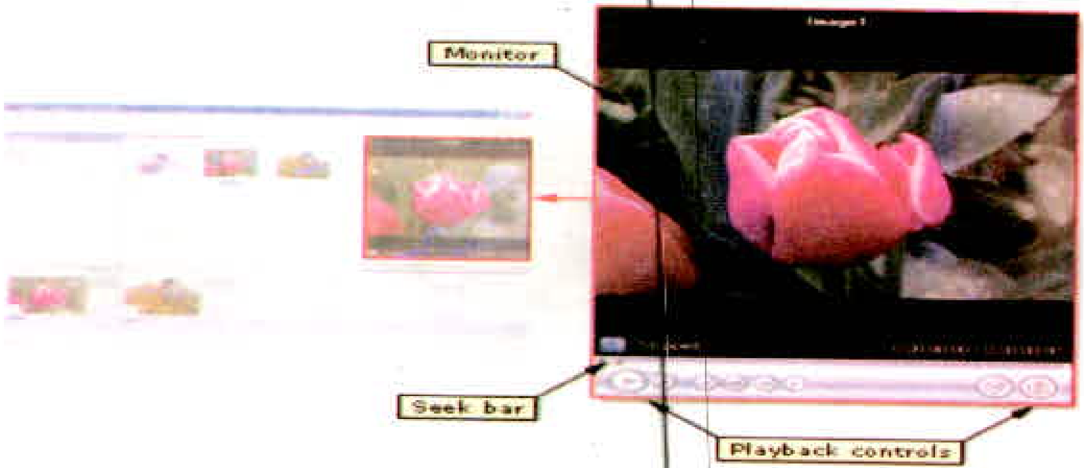
ਥੰਬ-ਨੇਲ (Thumbnail)

ਇਹ ਤੁਹਾਨੂੰ ਚੁਣੀ ਹੋਈ ਕਲਿਪ ਦੀ ਛੋਟੀ ਤਸਵੀਰ ਦਰਸਾਉਂਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਜੇਕਰ ਕੋਲੈਕਸ਼ਨਜ਼ ਵਿਚ imparted ਵੀਡੀਓ ਹੋਣ ਤਾਂ ਥੰਬ-ਨੇਲ ਤੁਹਾਨੂੰ ਚੁਣੇ ਹੋਏ ਵੀਡੀਓ ਤੇ ਅਧਾਰਿਤ image ਦਰਸਾਏਗੀ। ਜੇਕਰ ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ ਤਸਵੀਰ ਹੋਏ ਤਾਂ ਥੰਬ-ਨੇਲ ਉਸ ਪਿਕਚਰ ਨੂੰ ਦਰਸਾਏਗੀ।

ਮੋਨੀਟਰ (Monitor)

ਤੁਸੀਂ ਮੋਨੀਟਰ ਦੀ ਵਰਤੋਂ ਇੱਕਲੇ ਕਲਿਪ ਜਾਂ ਪੂਰੀ ਮੂਵੀ ਨੂੰ ਦੇਖਣ ਲਈ ਕਰ ਸਕਦੇ ਹੋ। ਮੋਨੀਟਰ ਦੀ ਵਰਤੋਂ ਨਾਲ ਤੁਸੀਂ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਮੂਵੀ ਸੇਵ ਕਰਨ ਤੋਂ ਪਹਿਲਾਂ ਦੇਖ ਸਕਦੇ ਹੋ। ਤੁਸੀਂ ਪਲੇਅ-ਬੈਕ (playback) ਕੰਟਰੋਲ ਦੀ ਮਦਦ ਨਾਲ ਸਾਰੇ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਨੇਵੀਗੇਟ (navigate) ਕਰ ਸਕਦੇ ਹੋ, ਮੋਨੀਟਰ ਤੇ ਬਣੇ ਬਟਨ ਦੀ ਮਦਦ

ਨਾਲ ਇਕ ਵੀਡੀਓ ਨੂੰ ਦੋ ਭਾਗਾਂ ਵਿਚ ਵੰਡ ਸਕਦੇ ਹੋ। ਮੋਨੀਟਰ ਅਤੇ ਇਸਦੇ ਸੰਬੰਧਤ ਬਟਨ ਹੇਠ ਲਿਖੇ ਚਿੱਤਰ ਵਿਚ ਦਿਖਾਏ ਗਏ ਹਨ :

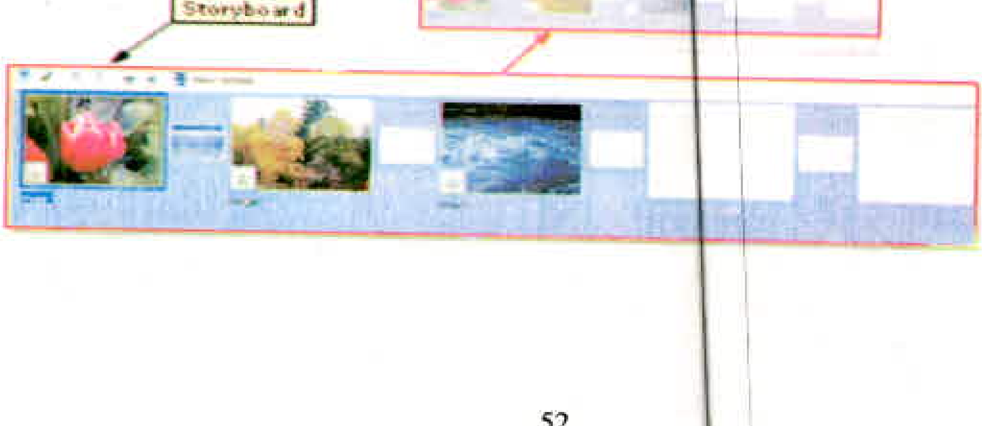


ਸਟੋਰੀ ਬੋਰਡ (Story Board)

ਇਹ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦਾ ਡਿਫਾਲਟ (default) ਵਿਊ ਹੈ। ਤੁਸੀਂ ਸਟੋਰੀ ਬੋਰਡ ਦੀ ਮਦਦ ਨਾਲ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਕਲਿਪ ਦੀ ਸ਼੍ਰੇਣੀ ਨੂੰ ਦੇਖ ਸਕਦੇ ਹੋ ਅਤੇ ਕਲਿਪ ਨੂੰ ਅਸਾਨੀ ਨਾਲ ਨਵੀਂ ਤਰਤੀਬ ਵਿਚ ਪਾ ਸਕਦੇ ਹੋ। ਇਹ ਵਿਊ ਤੁਹਾਨੂੰ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਪਾਏ ਨਵੇਂ ਵੀਡੀਓ ਅਫੈਕਟ (effect) ਨੂੰ ਦੇਖਣ ਵਿਚ ਵੀ ਮਦਦ ਕਰਦਾ ਹੈ। ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਪਾਏ ਗਏ ਆਡੀਓ ਕਲਿਪਸ ਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਨਹੀਂ ਦੇਖ ਸਕਦੇ, ਇਹ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਦਿਖਾਈ ਦਿੰਦੇ ਹਨ।

ਸਟੋਰੀ ਬੋਰਡ ਦੀ ਵਰਤੋਂ (Using Story Board)

ਸਟੋਰੀ ਬੋਰਡ ਅਤੇ ਟਾਈਮ ਲਾਈਨ ਇਕ ਹੀ ਤਰ੍ਹਾਂ ਦੇ ਟੂਲ ਹਨ। ਇਹ ਵਿਊ ਵਿਚਲੇ ਵੀਡੀਓ ਕਲਿਪ ਅਤੇ images ਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਰੱਖਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਇਕ ਤਰਤੀਬ ਵਿਚ ਦਰਸਾਏ ਜਾਂਦੇ ਹਨ। ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਦੀ ਮੱਦਦ ਨਾਲ ਡਿਵੈਲਪਰ (developer) ਆਪਣੀ ਇਮੇਜ (image) ਦੀ ਟਾਈਮ ਦੀ ਅਵਧੀ ਨੂੰ ਨਿਰਧਾਰਿਤ ਕਰਦਾ ਹੈ।ਸਟੋਰੀ ਬੋਰਡ ਹੇਠਾਂ ਦਿੱਤੀ ਤਸਵੀਰ ਵਿਚ ਦਿਖਾਇਆ ਗਿਆ ਹੈ ਜਿਸ ਵਿਚ ਕਲਿਪਸ ਤੁਹਾਡੇ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਦਰਸਾਉਂਦੇ ਹਨ।



ਸਟੋਰੀ ਬੋਰਡ ਦਾ ਨਿਰਮਾਣ (Making Story Board)

ਕੋਲੈਕਸ਼ਨ ਆਈਕੋਨ ਤੇ ਕਲਿੱਕ ਕਰੋ: ਉਸ ਫੋਲਡਰ ਨੂੰ ਸਿਲੈਕਟ ਕਰੋ ਜਿਥੇ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਭਰਨ ਲਈ ਪਿਕਚਰ ਹੋਣ।

- ਹਰ ਇਕ ਪਿਕਚਰ ਨੂੰ ਪ੍ਰੀ-ਵਿਊ ਵਿੰਡੋਜ਼ ਤੇ ਦੇਖਣ ਲਈ ਕਲਿੱਕ ਕਰੋ।
- ਇਹ ਨਿਰਧਾਰਿਤ ਕਰਨ ਮਗਰੋਂ ਕਿ ਤੁਹਾਡੀ ਮੂਵੀ ਵਿਚ ਕਿਹੜੀਆਂ ਪਿਕਚਰ ਸ਼ਾਮਿਲ ਹੋਣਗੀਆਂ, ਕਲਿਪ ਨੂੰ ਕਲਿੱਕ ਅਤੇ ਡਰੈਗ (drag) ਕਰਦੇ ਹੋਏ ਅਤੇ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਪਾਉ।

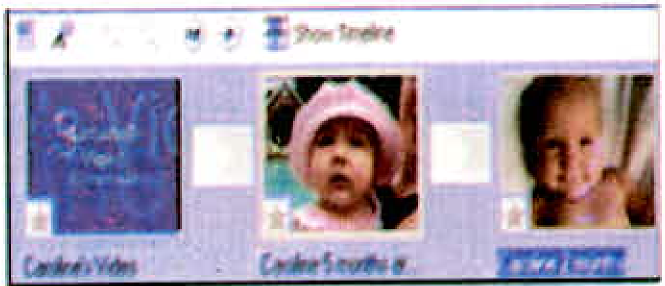
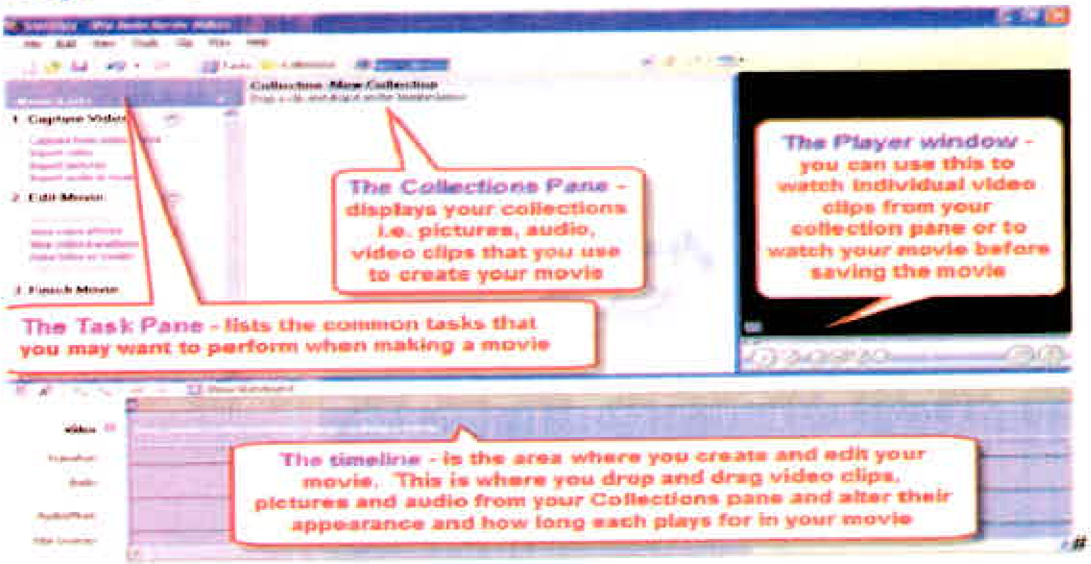


Fig: Showing the clips in Stroyboard

- ਸਟੋਰੀ ਬੋਰਡ ਤੇ ਪਿਕਚਰ ਦੀ ਤਰਤੀਬ ਨੂੰ ਸਹੀ ਕਰਨ ਲਈ ਕਲਿੱਕ ਅਤੇ ਡਰੈਗ ਦਾ ਪ੍ਰਯੋਗ ਕਰੋ।

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੀ ਵਰਤ ਮੁੱਖ ਸਕਰੀਨ



ਮੂਵੀ ਟਾਸਕ ਪੇਨ (Movie Task pane)

ਵੀਡੀਓ Capture (Video Capture)

ਵੀਡੀਓ capture ਡਿਵਾਈਸ ਦੀ ਮਦਦ ਨਾਲ ਤੁਸੀ live ਜਾਂ ਰਿਕਾਰਡਿਡ (recorded) ਵੀਡੀਓ ਨੂੰ ਆਪਣੇ ਕੰਪਿਊਟਰ ਵਿਚ ਤਬਦੀਲ ਕਰ ਸਕਦੇ ਹੋ। ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਤੁਸੀ ਹੇਠ ਲਿਖੇ ਪ੍ਰਕਾਰ ਦੇ capture ਡਿਵਾਈਸ ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਦੇ ਹੋ।

ਐਡਿਟ ਵੀਡੀਓ (Edit Video)

ਇਸ ਦੀ ਮਦਦ ਨਾਲ ਤੁਸੀ ਟਾਈਟਲ, effect ਨੂੰ ਆਪਣੀ ਮੂਵੀ ਵਿਚ ਸ਼ਾਮਲ ਕਰ ਸਕਦੇ ਹੋ।

ਫਿਨਿਸ਼ ਵੀਡੀਓ (Finish Video)

ਇਸ ਦੇ ਨਾਲ ਤੁਸੀ ਬਣਾਈ ਮੂਵੀ ਨੂੰ render ਕਰ ਸਕਦੇ ਹੋ।

ਕੋਲੈਕਸ਼ਨ (collection)

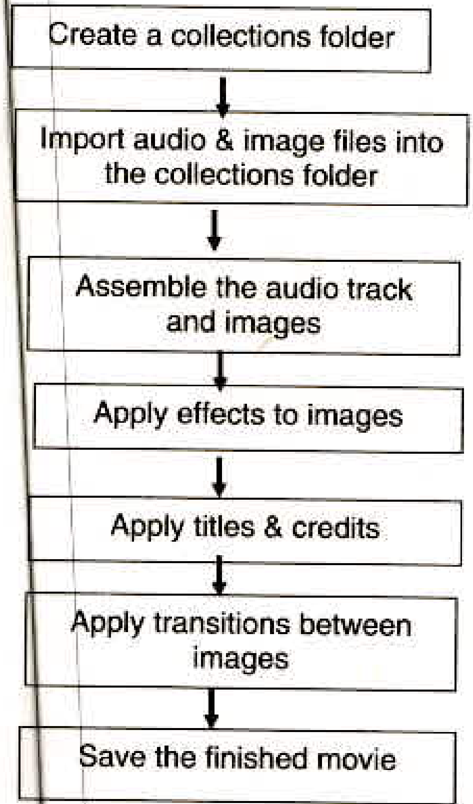
ਇਹ ਇਕ ਡਾਟਾ ਬੇਸ ਦੀ ਤਰ੍ਹਾਂ ਹੈ, ਜਿਥੇ ਤੁਹਾਡੇ imported ਅਤੇ captured ਵੀਡੀਓ ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ।

ਸਪਲਿਟ ਬਟਨ (split button)

ਇਕ ਵੀਡੀਓ ਨੂੰ ਐਡਿਟ ਕਰਨ ਵੇਲੇ ਜਦੋਂ ਤੁਸੀ ਵੀਡੀਓ ਫਾਈਲ ਨੂੰ pause ਕਰੋਗੇ ਤੁਸੀ ਇਕ ਬਲੂ ਬਾਰ ਨੂੰ ਵੇਖੋਗੇ। ਵੀਡੀਓ pause ਕਰਨ ਮਗਰੋਂ ਸਪਲਿਟ (split) ਬਟਨ ਤੇ ਕਲਿੱਕ ਕਰੋ। ਇਸ ਤਰ੍ਹਾਂ ਕਰਨ ਨਾਲ ਤੁਸੀਂ ਜੋ ਵੀ ਸਿਲੈਕਟ (select) ਕੀਤਾ ਹੋਵੇਗਾ ਉਹ ਦੋ ਭਾਗਾਂ ਵਿਚ ਸਪਲਿਟ (split) ਹੋ ਜਾਏਗਾ।

ਸਕਰੀਨ ਸ਼ਾਟ ਬਟਨ (Screen shot button)

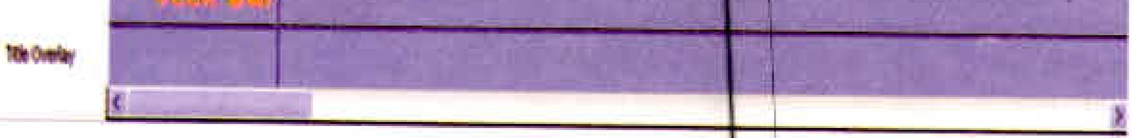
ਇਸ ਬਟਨ ਦੀ ਮਦਦ ਨਾਲ ਤੁਸੀਂ ਤੁਹਾਡੀ ਸਕਰੀਨ ਤੇ ਜੋ ਵੀ ਹੈ ਉਸਦਾ ਸਕਰੀਨ ਸ਼ਾਟ ਲੈ ਸਕਦੇ ਹੋ ਅਤੇ ਆਪਣੇ ਕੰਪਿਊਟਰ ਤੇ ਸੇਵ ਕਰ ਸਕਦੇ ਹੋ। ਇਸ ਸਕਰੀਨ ਸ਼ਾਟ ਨੂੰ ਅਸੀਂ ਆਪਣੀ ਮਰਜ਼ੀ ਨਾਲ ਕਿਸੇ ਵੀ ਮੂਵੀ ਵਿਚ ਵਰਤ ਸਕਦੇ ਹਾਂ। ਸਕਰੀਨ ਸ਼ਾਟ ਦੀ ਮਿਸਾਲ ਹੇਠਾਂ ਦਿੱਤੀ ਗਈ ਹੈ:



Flow chart of making Movie

ਵੀਡੀਓ ਟਰੈਕ (Video track)

ਇਸ ਥਾਂ ਤੇ ਅਸੀਂ ਆਪਣੇ ਵੀਡੀਓ ਨੂੰ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਭਰਦੇ ਹਾਂ। ਵੀਡੀਓ ਕਲਿਪ ਇਕ ਵੀਡੀਓ ਟਰੈਕ ਜਾਂ ਕੋਲੈਕਸ਼ਨ ਵਿਚ ਹੋ ਸਕਦਾ ਹੈ ਅਤੇ ਇਸ ਨੂੰ ਸਪਲਿਟ (split) ਵੀ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।



ਆਡੀਓ ਟਰੈਕ (Audio track)

ਇਥੇ ਆਡੀਓ ਜਾਂ ਮਿਊਜ਼ਿਕ (music) ਕਲਿਪ ਨੂੰ ਵੀਡੀਓ ਵਿਚ ਭਰਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ volume ਨੂੰ ਵੀ ਨਿਰਧਾਰਿਤ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

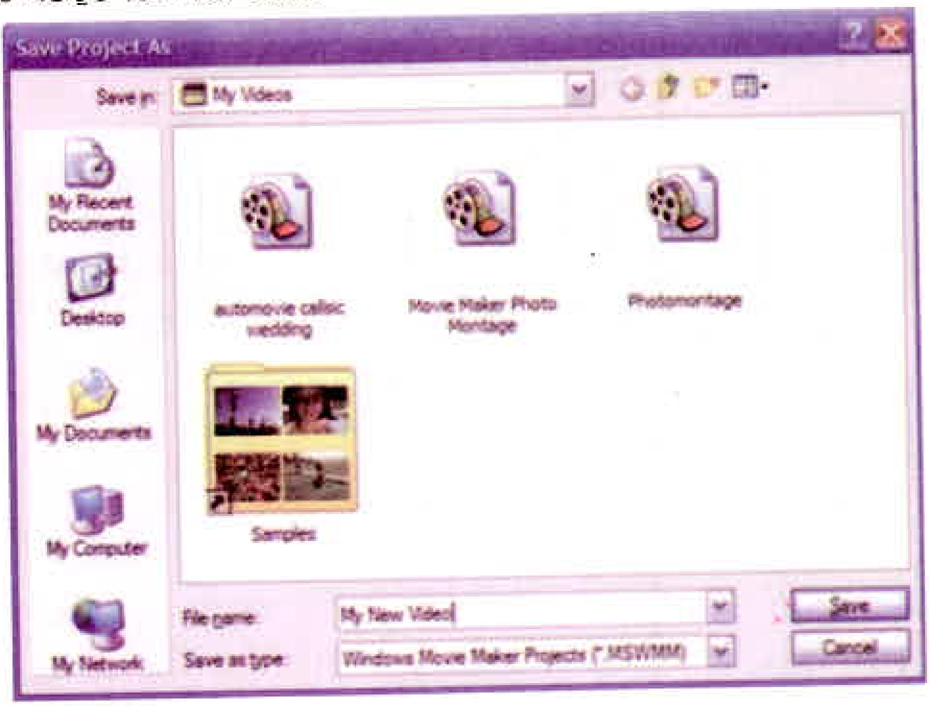
ਟਾਈਟਲ ਟਰੈਕ (Title track)

ਇਹ ਟਾਈਟਲ ਬਣਾਉਣ ਦੇ ਕੰਮ ਆਉਂਦਾ ਹੈ। ਇਸਦੇ ਜ਼ਰੂਰੀ ਬਟਨ ਸੇਵ, ਨਿਊ ਜਾਂ ਓਪਨ ਹਨ। ਐਡਿਟ ਕਰਨ ਲਈ, ਕੁਝ ਮੱਹਤਵਪੂਰਣ ਬਟਨ, ਟਾਈਮ ਲਾਈਨ ਦੇ ਸੱਜੇ ਪਾਸੇ ਵੇਖੇ ਜਾ ਸਕਦੇ ਹਨ।

ਨਿਊ ਪ੍ਰੋਜੈਕਟ ਖੋਲ੍ਹੋ (open new project)

ਮੂਵੀ ਮੇਕਰ ਨੂੰ ਖੋਲ੍ਹਣ ਮਗਰੋਂ, ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਦਾ ਨਾਮ ਰੱਖੋ ਅਤੇ ਉਸ ਨੂੰ ਸੇਵ ਕਰੋ। ਅਜਿਹਾ ਕਰਨ ਲਈ CTRL+S ਦਬਾਉ ਜਾਂ file->save ਤੇ ਕਲਿੱਕ ਕਰੋ।

ਜੇਕਰ ਤੁਸੀਂ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਐਕਸਟਰਨਲ (external) ਡਰਾਈਵ ਤੇ ਸੇਵ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਤਾਂ ਇਹ ਤੁਹਾਨੂੰ ਮੂਵੀ ਫੋਲਡਰ ਬਣਾਉਣ ਵਿਚ ਮਦਦ ਕਰੇਗਾ।



ਆਪਣੇ ਕੰਪਿਊਟਰ ਤੋਂ ਵੀਡੀਓ ਇਮਪੋਰਟ (import) ਕਰਨਾ

ਜੇਕਰ ਵੀਡੀਓ ਫਾਈਲਜ਼ ਜਿਹੜੀਆਂ ਤੁਸੀਂ ਐਡੀਟ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਤੁਹਾਡੇ ਕੰਪਿਊਟਰ ਤੇ ਸੇਵ ਹੋਣ, ਤਾਂ ਇਨ੍ਹਾਂ ਨੂੰ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਇਮਪੋਰਟ (import) ਕਰਨਾ ਬਹੁਤ ਅਸਾਨ ਹੈ। ਮੂਵੀ ਮੇਕਰ ਤਕਰੀਬਨ ਸਾਰੇ ਵੀਡੀਓ ਫਾਰਮੇਟ ਨੂੰ ਖੋਲ੍ਹਣ ਦੇ ਸਮਰਥ ਹੈ। ਇਸ ਲਈ capture ਵੀਡੀਓ ਮੀਨੂੰ ਤੋਂ ਇਮਪੋਰਟ (import) ਵੀਡੀਓ ਨੂੰ ਕਲਿਕ ਕਰੋ।

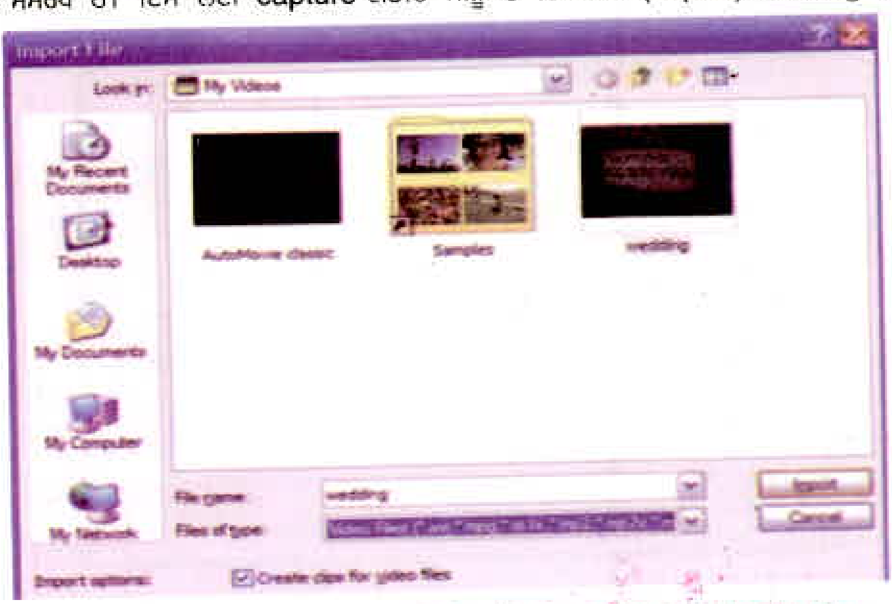


Fig A: ਮਾਈ ਵੀਡੀਓ ਫੋਲਡਰ ਵਿੱਚ ਮੌਜੂਦ ਫਾਈਲਾਂ ਦੀ ਸੂਚੀ, 'ਕਰੀਏਟ ਕਲਿਪਸ ਫਾਰ ਵੀਡੀਓ,' ਨੂੰ ਕਲਿੱਕ ਕੀਤਾ ਗਿਆ।

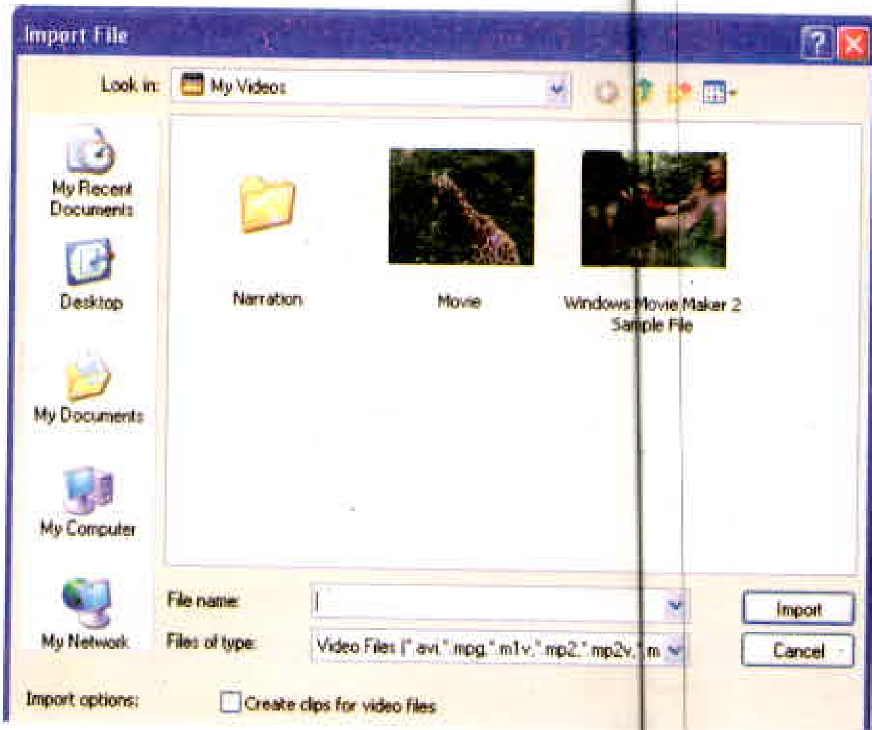


Fig B: ਮਾਈ ਵੀਡੀਓ ਫਾਲਡਰ ਵਿੱਚ ਮੌਜੂਦਾ ਫਾਈਲਾਂ ਦੀ ਸੂਚੀ, 'ਕਰੀਏਟ ਕਲਿਪਸ ਫਾਰ ਵੀਡੀਓ ' ਕਲਿੱਕ ਨਹੀਂ ਕੀਤਾ ਗਿਆ।

ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਵੀਡੀਓ ਇਮਪੋਰਟ (import) ਕਰਨ ਦੇ ਦੋ ਤਰੀਕੇ ਹਨ।

ਪਹਿਲਾ ਤਰੀਕਾ (ਤਸਵੀਰ "B") ਇਸ ਵਿਚ create clip for ਵੀਡੀਓ ਫਾਰਮੇਟ ਨੂੰ ਕਲਿੱਕ ਨਹੀ ਕੀਤਾ, ਵੀਡੀਓ ਨੂੰ ਇਮਪੋਰਟ (import) ਕਰਨ ਮਗਰੋ ਕਾਨਟੈਂਟ ਪੇਨ (content pane) ਵਿਚ ਕੋਈ ਕਲਿਪ ਨਹੀ ਬਣਦਾ।

ਦੂਸਰਾ ਤਰੀਕਾ (ਤਸਵੀਰ "A") ਜਿਸ ਵਿਚ create clip from video ਕਲਿਕ ਹੋਵੇ ਤਾਂ ਹੇਠਾਂ ਦਿਤੀ ਤਸਵੀਰ ਨਜ਼ਰ ਆਵੇਗੀ ।

ਇਮਪੋਰਟ (import) ਵੀਡੀਓ ਤੋਂ ਬਣਨ ਵਾਲੇ ਕਲਿਪ ਨਜ਼ਰ ਆਉਣਗੇ।

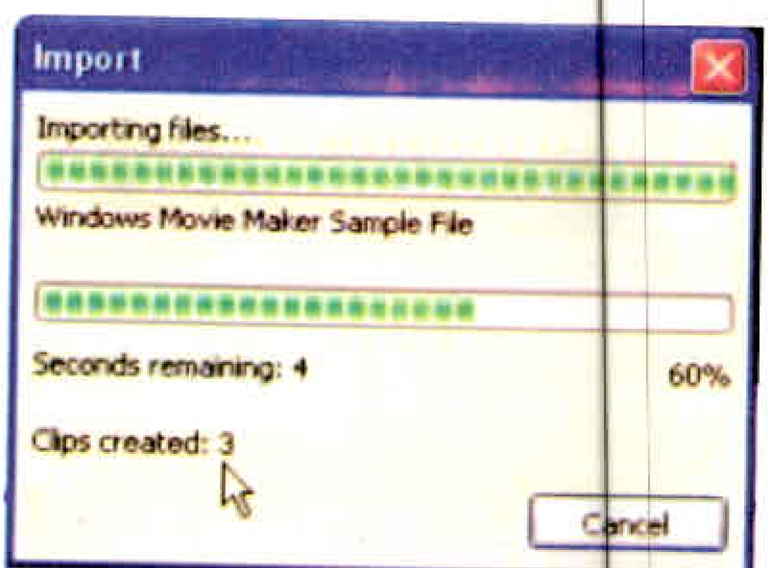


Fig: ਕਲਿਪ ਫਾਰਮੇਸ਼ਨ

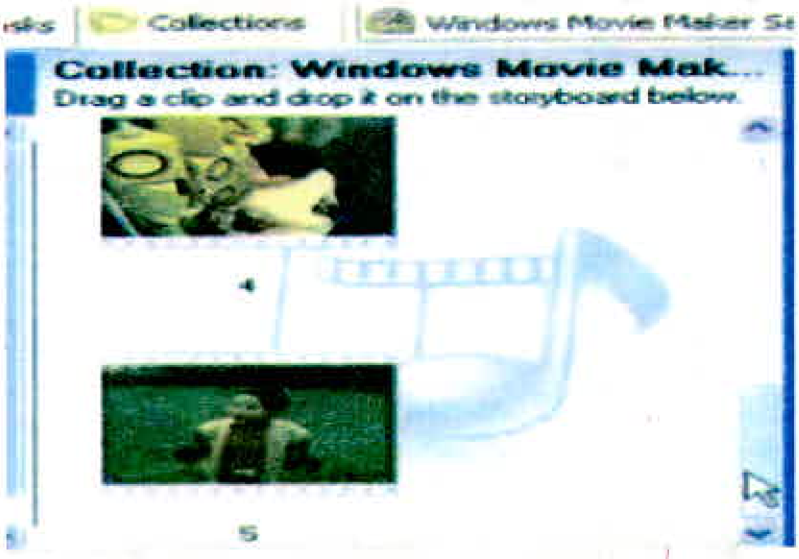
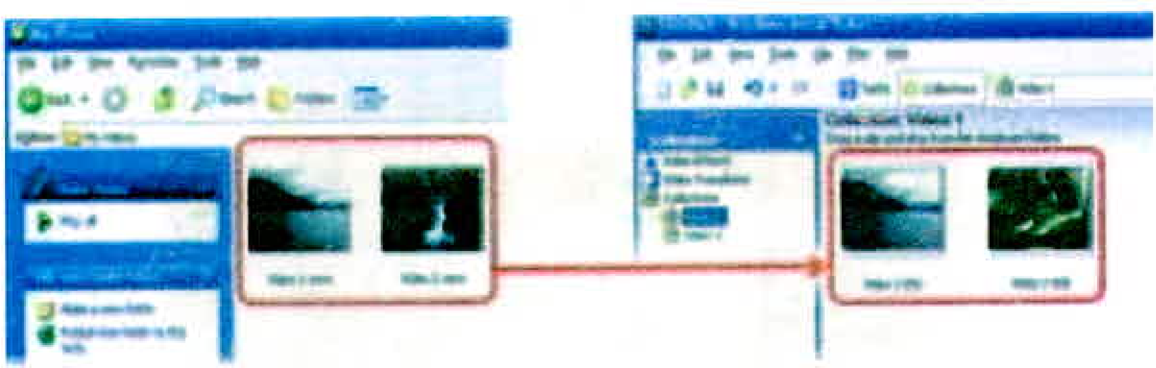


Fig: ਕਾਨਟੈਂਟ ਪੇਨ ਵਿੱਚ 4 ਅਤੇ 5 ਨੰਬਰ ਕਲਿਪਸ

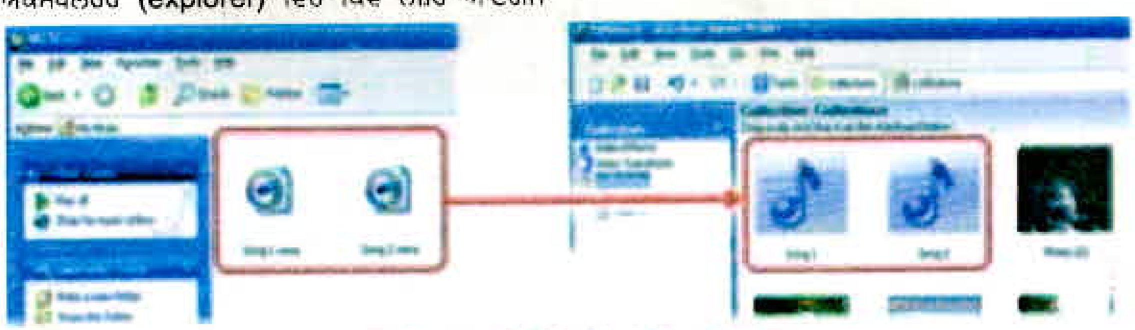
ਇਮਪੋਰਟ ਪਿਕਚਰ (Import picture)

ਇਹ ਤੁਹਾਡੇ ਦੁਆਰਾ ਮੂਵੀ ਵਿਚ ਭਰਨ ਵਾਲੀਆਂ ਤਸਵੀਰਾਂ ਨੂੰ ਦੇਖਣ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਇਮਪੋਰਟ ਪਿਕਚਰ (import picture) ਕਰਨਾ ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਇਮਪੋਰਟ (import) ਕਰਨ ਦੀ ਤਰ੍ਹਾਂ ਹੀ ਹੁੰਦਾ ਹੈ।



ਵਿੰਡੋ ਐਕਸਪਲੋਰਰ ਅਤੇ ਵਿੰਡੋ ਮੂਵੀ ਮੇਕਰ ਵਿੱਚ ਵੀਡੀਓ ਫਾਈਲ

ਦਿਤੀ ਤਸਵੀਰ ਵਿਚ ਦਰਸਾਇਆ ਗਿਆ ਹੈ ਕਿ ਸੋਰਸ ਵੀਡੀਓ ਫਾਈਲ, ਆਡੀਓ ਫਾਈਲ ਅਤੇ ਫੋਟੋ ਵਿੰਡੋਜ਼ ਐਕਸਪਲੋਰਰ (explorer) ਵਿਚ ਕਿਵੇਂ ਨਜ਼ਰ ਆਏਗੀ।



ਵਿੰਡੋ ਐਕਸਪਲੋਰਰ ਅਤੇ ਵਿੰਡੋ ਮੂਵੀ ਮੇਕਰ ਵਿੱਚ ਆਡੀਓ ਫਾਈਲ

ਵੀਡੀਓ ਸਕਰੀਨ / ਮੋਨੀਟਰ

ਤੁਸੀਂ ਖੱਬੇ ਪਾਸੇ ਥੱਲੇ ਬੇਸਿਕ ਪਲੇਅ, ਸਟਾਪ, ਫਰੇਮ ਫਾਰਵਰਡ / ਬੈਕਵਰਡ ਬਟਨ ਦੇਖ ਸਕਦੇ ਹੋ। ਫਰੇਮ ਬਟਨ ਤੁਹਾਨੂੰ ਇਕ ਫਰੇਮ ਅੱਗੇ ਜਾਂ ਪਿੱਛੇ ਜਾਣ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ। ਤੁਸੀਂ ਮੋਨੀਟਰ ਤੋਂ ਵੀਡੀਓ ਕਲਿਪ ਨੂੰ ਸਪਲਿਟ (split) ਕਰਨ ਲਈ ਜਾਂ ਪਿਕਚਰ ਹਾਸਿਲ ਕਰਨ ਲਈ ਜਿਹੜੇ ਬਟਨ ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹੋ ਉਹਨਾਂ ਬਟਨ ਦੀ ਵਰਤੋਂ ਹੇਠਾਂ ਦਰਸਾਈ ਗਈ ਹੈ :

ਕਲਿਪ ਨੂੰ ਦੇਖਣ ਲਈ

1. ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ ਦਿਖਣ ਵਾਲੇ ਕਲਿਪ ਤੇ ਕਲਿੱਕ ਕਰੋ।
2. ਮੋਨੀਟਰ ਉੱਪਰ ਵੀਡੀਓ ਜਾਂ ਆਡੀਓ ਕਲਿਪ ਦਾ ਪ੍ਰੀ-ਵਿਊ ਦੇਖਣ ਲਈ ਪਲੇਅ ਬਟਨ ਤੇ ਕਲਿੱਕ ਕਰੋ।

ਨਿਊ ਕੋਲੈਕਸ਼ਨ ਬਣਾਉਣ ਲਈ

1. ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ, ਕੋਲੈਕਸ਼ਨ ਫੋਲਡਰ ਤੇ ਕਲਿੱਕ ਕਰੋ।
2. ਟੂਲਜ਼ ਮੀਨੂੰ ਵਿਚ ਨਿਊ ਕੋਲੈਕਸ਼ਨ ਫੋਲਡਰ ਤੇ ਕਲਿੱਕ ਕਰੋ।
3. ਨਿਊ ਕੋਲੈਕਸ਼ਨ ਦਾ ਨਾਮ ਰਖੋ।

ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਨੂੰ ਦੋ ਭਾਗਾ ਵਿਚ ਵੰਡਣ ਲਈ

1. ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ ਸਪਲਿਟ (split) ਕਰਨ ਵਾਲੇ ਜਾਂ ਵੀਡੀਓ ਜਾਂ ਆਡੀਓ ਤੇ ਕਲਿੱਕ ਕਰੋ।
2. ਮੋਨੀਟਰ ਥੱਲੇ, ਸਪਲਿਟ (split) ਕਰਨ ਵਾਲੇ ਕਲਿਪ ਤੱਕ ਸਲਾਈਡ ਲੈਕੇ ਜਾਉ।
3. ਮੋਨੀਟਰ ਤੇ ਸਪਲਿਟ (split) ਬਟਨ ਨੂੰ ਕਲਿੱਕ ਕਰੋ।

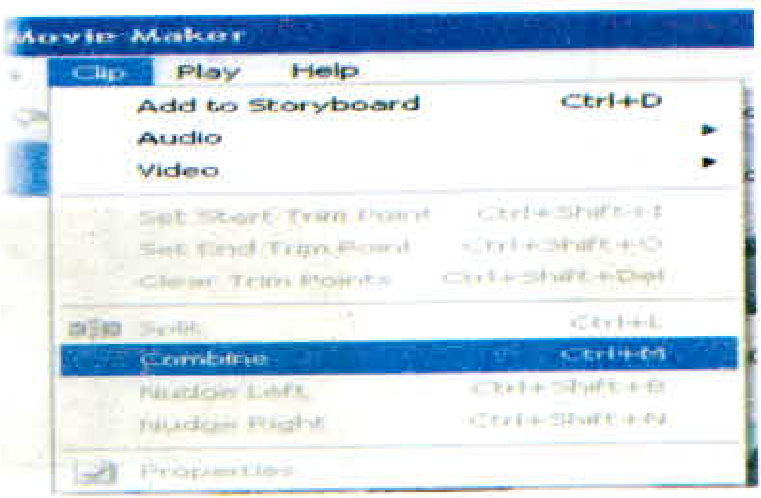
The resulting video clips now appear in the Contents pane.

ਨੋਟ : ਸਲਾਈਡਜ਼ ਨੂੰ ਸਹੀ ਜਗ੍ਹਾ ਤੇ ਰੱਖਣ ਲਈ, ਤੁਸੀ ਸਲਾਈਡਜ਼ ਨੂੰ ਖਿੱਚ ਕੇ ਕਲਿਪ ਨੂੰ ਸਪਲਿਟ (split) ਕਰਨ ਵਾਲੇ ਬਿੰਦੂ ਤੋਂ ਪਹਿਲਾਂ ਰੱਖੋ, ਪਲੇਅ ਬਟਨ ਤੇ ਕਲਿੱਕ ਕਰੋ, ਅਤੇ ਫਿਰ ਸਪਲਿਟ (split) ਕਰਨ ਵਾਲੇ ਅਸਲ ਸਥਾਨ ਤੇ ਪਹੁੰਚ ਕੇ ਸਟਾਪ ਬਟਨ ਤੇ ਕਲਿੱਕ ਕਰੋ।

ਸਪਲਿਟ (split) ਕੀਤੀ ਹੋਈ ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਨੂੰ ਜੋੜਨ ਲਈ

1. ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ CTRL ਕੀ ਨੂੰ ਦਬਾਉ ਅਤੇ ਫਿਰ ਜੋੜਨ ਵਾਲੇ ਦੋਵੇਂ ਕਲਿਪਸ ਨੂੰ ਸਲੈਕਟ ਕਰੋ।

ਨੋਟ : ਲੜੀਵਾਰ ਕਲਿਪਸ ਨੂੰ ਸਲੈਕਟ ਕਰਨ ਲਈ, ਪਹਿਲੇ ਕਲਿਪ ਨੂੰ ਚੁਣੋ, Shift ਕੀ ਨੂੰ ਦਬਾਈ ਰਖੋ ਅਤੇ ਆਖਰੀ ਕਲਿਪ ਨੂੰ ਚੁਣੋ।



2. ਕਲਿਪ ਮੀਨੂੰ ਤੇ ਜਾਉ ਅਤੇ ਕੰਬਾਈਨ (combine) ਨੂੰ ਕਲਿੱਕ ਕਰੋ। ਬਣਾਇਆ ਗਿਆ ਕਲਿਪ, ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ ਦੇਖੋ।

ਨੋਟ : ਤੁਸੀਂ ਇਹ ਸਾਰੀ ਪ੍ਰਕਿਰਿਆ ਸਟੋਰੀ ਬੋਰਡ / ਟਾਈਮ ਲਾਈਨ ਨਾਲ ਵੀ ਕਰ ਸਕਦੇ ਹੋ।

ਕਲਿਪ ਨੂੰ ਕੇਲੈਕਸ਼ਨ ਵਿਚ ਕਾਪੀ ਕਰਨ ਲਈ

1. ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ ਕਾਪੀ ਕਰਨ ਵਾਲੇ ਕਲਿਪ ਤੇ ਕਲਿੱਕ ਕਰੋ।
2. ਐਡਿਟ ਮੀਨੂੰ ਵਿਚ ਕਾਪੀ ਤੇ ਕਲਿੱਕ ਕਰੋ।
3. ਕੇਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ ਕਾਪੀ ਕਰਨ ਵਾਲੇ ਕਲਿਪਸ ਨੂੰ ਕਲਿੱਕ ਕਰੋ ਅਤੇ ਫਿਰ ਕਾਨਟੈਂਟ ਪੇਨ ਨੂੰ ਕਲਿੱਕ ਕਰੋ।
4. ਐਡਿਟ ਮੀਨੂੰ ਤੇ ਜਾ ਕੇ ਪੇਸਟ (paste) ਕਲਿੱਕ ਕਰੋ।

ਇਕ ਵੀਡੀਓ ਤੋਂ ਮੋਨੀਟਰ ਰਾਹੀ ਪਿਕਚਰ ਲੈਣ ਲਈ

1. ਕਾਨਟੈਂਟ ਪੇਨ ਵਿਚ, ਪਿਕਚਰ ਲੈਣ ਵਾਲੇ ਵੀਡੀਓ ਕਲਿਪ ਤੇ ਕਲਿੱਕ ਕਰੋ। ਮੋਨੀਟਰ ਉੱਪਰ, ਪਲੇਅ-ਬੈਕ (playback) ਨਿਸ਼ਾਨ ਨੂੰ seekbar ਰਾਹੀ ਉਸ ਫਰੇਮ ਤੱਕ ਲੈਕੇ ਜਾਉ ਜਿਥੋਂ ਤੁਸੀਂ ਵੀਡੀਓ ਦੀ ਪਿਕਚਰ ਹਾਸਿਲ ਕਰਨੀ ਚਾਹੁੰਦੇ ਹੋ।
2. ਟੂਲ ਮੀਨੂੰ ਵਿਚ, take ਪਿਕਚਰ ਤੇ ਕਲਿੱਕ ਕਰੋ ਜਾਂ ਫਿਰ ਮੋਨੀਟਰ ਤੇ take ਪਿਕਚਰ ਤੇ ਕਲਿੱਕ ਕਰੋ। ਫਾਈਲ ਮੀਨੂੰ ਬਾਕਸ ਵਿਚ ਨਵੀਂ ਪਿਕਚਰ ਦਾ **ਨਾਮ** ਲਿਖੋ ਅਤੇ ਸੇਵ ਤੇ ਕਲਿੱਕ ਕਰੋ।

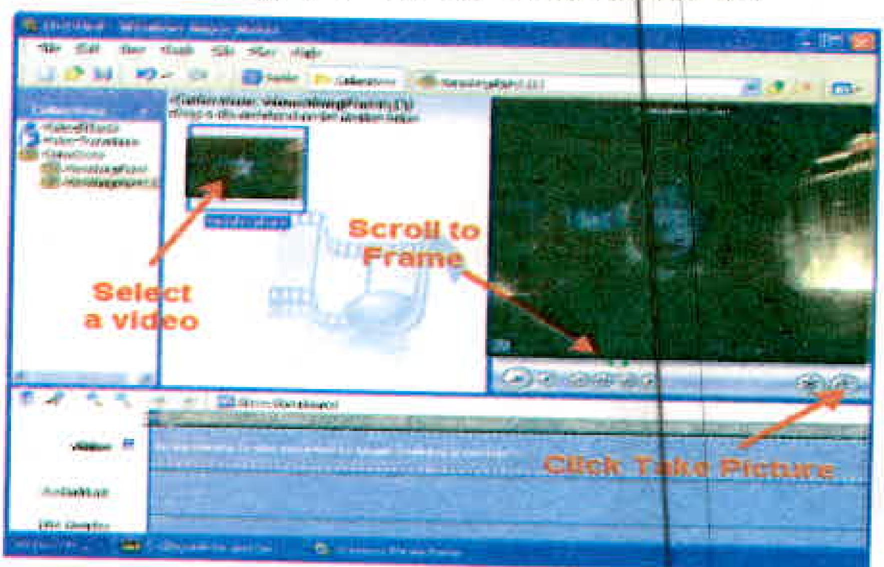


Fig: ਵੀਡੀਓ ਤੋਂ ਪਿਕਚਰ ਲੈਣ ਲਈ ਪਿਕਚਰ ਬਟਨ ਨੂੰ ਵਰਤਣਾ

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1. ਹੇਠਾਂ ਦਿੱਤੇ ਪ੍ਰਸ਼ਨਾਂ ਦੇ ਜਵਾਬ ਦਿਓ ?

1 ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਸਾਫਟਵੇਅਰ ਦੇ ਕੀ ਉਪਯੋਗ ਹਨ ?
2 ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੇ ਮੂਵੀ ਟਾਸਕ ਦੇ ਵਿਚ ਕਿਹੜੇ - ਕਿਹੜੇ ਔਪਸ਼ਨ ਹਨ ?
3 ਇਮਪੋਰਟ ਵੀਡੀਓ ਫਾਈਲ ਆਪਸ਼ਨ ਦੇ "ਵੀਡੀਓ ਫਾਈਲ ਤੋਂ create ਕਲਿਪ" ਦਾ ਕੀ ਉਪਯੋਗ ਹੈ ?
4 ਕਲਿਪ ਅਤੇ ਵੀਡੀਓ ਵਿਚ ਮੁੱਖ ਅੰਤਰ ਕੀ ਹੈ ?
5 ਮਿਸਾਲ ਦੇ ਕੇ ਸਮਝਾਉ ਕਿ ਤੁਸੀ "take picture of monitor" ਦਾ ਇਸਤੇਮਾਲ ਕਿਵੇਂ ਕਰੋਗੇ ?

ਪ੍ਰਸ਼ਨ 2: ਖਾਲੀ ਸਥਾਨ ਭਰੋ

1 ਇਕ ਵੱਡੀ ਵੀਡੀਓ ਫਾਈਲ ਦੇ ਛੋਟੇ ਭਾਗ ਨੂੰ ________ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
2 ਮੂਵੀ ਮੇਕਰ ਦੇ ________ ਹਿੱਸੇ ਵਿਚ ਅਲੱਗ-ਅਲੱਗ ਕਲਿਪ ਦੀ ਸੰਭਾਲ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ?
3 ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਕਲਿਪਸ ਨੂੰ ਵੱਖ-ਵੱਖ ਭਾਗਾਂ ਵਿਚ ਵੰਡਣ ਲਈ __________ ਕੰਮਾਂਡ ਦਾ ਉਪਯੋਗ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ?
4 ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੇ ਯੂਜ਼ਰ ਇੰਟਰਫੇਸ ਦੇ 1______, 2 _____, 3 ______ ਭਾਗ ਹਨ ?
5 ਡਿਜ਼ਿਟਲ ਮੀਡਿਆ ਫਾਈਲ ਅਤੇ ਪਿਕਚਰਜ਼ ਫਾਈਲ ਜਿਹੜੀਆਂ ਅਸੀ ਇਮਪੋਰਟ ਕਰਦੇ ਹਾਂ, ਉਹਨਾਂ ਨੂੰ __________ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
6

ਪ੍ਰਸ਼ਨ 3: ਦੱਸੋ ਸਹੀ ਹੈ ਜਾਂ ਗਲਤ ਹੈ

1 ਜਦੋਂ ਵੀ ਅਸੀ ਵੀਡੀਓ, ਆਡੀਓ ਜਾਂ ਪਿਕਚਰ ਫਾਈਲ ਨੂੰ ਇਮਪੋਰਟ ਕਰਦੇ ਹਾਂ ਤਾਂ ਅਸਲ ਫਾਈਲ ਆਪਣੀ ਹੀ ਜਗ੍ਹਾ ਤੇ ਰਹਿੰਦੀ ਹੈ ?
2 ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦੀ ਪ੍ਰੋਜੈਕਟ ਫਾਈਲ ਦੀ ਐਕਸਟੈਂਸ਼ਨ .mswmm ਹੁੰਦੀ ਹੈ ?
3 ਕੋਲੈਕਸ਼ਨ ਪੇਨ ਵਿਚ ਅਸੀ ਕੇਵਲ ਵੀਡੀਓ ਕਲਿਪ ਹੀ ਰੱਖ ਸਕਦੇ ਹਾਂ ?
4 ਇਕ ਤੋਂ ਵੱਧ ਫਾਈਲਸ ਨੂੰ ਇਕੋ ਸਮੇਂ ਵਿਚ ਇਮਪੋਰਟ ਕਰਨ ਲਈ ਅਸੀਂ ਕਲਿਕ ਅਤੇ ਸ਼ਿਫਟ ਕੀ ਦਾ ਇਸਤੇਮਾਲ ਕਰਦੇ ਹਾਂ ?

Answer to objective questions

Q	.1	.2	.3	.4	.5
Q2	Clip	Collection	Split	Menu Bar Tool Bar ,Panes ,Timeline	Source File
Q3	T	T	F	T	NIL

ਅਧਿਆਇ

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ (ਭਾਗ – 2)
Windows Movie Maker (Part – 2)

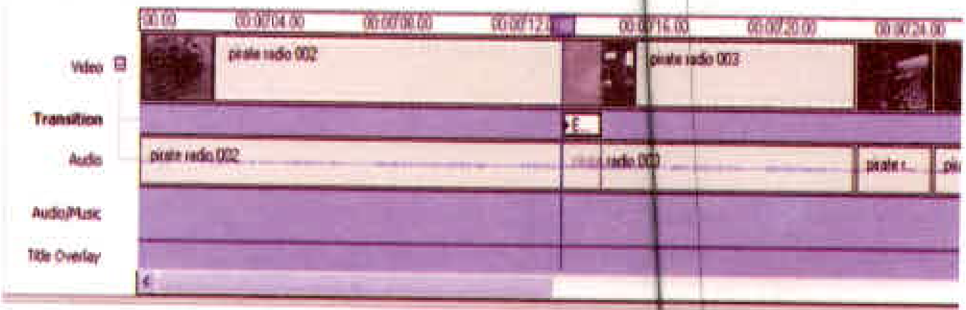
5

ਭੂਮਿਕਾ

ਇਕ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਸ਼ੁਰੂ ਕਰਨ ਲਈ ਅਤੇ ਮੂਵੀ ਨੂੰ ਬਣਾਉਣ ਲਈ ਤੁਹਾਨੂੰ ਵੀਡੀਓ, ਆਡੀਓ ਜਾਂ ਸਟੋਰੀ ਬੋਰਡ/ਟਾਈਮ ਲਾਈਨ ਉਪਰ ਇਮਪੋਰਟ ਕਰਨਾ ਪਵੇਗਾ। ਸਟੋਰੀ ਬੋਰਡ/ਟਾਈਮ ਲਾਈਨ ਉੱਪਰ ਕਲਿਪਸ ਤੁਹਾਡੇ ਲਈ ਪ੍ਰੋਜੈਕਟ ਜਾਂ ਮੂਵੀ ਬਣਾਉਣ ਲਈ ਕਾਨਟੈਂਟ ਪ੍ਰਦਾਨ ਕਰਦੇ ਹਨ। ਤੁਸੀ ਪ੍ਰੋਜੈਕਟ ਉੱਪਰ ਕੰਮ ਕਰਨ ਵੇਲੇ ਸਟੋਰੀ ਬੋਰਡ ਜਾਂ ਸਟੋਰੀ ਬੋਰਡ/ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਜਾ ਸਕਦੇ ਹੋ।

ਟਾਈਮ ਲਾਈਨ ਵਿਊ (Time line View)

ਇਕ ਵਾਰ ਜਦੋਂ ਤੁਸੀ ਹੋਰ ਜਿਆਦਾ ਗੁੰਝਲਦਾਰ ਪ੍ਰਭਾਵ ਪਾਉਣ ਲਈ ਅਤੇ ਐਡੀਟਿੰਗ ਕਰਨ ਲਈ ਤਿਆਰ ਹੁੰਦੇ ਹੋ ਤਾਂ ਤੁਸੀ ਪਾਉਗੇ ਕਿ ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਵਿਚ ਇਹ ਸਟੋਰੀ ਬੋਰਡ ਤੋਂ ਜਿਆਦਾ ਸੌਖਾ ਹੈ। ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਵਿਚ ਜਾਣ ਲਈ ਸਟੋਰੀ ਬੋਰਡ ਖੇਤਰ ਦੇ ਉੱਪਰ ਸੋ ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਉੱਤੇ ਕਲਿੱਕ ਕਰੋ। ਸਟੋਰੀ ਬੋਰਡ ਦੇ ਇਕ ਇਕੱਲੇ ਟਰੈਕ ਦੀ ਬਜਾਏ ਤੁਸੀਂ ਇਥੇ ਪੰਜ ਅਲੱਗ-ਅਲੱਗ ਟਰੈਕ ਦੇਖੋਗੇ।



ਉੱਪਰ ਤੋਂ ਹੇਠਾਂ ਤੱਕ ਇਹ ਪੰਜ ਅਲੱਗ-ਅਲੱਗ ਹਨ :-

ਵੀਡੀਓ (Video): ਦ੍ਰਿਸ਼ ਜਿਹੜੇ ਤੁਹਾਡੇ ਕੈਮਰੇ ਜਾਂ ਕਿਸੇ ਫਾਈਲ ਵਿਚੋਂ ਲਏ ਗਏ ਹਨ।

ਟਰਾਂਜ਼ਿਸ਼ਨ (Transition): ਹਰਕਤਾਂ ਜਿਹੜੀ ਇਕ ਤੋਂ ਬਾਅਦ ਇਕ ਕਲਿਪ ਵਾਈਜ਼ ਹਨ।

ਆਡੀਓ (Audio): ਜਿਹੜਾ ਤੁਸੀਂ ਖੁਦ ਬਣਾਇਆ ਹੈ ਜਾਂ ਕਿਸੇ ਫਾਈਲ ਤੋਂ ਆਯਾਤ ਕੀਤਾ ਹੈ।

ਆਡੀਓ/ਮਿਊਜ਼ਿਕ (Audio/Music): ਅਲੱਗ ਆਡੀਓ/ਸੰਗੀਤ ਜਾਂ ਵਾਰਤਾ ਇਥੇ ਜੋੜੀ ਜਾਂਦੀ ਹੈ।

ਟਾਈਟਲ ਓਵਰ-ਲੇਅ (Title overlay): ਟਾਈਟਲ ਇਥੇ ਇਕ ਓਵਰ-ਲੇਅ ਵਾਂਗ ਸੋ ਹੁੰਦਾ ਹੈ।

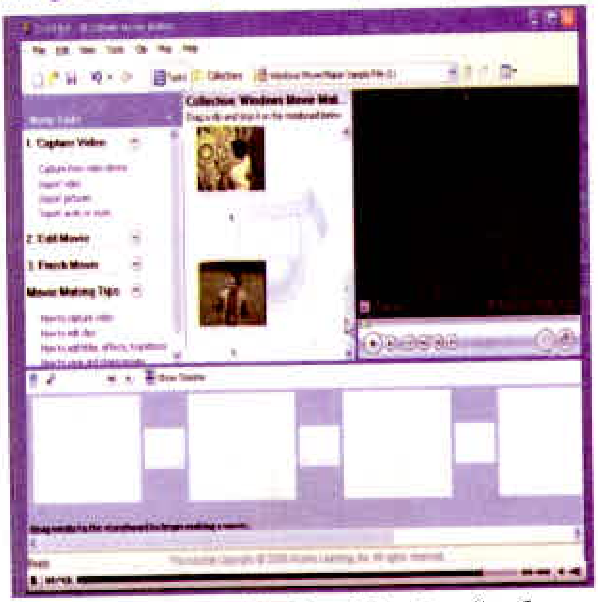
ਇਕ ਸਟੋਰੀ ਬੋਰਡ ਜਾਂ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਕਲਿਪਸ ਜੋੜਣ ਬਾਅਦ ਤੁਸੀਂ ਇਕ ਪ੍ਰੋਜੈਕਟ ਬਣਾਉਣ ਲਈ ਹੇਠ ਲਿਖਿਤ ਕਦਮਾਂ ਦਾ ਅਨੁਸਰਨ ਕਰ ਸਕਦੇ ਹੋ :

- ਜਿਸ ਤਰ੍ਹਾਂ ਤੁਸੀ ਚਾਹੁੰਦੇ ਹੋ ਕਲਿਪਸ ਨੂੰ ਉਸ ਤਰਤੀਬ ਵਿਚ ਲਗਾ ਸਕਦੇ ਹੋ।
- ਕਲਿਪਸ ਦੇ ਵਿਚ ਸੰਬਧ ਬਣਾ ਸਕਦੇ ਹੋ।
- ਆਪਣੀ ਵੀਡੀਓ ਕਲਿਪਸ ਜਾਂ ਤਸਵੀਰਾਂ ਵਿਚ ਵੀਡੀਓ ਇੰਟਰਫੇਸ (interface) ਦੇ ਸਕਦੇ ਹੋ।
- ਕਲਿਪਸ ਨੂੰ ਘਟਾ ਜਾਂ ਛੋਟਾ ਕਰ ਸਕਦੇ ਹੋ। ਉਹਨਾ ਵਿਚੋਂ ਅਣਚਾਹੇ ਹਿੱਸੇ ਨੂੰ ਖਤਮ ਕਰਨ ਲਈ (ਸਿਰਫ ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਵਿਚ)।
- ਕਲਿਪਸ ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜਨਾ ਜਾਂ ਤੋੜਨਾ।

- ਹਰ ਕਲਿਪ ਵਿਚ ਵਾਕ ਰਚਨਾ ਜੋੜਨਾ ਜਿਹੜੀ ਕਿ ਕਲਿਪ ਦੇ ਨਾਲ ਨਾਲ ਹੋਵੇਗੀ (ਸਿਰਫ ਟਾਈਮ ਲਾਈਨ ਵਿਊ ਵਿਚ)।

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਆਡੀਓ ਤੇ ਵੀਡੀਓ ਦੇ ਅੰਸ਼ ਐਡਿਟ ਕਰਨ (ਉੱਪਰ ਲਿਖਿਤ ਸਾਰੇ ਕੰਮ ਕਰਨ) ਤੋਂ ਪਿਛੋਂ ਤੁਸੀਂ ਆਪਣੀ ਤਿਆਰ ਮੂਵੀ ਨੂੰ ਸੇਵ ਕਰ ਸਕਦੇ ਹੋ ਅਤੇ ਆਦਾਨ ਪ੍ਰਦਾਨ ਵੀ ਕਰ ਸਕਦੇ ਹੋ।

ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਵੀਡੀਓ ਕਲਿਪਸ ਜੋੜਨਾ (Adding Videos clips to storyboard)



ਇਸ ਤੋਂ ਪਹਿਲਾਂ ਕਿ ਤੁਸੀਂ ਸ਼ੁਰੂ ਕਰੋ ਇਹ ਸੁਨਿਸ਼ਚਿਤ ਕਰ ਲਵੋ ਕਿ ਤੁਹਾਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਹੇਠਲੇ ਤਲੇ ਤੇ ਨਜ਼ਰ ਆ ਰਿਹਾ ਹੈ। ਜੇਕਰ ਤੁਸੀਂ ਟਾਈਮ ਲਾਈਨ ਨੂੰ ਦੇਖ ਰਹੇ ਹੋ ਤਾਂ ਸ਼ੋ ਸਟੋਰੀ ਬੋਰਡ ਬਟਨ ਨੂੰ ਕਲਿੱਕ (click the show storyboard button) ਕਰੋ।

ਜਦੋਂ ਇਕ ਵਾਰ ਤੁਸੀਂ ਇਹ ਕੰਮ ਕਰ ਲਿਆ ਹੈ ਤਾਂ ਤੁਸੀਂ ਆਪਣੀ ਮੂਵੀ ਵਿਚ ਕਲਿਪ ਜੋੜਨ ਲਈ ਤਿਆਰ ਹੋ। ਤੁਸੀਂ ਸਟੋਰੀ ਬੋਰਡ ਜਾਂ ਟਾਈਮ ਲਾਈਨ ਦਾ ਪ੍ਰਯੋਗ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਸੁਧਾਰਨ ਲਈ ਕਰ ਸਕਦੇ ਹੋ। ਸਟੋਰੀ ਬੋਰਡ ਤੇ ਟਾਈਮ ਲਾਈਨ ਦੋਵੇਂ ਹੀ ਤੁਹਾਨੂੰ ਆਪਣੇ ਕੰਮ ਦੇ ਪ੍ਰੋਸੈਸ ਬਾਰੇ ਦਸਦੇ ਹਨ। ਪਰ ਦੋਵਾਂ ਦਾ ਦੱਸਣ ਦਾ ਤਰੀਕਾ ਜਾਂ ਨਜ਼ਰੀਆ ਵੱਖਰਾ ਹੈ।

- ਸਟੋਰੀ ਬੋਰਡ ਕਲਿਪਸ ਦੀ ਤਰਤੀਬ ਦੱਸਦਾ ਹੈ।
- ਟਾਈਮ ਲਾਈਨ ਕਲਿਪਸ ਦੀ ਟਾਈਮਿੰਗ ਦਸਦਾ ਹੈ।

ਸਟੋਰੀ ਬੋਰਡ ਵਿਊ (storyboard view) ਵਿਚ ਤਸਵੀਰ ਤੇ ਕਲਿੱਕ ਕਰ ਕੇ ਖਿੱਚ ਕੇ ਪਹਿਲੇ ਫਰੇਮ ਤੱਕ ਲਿਆ ਸਕਦੇ ਹਾਂ।

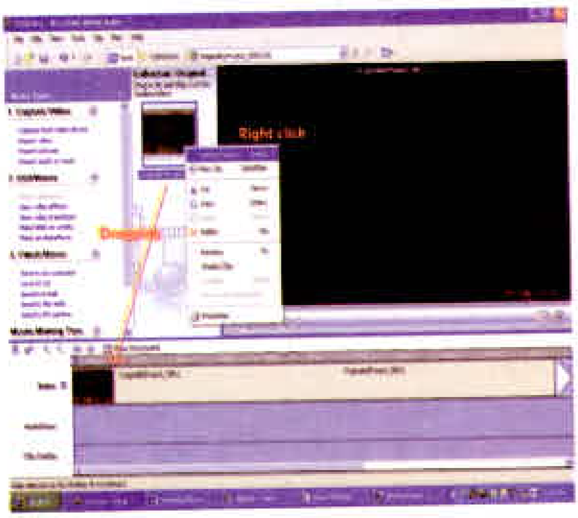


Fig: Showing Drag and Drop image process

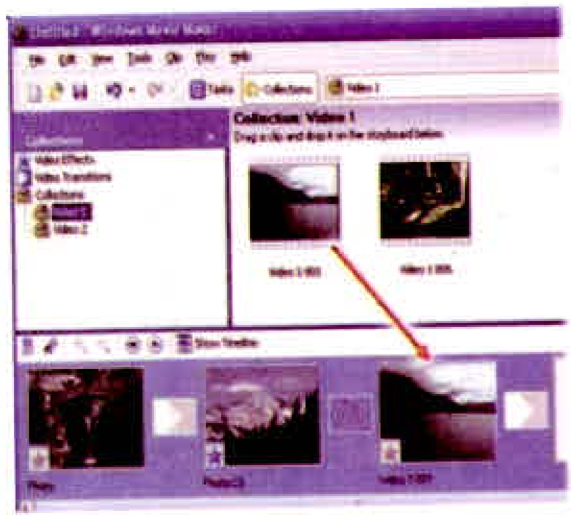


Fig: Showing 3 clips drop from contents pane

1 ਇਹ ਤਸਵੀਰਾਂ ਦੀ ਤਰਤੀਬ-ਵਾਰ ਕਰਨ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਉਦੋਂ ਤੱਕ ਜਾਰੀ ਰਹਿੰਦੀ ਹੈ ਜਦੋਂ ਤੱਕ ਇਹ ਉਸ ਤਰਤੀਬ ਵਿਚ ਨਹੀਂ ਆ ਜਾਂਦੀਆਂ ਜਿਸ ਵਿਚ ਤੁਸੀਂ ਉਸ ਨੂੰ ਰੱਖਣਾ ਚਾਹੁੰਦੇ ਹੋ। ਤੁਸੀਂ ਇਹਨਾਂ ਤਸਵੀਰਾਂ ਨੂੰ ਖਿੱਚ ਕੇ ਇਕ ਦੂਜੇ ਅੱਗੇ ਜਾਂ ਪਿਛੇ ਵੀ ਰੱਖ ਸਕਦੇ ਹੋ ਅਤੇ ਤੁਸੀਂ ਉਹਨਾਂ ਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਤੋਂ ਇਕਠੇ ਵੀ ਡਿਲੀਟ ਕਰ ਸਕਦੇ ਹੋ।

2 ਹੁਣ ਪਹਿਲੀ ਤਸਵੀਰ ਨੂੰ ਕਲਿੱਕ ਕਰੋ ਤੇ ਪਲੇਅਰ ਵਿੰਡੋ ਵਿਚ ਪਲੇਅ ਤੇ ਕਲਿੱਕ ਕਰ ਕੇ ਆਪਣੀ ਫਿਲਮ ਦੇਖ ਸਕਦੇ ਹੋ।

3 ਆਪਣੀ ਫਿਲਮ ਨੂੰ ਡੈਸਕਟੌਪ ਉੱਤੇ ਫੋਲਡਰ ਵਿਚ ਸੁਰੱਖਿਅਤ ਕਰੋ।

ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਦੁਬਾਰਾ ਤੋਂ ਤਰਤੀਬਵਾਰ ਕਰਨਾ (Rearranging Sequence in Stroyboard)

ਜਦੋਂ ਸਾਰੀ ਕਲਿਪਸ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਆ ਜਾਣ ਤਾਂ ਉਹਨਾਂ ਨੂੰ ਦੁਬਾਰਾ ਤੋਂ ਤਰਤੀਬ ਵਾਰ ਕਰਨਾ : ਜਦੋਂ ਸਾਰੀ ਕਲਿਪਸ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਆ ਜਾਣ ਤਾਂ ਉਹਨਾਂ ਨੂੰ ਦੁਬਾਰਾ ਤੋਂ ਤਰਤੀਬਵਾਰ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ। ਇਹ ਕਰਨ ਲਈ ਕਲਿਪ ਉਤੇ ਖੱਬਾ ਕਲਿੱਕ ਕਰੋ ਤੇ ਫਿਰ ਉਸਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਦੀ ਨਵੀਂ ਜਗਾ ਤੇ ਲੈ ਕੇ ਜਾਓ ਜਿਥੇ ਤੁਸੀ ਉਸਨੂੰ ਰਖਣਾ ਚਾਹੁੰਦੇ ਹੋ।

ਟਾਈਮ ਲਾਈਨ ਦਾ ਪ੍ਰਯੋਗ ਕਰਦੇ ਹੋਏ ਕਲਿਪਸ ਤੋੜਨਾ (Split clips using timeline)

ਪਿਛਲੇ ਅਧਿਆਇ ਵਿਚ ਅਸੀ ਸਕਰੀਨ ਉੱਪਰ ਸਪਲਿਟ (split) ਬਟਨ ਦਾ ਪ੍ਰਯੋਗ ਕਰਦੇ ਹੋਏ ਵੀਡੀਓ ਨੂੰ ਤੋੜਣ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਵੇਖੀ ਸੀ। ਇਥੇ ਅਸੀਂ ਦੇਖਾਂਗੇ ਕਿ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਕਿਸ ਤਰ੍ਹਾਂ ਵੀਡੀਓ ਨੂੰ ਅਲੱਗ-ਅਲੱਗ ਭਾਗਾਂ ਜਾਂ ਕਲਿਪਸ ਵਿਚ ਤੋੜਦੇ ਹਾਂ ਤਾਂ ਕਿ ਅਸੀ ਉਹਨਾਂ ਕਲਿਪਸ ਨੂੰ ਦੁਬਾਰਾ ਤੋਂ ਤਰਤੀਬ ਵਾਰ ਕਰ ਕੇ ਇਕ ਨਵਾਂ ਵੀਡੀਓ ਪ੍ਰਾਪਤ ਕਰ ਸਕੀਏ। ਇਸ ਲਈ ਹੇਠ ਲਿਖਿਤ ਕਦਮਾਂ ਦਾ ਅਨੁਸਰਣ ਕਰਾਂਗੇ।

ਇਕ ਵੀਡੀਓ ਨੂੰ ਤੋੜਣ ਲਈ

- ਵੀਡੀਓ ਤੇ ਸੱਜਾ ਕਲਿੱਕ ਕਰੋ ਅਤੇ Choose -> play timeline ਪਲੇਅ ਟਾਈਮ ਲਾਈਨ ਚੁਣੋ। ਜਦੋਂ ਵੀਡੀਓ ਚਲਨਾ ਸ਼ੁਰੂ ਹੋ ਜਾਂਦਾ ਹੈ, ਨੀਲੀ ਲਾਈਨ ਨੂੰ ਅੱਗੇ ਵਧਣ ਵਾਲੀ ਦਿਸ਼ਾ ਵਿਚ ਮੂਵ ਹੋਣ ਦਿਉ।
- ਕਲਿਪ ਦਾ ਪ੍ਰਯੋਗ ਕਰੋ - ਤੋੜੋ, (Clip ->split) ਆਉਟਪੁੱਟ ਹੇਠਾਂ ਦਰਸਾਏ ਵਾਂਗ ਹੋਵੇਗੀ
- ਤਸਵੀਰ ਨੂੰ ਸੇਵ ਕਰੋ।

Fig: Showing two videos without split

Fig: Showing two videos with split on first using the above said process

ਕਲਿਪਸ ਦੀ ਟਰਿਮਿੰਗ (Trimming the Clips)

ਇਕ ਫਾਈਲ ਜਾਂ ਕਲਿਪ ਦੇ ਹਿੱਸਿਆਂ ਨੂੰ ਅਸਲੀ ਸ੍ਰੋਤ ਤੋ ਡਿਲੀਟ ਕੀਤੇ ਬਗੈਰ ਛੁਪਾਣਾ। ਫਾਈਲ ਜਾਂ ਕਲਿਪਸ ਨੂੰ ਉਹਨਾਂ ਦੇ ਸ਼ੁਰੂ (start points) ਜਾਂ ਅੰਤ (end points) ਪੁਆਇੰਟ ਨੂੰ ਅਡੱਜਸਟ ਕਰਦੇ ਹੋਏ ਛੋਟਾ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

ਕਲਿਪਸ ਦੀ ਛਟਾਈ ਕਰਨਾ

ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਤੁਹਾਡੇ ਵੀਡੀਓ ਕਲਿਪਸ ਨੂੰ ਤਰਤੀਬਵਾਰ ਰਖਣ ਤੇ ਬਾਅਦ ਤੁਸੀਂ ਕਲਿਪਸ ਦੀ ਲੰਬਾਈ ਨੂੰ ਵੀ ਨਿਰਧਾਰਿਤ ਕਰ ਸਕਦੇ ਹੋ। ਇਹ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਵੀਡੀਓ ਕਲਿਪਸ ਦੀ ਛਟਾਈ ਕਰ ਕੇ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।

ਕਲਿਪਸ ਨੂੰ ਛਟਾਈ ਕਰਕੇ ਤੁਸੀਂ ਇਕ ਫਾਈਲ ਜਾਂ ਕਲਿਪ ਦੇ ਅਣਚਾਹੇ ਹਿੱਸੇ ਨੂੰ (ਜੋ ਕਿ ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਹੋ ਸਕਦਾ ਹੈ) ਛੁਪਾ ਸਕਦੇ ਹੋ। ਇਹ ਕ੍ਰਮ ਦਰਸ਼ਕਾਂ ਤੋਂ ਛੁਪ ਜਾਂਦੇ ਹਨ ਅਤੇ ਇਹ ਤੁਹਾਡੇ ਫਾਈਨਲ ਪ੍ਰੋਜੈਕਟ ਅਤੇ ਫਾਈਨਲ ਸੇਵਡ ਫਿਲਮ ਤੋ ਛੁਪ ਜਾਂਦੇ ਹਨ। ਤੁਸੀ ਟਰਿਮ ਹੈਂਡਲਜ਼ (trim handles) ਵੀ ਖਿੱਚ ਸਕਦੇ ਹੋ ਜੋ ਕਿ ਅੱਗੇ ਲਿਖੀ ਮਿਸਾਲ ਵਿਚ ਦਰਸਾਏ ਗਏ ਹਨ ਅਤੇ ਇਹ ਕਲਿਪ ਦੇ ਅਣਚਾਹੇ ਭਾਗ ਦੇ ਹਿੱਸਿਆਂ ਨੂੰ ਛਾਂਟ ਦੇਣਗੇ।

Fig: ਕਲਿਪ ਦੀ ਸੱਜੀ ਧੁਰੀ ਤੇ ਐਰੋ ਹੈਡ ਨੂੰ ਲੈ ਕੇ ਜਾਓ. ਇਕ ਦੋਹਰੀ ਲਾਲ ਐਰੋ ਹੈਡ ਦਿਖੇਗੀ ਜਿਵੇ ਪਹਿਲੀ ਤਸਵੀਰ ਵਿਚ ਦਿਖਾਇਆ ਗਿਆ ਇਥੇ ਕਲਿੱਕ ਕਰੋ ਤੇ ਟਰਿਮ ਕਰਨ ਲਈ ਖਿੱਚੋ ਜਿਵੇ ਦੂਜੀ ਤਸਵੀਰ ਵਿਚ ਦਿਖਾਇਆ ਗਿਆ ਹੈ।

ਇਕ ਕਲਿਪ ਨੂੰ ਟਰਿਮ ਜਾਂ ਛਾਂਟੀ ਕਰਨ ਲਈ :

1. ਵਿਊ ਮੀਨੂੰ (View menu) ਵਿਚ ਟਾਈਮ ਲਾਈਨ (Timeline) ਤੇ ਕਲਿੱਕ ਕਰੋ। ਜੇਕਰ ਸਟੋਰੀ ਬੋਰਡ ਹੈ ਤਾਂ ਟਾਈਮ ਲਾਈਨ (Timeline) ਤੇ ਲਿਆਉਣ ਲਈ ਕਲਿੱਕ ਕਰੋ।
2. ਕੋਲੈਕਸ਼ਨ ਬਾਕਸ ਵਿਚ ਉਸ ਕੋਲੈਕਸ਼ਨ ਤੇ ਕਲਿੱਕ ਕਰੋ ਜਿਸ ਵਿਚ ਉਹ ਕਲਿਪ ਪਈ ਹੈ ਜੋ ਤੁਸੀ ਐਡ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਅਤੇ ਫਿਰ ਕਾਂਨਟੈਂਟਸ ਪੇਨ ਜਾਂ ਬਾਕਸ ਵਿਚ ਉਸ ਕਲਿਪ ਤੇ ਕਲਿਪ ਕਰੋ ਜਿਸ ਨੂੰ ਤੁਸੀ ਟਰਿਮ ਜਾਂ ਛਾਂਟੀ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ।
3. ਕਲਿਪ ਮੀਨੂੰ (Clip menu) ਵਿਚ ਐਡ ਟੂ ਟਾਈਮ ਲਾਈਨ (Add to Timeline) ਤੇ ਕਲਿੱਕ ਕਰੋ।
4. ਟਾਈਮ ਲਾਈਨ (Timeline) ਵਿਚ ਤੁਸੀ ਜਿਸ ਕਲਿਪ ਨੂੰ ਟਰਿਮ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਨੂੰ ਚੁਣੋ।
5. ਟਾਈਮ ਲਾਈਨ (Timeline) ਵਿਚ ਨੀਲੀ ਲਾਈਨ ਨੂੰ ਕਲਿੱਕ ਕਰੋ ਅਤੇ ਇਸ ਨੂੰ ਉੱਥੇ ਤੱਕ ਖਿਚੋ ਜਿਥੇ ਤਕ ਤੁਸੀ ਕਲਿਪ ਨੂੰ ਟਰਿਮ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ। ਜਿਵੇਂ ਕਿ ਉੱਪਰ ਦਿਖਾਇਆ ਗਿਆ ਹੈ।
6. ਹੇਠ ਲਿਖੇ ਵਾਂਗ ਕਰੋ :
   - ਜਦੋ ਨੀਲੀ ਲਾਈਨ ਉਸ ਪੁਆਇੰਟ ਤੇ ਹੋ ਜਿੱਥੇ ਤੁਸੀ ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਕਲਿਪ ਨੂੰ ਪਲੇਅ ਬੈਕ ਸ਼ੁਰੂ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਤਾਂ ਕਲਿਪ ਮੀਨੂੰ (clip menu) ਵਿਚ ਸੈੱਟ ਸਟਾਰਟ ਟਰਿਮ ਪੁਆਇੰਟ (set start trim point) ਤੇ ਕਲਿੱਕ ਕਰੋ।

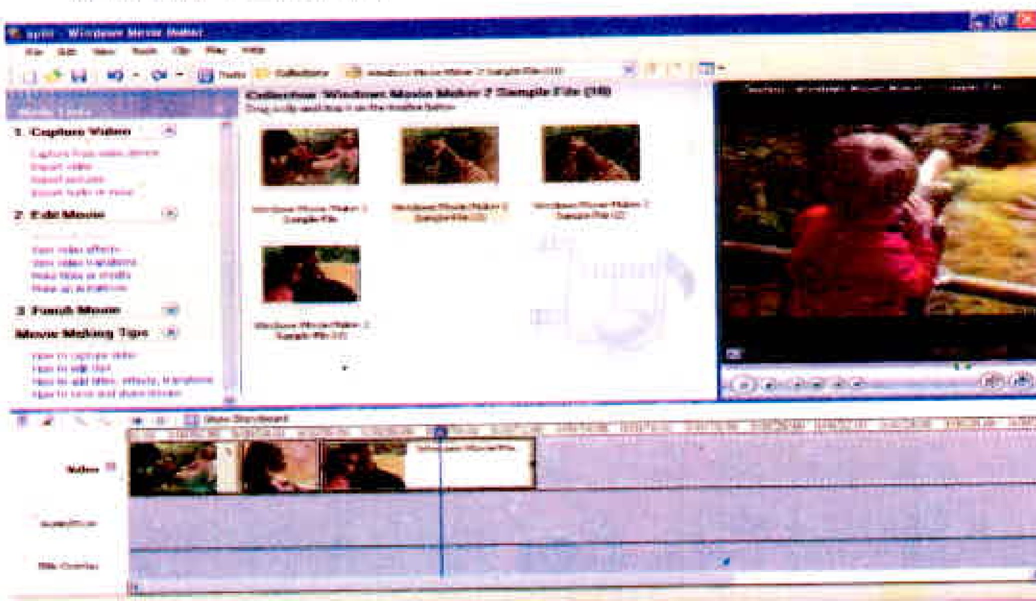


Fig: ਸਟੋਰੀ ਬੋਰਡ ਤੇ 3 ਕਲਿਪ ਦਿਖਾ ਰਹੀ ਹੈ ਤੇ ਤੀਜਾ ਨੀਲੀ ਲਾਈਨ ਦੇ ਨਿਸ਼ਾਨ ਤਕ ਟਰਿਮ ਕਰ ਦਿਤਾ ਹੋਇਆ ਹੈ

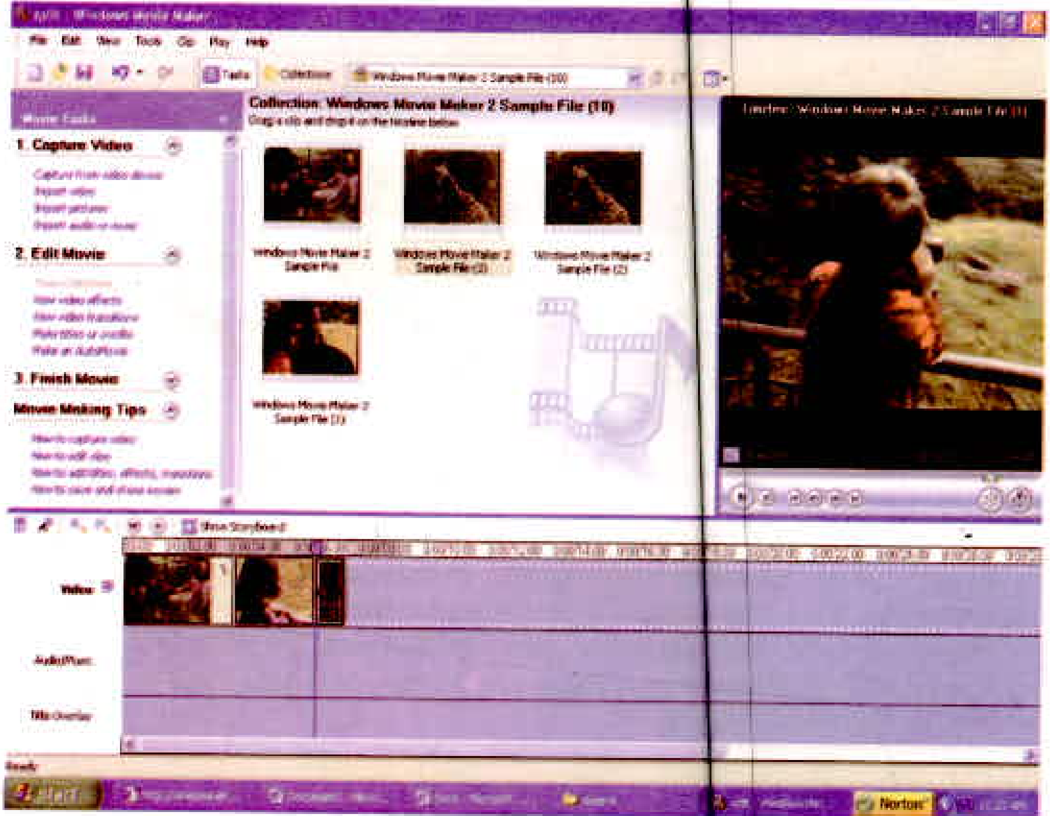


Fig: ਕਲਿਪ - ਸੈਟ ਸਟਾਰਟ ਟਰਿਮ ਪੁਆਇੰਟ ਦੀ ਨਤੀਜਨ ਸਕਰੀਨ

- ਜਦੋਂ ਨੀਲੀ ਲਾਈਨ ਦਾ ਸੰਕੇਤਕ ਉਥੇ ਹੁੰਦਾ ਹੈ ਜਿਥੋਂ ਤੁਸੀ ਵੀਡੀਓ ਜਾਂ ਆਡੀਓ ਕਲਿਪ ਨੂੰ ਚਲਣ ਤੋਂ ਰੋਕਣਾ ਚਾਹੁੰਦੇ ਹੋ ਕਲਿਪ ਮੀਨੂੰ (clip menu) ਦੇ ਸੈਟ ਐਂਡ ਟਰਿਮ ਪੁਆਇੰਟ (Set End trim point) ਤੇ ਕਲਿੱਕ ਕਰੋ।

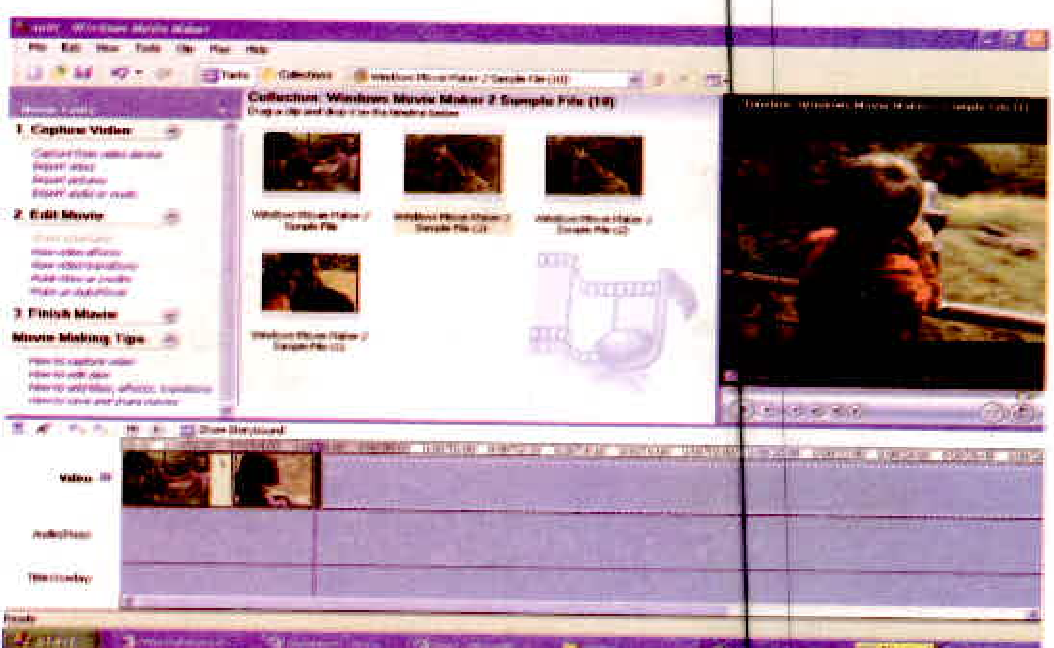


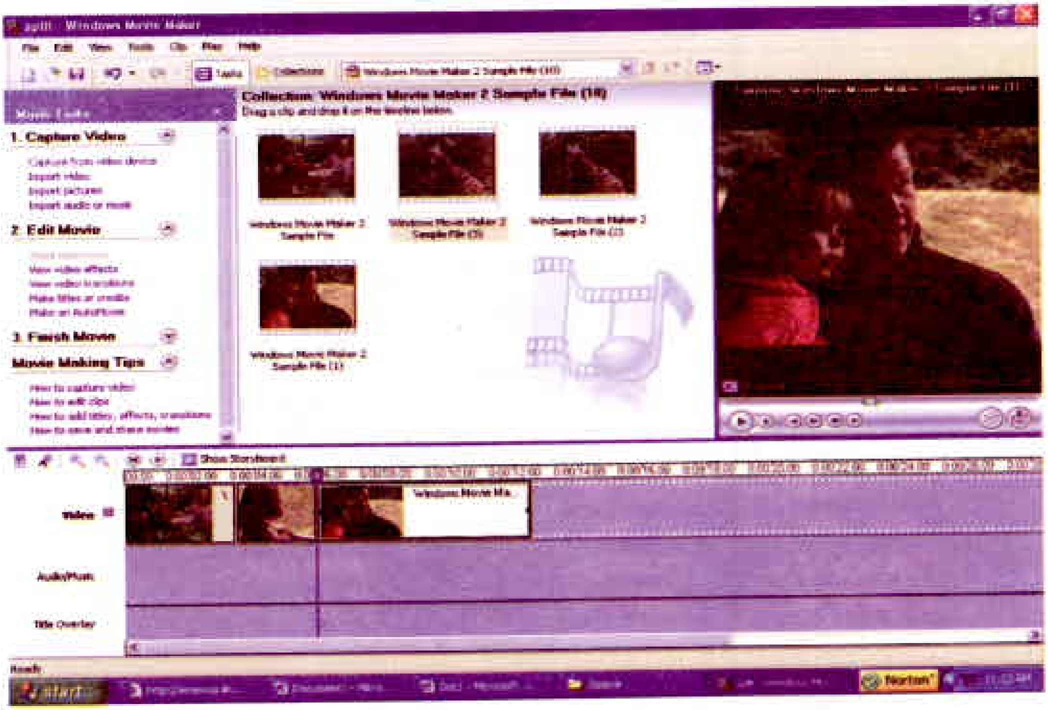
Fig: ਕਲਿਪ ਦੇ ਟਰਿਮ ਹਿਸੇ ਦਿਖਾ ਰਹੀ ਹੈ।

- ਹੁਣ ਨੀਲੀ ਲਾਈਨ ਨੂੰ ਉਥੋਂ ਤਕ ਲੈ ਕੇ ਆਉ ਜਿਥੇ ਤੁਸੀ ਟਰਿਮਿੰਗ ਜਾਂ ਛਾਂਟੀ ਨੂੰ ਬੰਦ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ। (ਤੁਸੀ ਮੋਨੀਟਰ ਜਾਂ ਮੂਵੀ ਵਿੰਡੋ ਵਿਚ ਇਸ ਨੂੰ ਜਿਆਦਾ ਸਪਸ਼ਟ ਤਰੀਕੇ ਨਾਲ ਦੇਖ ਸਕਦੇ ਹੋ)। ਹੁਣ ਕਲਿਪ ਸੈੱਟ ਅਤੇ ਟਰਿਮ ਪੁਆਇੰਟ ਆਪਸ਼ਨ ਚੁਣੋ। ਹੁਣ ਤੁਹਾਡੇ ਕੋਲ ਵੀਡੀਓ ਵਿਚੋਂ ਸਿਰਫ ਦਿਖਾਏ ਜਾਣ ਵਾਲਾ ਕਲਿਪ ਹੈ।

ਟਰਿਮ ਪੁਆਇੰਟ ਨੂੰ ਸਾਫ ਕਰਨਾ (To clear the trim points)

1 ਟਾਈਮ ਲਾਈਨ ਉੱਪਰ ਪ੍ਰੋਜੈਕਟ ਓਪਨ ਕਰਨ ਲਈ ਟਰਿਮਡ ਕਲਿਪ ਤੇ ਕਲਿੱਕ ਕਰੋ ਜਿਸ ਤਰ੍ਹਾਂ ਉੱਪਰ ਦਿਖਾਇਆ ਗਿਆ ਹੈ

2 ਕਲਿਪ ਮੀਨੂੰ (clip menu)ਤੇ ਟਰਿਮ ਪੁਆਇੰਟਸ ਨੂੰ ਕਲਿੱਕ (Clear trim points) ਕਰੋ



ਟੁੱਟੇ ਹੋਏ ਵੀਡੀਓ ਕਲਿਪਸ ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜਨਾ (To combine a split video clip)

ਤੁਹਾਡੇ ਕੋਲ ਇਕ ਫਾਈਲ ਜਿਸ ਵਿਚ ਅਲੱਗ-ਅਲੱਗ ਛੋਟੇ-ਛੋਟੇ ਕਲਿਪਸ ਜਾਂ ਇਕੋ ਤਰ੍ਹਾਂ ਦੇ ਲਗਾਤਾਰ ਕਲਿਪਸ ਹਨ ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜ ਸਕਦੇ ਹੋ। ਲਗਾਤਾਰ ਦਾ ਮਤਲੱਬ ਹੈ ਕਿ ਜਿਥੇ ਪਹਿਲੇ ਕਲਿਪ ਦਾ ਟਾਈਮ ਖ਼ਤਮ ਹੁੰਦਾ ਹੈ, ਦੂਜੇ ਦਾ ਸ਼ੁਰੂ ਹੋ ਜਾਂਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ ਜੇਕਰ ਤੁਹਾਡੇ ਕੋਲ ਆਡੀਓ ਜਾਂ ਵੀਡੀਓ ਫਾਈਲ ਹੈ ਜੋ ਕਲਿਪਸ ਵਿਚ ਵੰਡੀ ਹੋਈ ਹੈ ਅਤੇ ਇਹ ਵਿੰਡੋ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਲਿਆਈ ਜਾ ਸਕਦੀ ਹੈ ਅਤੇ ਕਲਿਪਸ ਦੇ ਨਾਮ ; ਕਲਿਪ 1, ਕਲਿਪ 2 ਤੇ ਕਲਿਪ 3 ਹਨ, ਇਸ ਤਰ੍ਹਾਂ ਤੁਸੀਂ ਕਲਿਪ 1 ਅਤੇ ਕਲਿਪ 2 ਜਾਂ ਕਲਿਪ 2 ਅਤੇ ਕਲਿਪ 3 ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜ ਸਕਦੇ ਹੋ ਪ੍ਰੰਤੂ ਤੁਸੀਂ ਕਲਿਪ 1 ਅਤੇ ਕਲਿਪ 3 ਨੂੰ ਆਪਸ ਵਿਚ ਨਹੀਂ ਜੋੜ ਸਕਦੇ ਕਿਉਂਕਿ ਉਹ ਕਤਾਰ ਵਿਚ ਨਹੀ ਹਨ। ਤੁਸੀ ਲਗਾਤਾਰ ਵੀਡੀਓ ਕਲਿਪਸ ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜ ਸਕਦੇ ਹੋ। ਕਲਿਪ ਨੂੰ ਤੋੜਣ ਵਾਸਤੇ ਤੁਸੀ ਇਸ ਨੂੰ ਚਲਾਉ ਅਤੇ ਜਿਸ ਪੁਆਇੰਟ ਤੋਂ ਇਸ ਨੂੰ ਤੋੜਨਾ ਚਾਹੁੰਦੇ ਹੋ ਉਥੇ ਕਲਿੱਕ ਕਰੋ ਅਤੇ ਕਲਿਪ ਵਿਚੋਂ ਸਪਲਿਟ (split) ਅਤੇ ਇਸ ਨੂੰ ਅਸਲ ਨਾਲ ਮਿਲਾਉਣ ਲਈ ਹੇਠ ਲਿਖਿਆਂ ਵਿਚੋਂ ਇਕ ਕਰੋ :

1 ਕਾਨਟੈਂਟ ਬੱਾਕਸ ਵਿਚ ਜਾਂ ਸਟੋਰੀ ਬੋਰਡ/ਟਾਈਮ ਲਾਈਨ **(stroyboard / time line)** ਵਿਚ CTRL-ਕੀ ਨੂੰ ਪਕੜੇ ਰਖੋ ਅਤੇ ਲਗਾਤਾਰ ਕਲਿਪਸ ਨੂੰ ਚੁਣੋ ਜਿਹਨਾਂ ਨੂੰ ਆਪਸ ਵਿਚ ਜੋੜਨਾ ਚਾਹੁੰਦੇ ਹੋ।

2 ਹੁਣ ਕਲਿਪ ਮੀਨੂੰ ਵਿਚ ਕੰਬਾਈਨ ਤੇ ਕਲਿੱਕ ਕਰੋ, ਨਾਮ ਤੇ ਗੁਣ ਸੰਬੰਧੀ ਸੂਚਨਾ ਗਰੁੱਪ ਵਿਚ ਪਹਿਲੀ ਕਲਿਪ ਦੀ ਨਵੇਂ ਕਲਿਪ ਲਈ ਵਰਤੀ ਜਾਂਦੀ ਹੈ ਅਤੇ ਟਾਈਮ ਉਹਨਾਂ ਦੇ ਅਨੁਸਾਰ ਸੈਟ ਹੋ ਜਾਵੇਗਾ।

ਮੂਵੀ ਨੂੰ ਐਡਿਟ ਕਰਨਾ (Editing your Movie)

ਇਫੈਕਟਸ, ਟਰਾਂਜ਼ੀਸ਼ਨਜ਼ ਤੇ ਟਾਈਟਲ ਨੂੰ ਐਡ ਕਰਨਾ (**Adding Effects, Transitions & Titles**): ਮੂਵੀ ਮੇਕਰ ਤੁਹਾਨੂੰ ਆਪਣੀ ਮੂਵੀ ਵਿਚ ਇਫੈਕਟਸ ਅਤੇ ਟਾਈਟਲ ਪਾਉਣ ਦੀ ਸੁਵਿਧਾ ਦਿੰਦਾ ਹੈ ਇਫੈਕਟਸ ਤਸਵੀਰਾਂ ਅਤੇ ਕਲਿਪਸ ਉੱਪਰ ਲਗਾਏ (apply) ਜਾ ਸਕਦੇ ਹਨ। ਟਰਾਂਜ਼ਿਸ਼ਨ ਤਸਵੀਰਾਂ ਅਤੇ ਕਲਿਪਸ ਦੇ ਵਿਚਕਾਰ ਜੋੜੇ ਜਾ ਸਕਦੇ ਹਨ। ਟਾਈਟਲਜ਼ ਮੂਵੀ ਜਾਂ ਮੂਵੀ ਦੇ ਕਲਿਪਸ ਦੇ ਸ਼ੁਰੂ ਅਤੇ ਅੰਤ ਵਿਚ ਜੋੜੇ ਜਾ ਸਕਦੇ ਹਨ। ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ, ਵੀਡੀਓ ਪ੍ਰਭਾਵ ਅਤੇ ਟਾਈਟਲਜ਼ ਦਾ ਪ੍ਰਯੋਗ ਕਰ ਕੇ ਤੁਸੀਂ ਹੇਠ ਲਿਖੇ ਤੱਤ ਐਡ ਕਰ ਕੇ ਆਪਣੀ ਮੂਵੀ ਨੂੰ ਵਧੇਰੇ ਆਕਰਸ਼ਿਤ ਬਣਾ ਸਕਦੇ ਹੋ :

ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ (Video Transistion)

ਇਕ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਤੁਹਾਡੀ ਮੂਵੀ ਦੀ ਇਕ ਤੋਂ ਬਾਅਦ ਇਕ ਕਲਿਪ ਨੂੰ ਚਲਾਉਣ ਲਈ ਨਿਰਧਾਰਿਤ ਕਰਦੀ ਹੈ। ਤੁਸੀਂ ਸਟੋਰੀ ਬੋਰਡ ਜਾਂ ਟਾਈਮ ਲਾਈਨ ਉੱਤੇ ਦੋ ਤਸਵੀਰਾਂ, ਵੀਡੀਓ ਕਲਿਪਸ ਜਾਂ ਟਾਈਟਲਸ ਦੇ ਵਿਚ ਇਕ ਟਰਾਂਜ਼ਿਸ਼ਨ ਨੂੰ ਐਡ ਕਰ ਸਕਦੇ ਹੋ। ਟਰਾਂਜ਼ਿਸ਼ਨ ਇਕ ਕਲਿਪ ਦੇ ਖਤੱਮ ਹੋਣ ਤੇ ਦੂਜੇ ਦੇ ਸ਼ੁਰੂ ਹੋਣ ਦੇ ਵਿਚਕਾਰ ਚਲਦੀ ਹੈ। ਵਿੰਡੋ ਮੂਵੀ ਮੇਕਰ ਦੇ ਵਿਚ ਕਈ ਤਰ੍ਹਾਂ ਦੇ ਟਰਾਂਜ਼ਿਸ਼ਨ ਹੁੰਦੇ ਹਨ ਜੋ ਤੁਸੀਂ ਆਪਣੇ ਪ੍ਰੋਜੈਕਟ ਵਿਚ ਸ਼ਾਮਿਲ ਕਰ ਸਕਦੇ ਹੋ। ਟਰਾਂਜ਼ਿਸ਼ਨ ਨੂੰ ਕੋਲੈਕਸ਼ਨ ਬਾਕਸ ਵਿਚ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਦੇ ਨਾਂ ਦੇ ਫੋਲਡਰ ਵਿਚ ਰੱਖਿਆ ਜਾਂਦਾ ਹੈ।

ਵੀਡੀਓ ਇਫੈਕਟਸ (Video Effects)

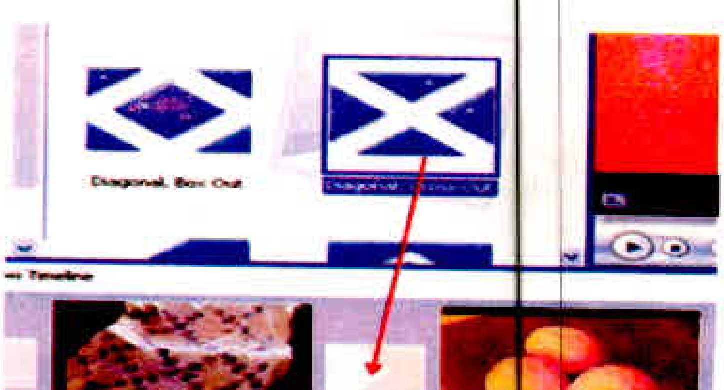
ਇਕ ਵੀਡੀਓ ਇਫੈਕਟ ਇਹ ਨਿਰਧਾਰਿਤ ਕਰਦਾ ਹੈ ਕਿ ਤੁਹਾਡੇ ਪ੍ਰੋਜੈਕਟ ਤੇ ਮੂਵੀ ਵਿਚ ਵੀਡੀਓ ਕਲਿਪ, ਤਸਵੀਰਾਂ ਅਤੇ ਟਾਈਟਲ ਕਿਵੇਂ ਦਰਸਾਏ ਜਾਣਗੇ । ਵੀਡੀਓ ਇਫੈਕਟਸ ਤੁਹਾਡੀ ਮੂਵੀ ਵਿਚ ਖਾਸ ਇਫੈਕਟਸ ਨੂੰ ਜੋੜਦੇ ਹਨ। ਇਕ ਵੀਡੀਓ ਇਫੈਕਟਸ ਸਾਰੀ ਮੂਵੀ ਉੱਤੇ ਕਲਿਪ, ਤਸਵੀਰ ਅਤੇ ਟਾਈਟਲ ਨੂੰ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਦੇ ਹਨ। ਤੁਸੀਂ ਕੋਲੈਕਸ਼ਨ ਬਾਕਸ ਵਿਚ ਵੀਡੀਓ ਫੋਲਡਰ ਵਿਚੋਂ ਕੋਈ ਵੀ ਇਫੈਕਟਸ ਆਪਣੀ ਮੂਵੀ ਨੂੰ ਦੇ ਸਕਦੇ ਹੋ।

ਟਾਈਟਲ ਅਤੇ ਕਰੈਡਿਟਸ (Titles and Credits)

ਇਹ ਤੁਹਾਡੀ ਮੂਵੀ ਨੂੰ ਟੈਕਸਟ ਅਤੇ ਟਾਈਟਲ ਇਫੈਕਟਸ ਦੇਣ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ। ਤੁਸੀਂ ਕੋਈ ਵੀ ਟੈਕਸਟ, ਟਾਈਟਲ, ਨਾਮ ਅਤੇ ਤਰੀਖ ਆਦਿ ਮੂਵੀ ਵਿਚ ਸ਼ਾਮਿਲ ਕਰ ਸਕਦੇ ਹੋ। ਤੁਸੀਂ ਟਾਈਟਲ ਤੇ ਕਰੈਡਿਟ ਦੀ ਅਪੀਅਰੈਂਸ (appearance) ਨੂੰ ਬਦਲ ਵੀ ਸਕਦੇ ਹੋ। ਇਸ ਤੋਂ ਇਲਾਵਾ ਟਾਈਟਲ ਦੀ ਐਨੀਮੇਸ਼ਨ ਨੂੰ ਵੀ ਬਦਲਿਆ ਜਾ ਸਕਦਾ ਹੈ ਜੋ ਤੁਹਾਡੀ ਮੂਵੀ ਵਿਚ ਟਾਈਟਲ ਅਤੇ ਕਰੈਡਿਟ ਦੀ ਪ੍ਰਦਰਸ਼ਿਤਾ ਨੂੰ ਨਿਰਧਾਰਿਤ ਕਰਦੀ ਹੈ।ਜਦੋਂ ਵੀ ਤੁਸੀਂ ਇਕ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਨੂੰ ਚੁਣਦੇ ਹੋ ਤਾਂ ਸਟੋਰੀ ਬੋਰਡ ਤੋਂ ਇਸ ਨੂੰ ਦੋ ਤਸਵੀਰਾਂ ਦੇ ਵਿਚ ਖਿੱਚੋ।

ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਨੂੰ ਲਾਗੂ ਕਰਨਾ (Apply Video Transistion)

ਕਿਸੇ ਵੀ ਤਸਵੀਰ ਵਿਚ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਨੂੰ ਲਾਗੂ (apply) ਕਰਨ ਲਈ ਹੇਠਾਂ ਦਿਤੀ ਪ੍ਰਕ੍ਰਿਆ ਨੂੰ ਦੁਹਰਾਓ



Drag the chosen transition between the two pictures

Fig. ਇਕ ਐਡਿਡ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਨਾਲ ਇਕ ਟਾਈਮ ਲਾਈਨ

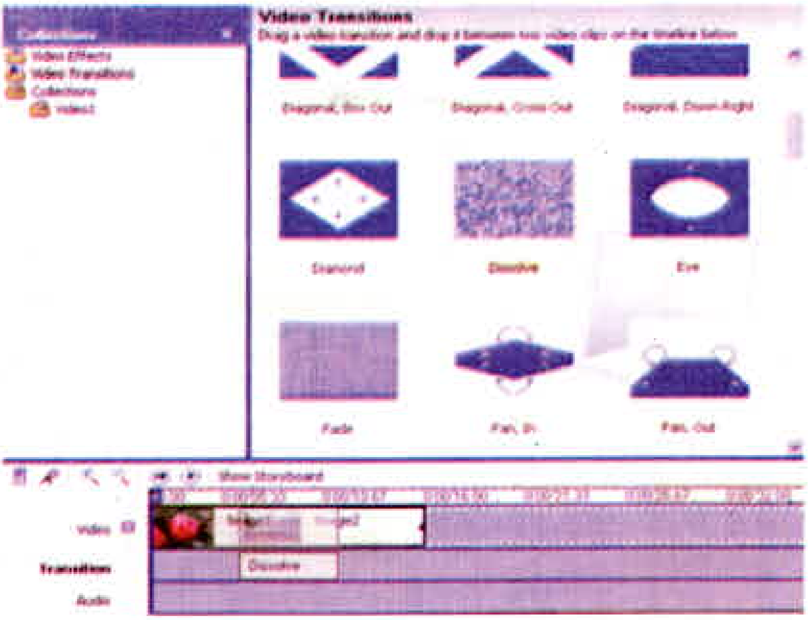


Fig. ਸਟੋਰੀ ਬੋਰਡ ਨਾਲ ਇਕ ਵੀਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਦਾ ਪ੍ਰੋਜੇਕਟ

ਤੁਸੀ ਸਾਰੀਆਂ ਤਸਵੀਰਾਂ ਲਈ ਉਹੀ ਜਾਂ ਦੂਸਰੀ ਟਰਾਂਜ਼ਿਸ਼ਨ ਦਾ ਪ੍ਰਯੋਗ ਕਰ ਸਕਦੇ ਹੋ। ਆਪਣੀ ਮੂਵੀ ਦੇਖੋ।

You can apply different transitions

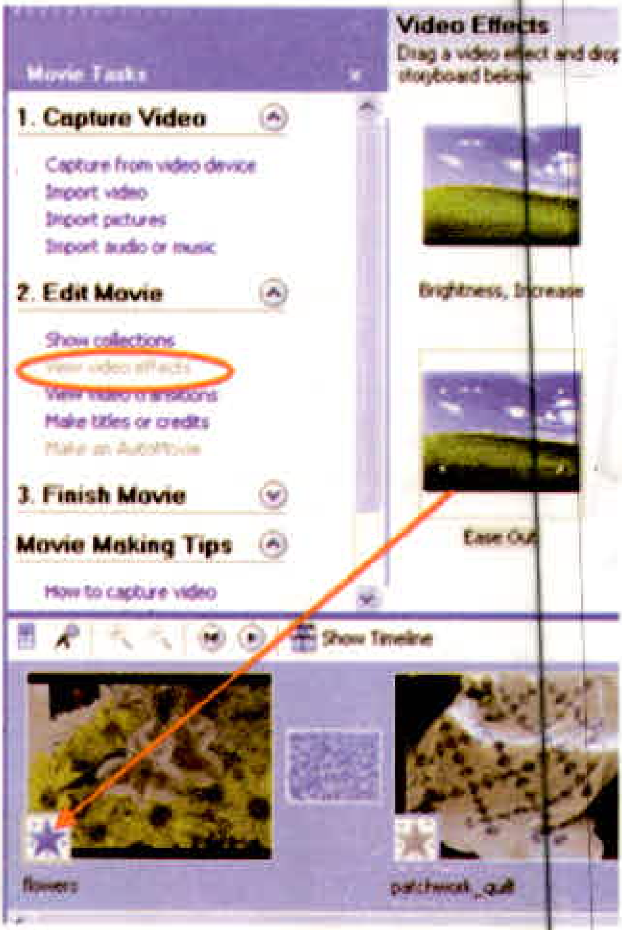
ਤੁਸੀ ਅਲੱਗ-ਅਲੱਗ ਟਰਾਂਜ਼ਿਸ਼ਨ ਵੀ ਲਾਗੂ ਕਰ ਸਕਦੇ ਹੋ।

ਵੀਡੀਓ ਇਫੈਕਟਸ ਜੋੜਨਾ (Adding video effects)

ਤੁਸੀ ਮੂਵੀ ਵਿਚ ਵੀਡੀਓ ਇਫੈਕਟ (effect) ਵੀ ਭਰ ਸਕਦੇ ਹੋ। ਵੀਡੀਓ ਇਫੈਕਟ (effect) ਤਸਵੀਰ ਉੱਪਰ ਹੀ ਲਾਗੂ ਹੁੰਦਾ ਹੈ।

ਐਡਿਟ ਮੂਵੀ ਸ਼ੈਕਸ਼ਨ ਵਿਚੋ ਵਿਊ ਵੀਡੀਓ ਇਫੈਕਟ (effect) ਚੁਣੋ ਅਤੇ ਚੁਣੇ ਹੋਏ ਇਫੈਕਟ (effect) ਨੂੰ ਤਸਵੀਰ ਦੇ ਖੱਬੇ ਪਾਸੇ ਹੇਠਾਂ *ਸਟਾਰ* ਤੇ ਲੈ ਕੇ ਜਾਉ। **(Choose Edit Menu ->View Video effects & drag the chosen effect to the *star* in the lower left corner of the picture)**

ਆਪਣੇ ਇਫੈਕਟ (effect) ਨੂੰ ਦੇਖਣ ਲਈ ਮੂਵੀ ਪ੍ਰੀ-ਵਿਊ ਨੂੰ ਦੇਖੋ।



Apply Video Effects

ਟਾਈਟਲ ਅਤੇ ਕਰੈਡਿਟ (Titles and credits)

ਤੁਸੀ ਆਪਣੀ ਪਸੰਦ ਦਾ ਕੋਈ ਵੀ ਟੈਕਸਟ ਐਡ ਕਰ ਸਕਦੇ ਹੋ ਜਿਵੇਂ ਕਿ ਮੂਵੀ ਦਾ ਟਾਈਟਲ, ਤਾਰੀਖ ਅਤੇ ਐਕਸ਼ਨ ਦਾ ਨਾਮ ਆਦਿ। ਤੁਸੀ ਆਪਣੇ ਟਾਈਟਲ ਨੂੰ ਮੂਵੀ ਦੀ ਸ਼ੁਰੂਆਤ ਵਿਚ ਜਾਂ ਆਖਿਰ ਵਿਚ ਰੱਖ ਸਕਦੇ ਹੋ। ਤੁਸੀ ਟਾਈਟਲ ਨੂੰ ਦਿਖਾਉਣ ਲਈ ਸਮਾਂ ਨਿਰਧਾਰਿਤ ਵੀ ਕਰ ਸਕਦੇ ਹੋ।

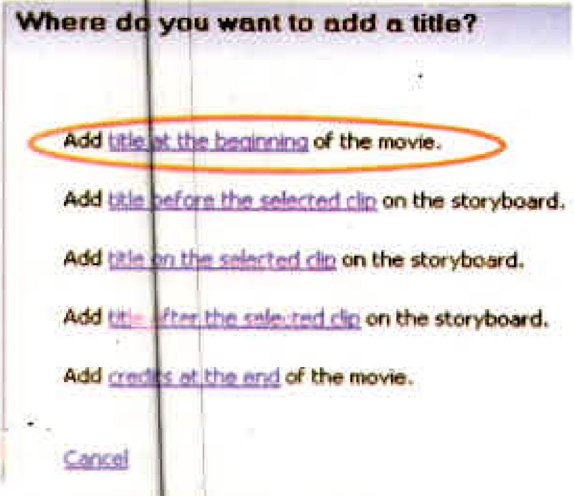
ਕਦਮ 1

ਐਡਿਟ ਮੂਵੀ ਸੈਕਸ਼ਨ ਵਿਚ, ਮੇਕ ਟਾਈਟਲ ਜਾਂ ਕਰੈਡਿਟ (credit) ਨੂੰ ਚੁਣੋ ਟਾਈਟਲ ਜਾਂ ਕਰੈਡਿਟ ਨੂੰ ਰੱਖਣ ਲਈ ਜਗ੍ਹਾ ਨੂੰ ਚੁਣੋ

ਨੋਟ : ਇਸ ਉਦਾਹਰਨ ਵਿਚ ਅਸੀ ਟਾਈਟਲ ਨੂੰ ਮੂਵੀ ਦੇ ਸ਼ੁਰੂਆਤ ਵਿਚ ਰਖਾਂਗੇ

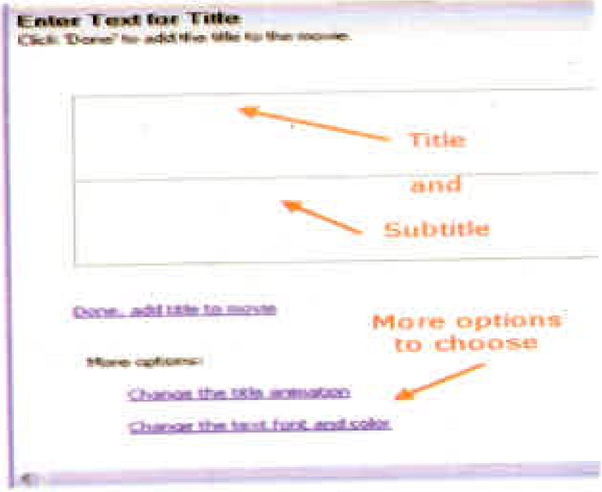
ਮੂਵੀ ਦਾ ਟਾਈਟਲ ਅਤੇ ਸਬ-ਟਾਈਟਲ ਲਿਖੋ

ਤੁਸੀ ਟਾਈਟਲ ਦੇ ਫੌਂਟ ਅਤੇ ਕਲਰ ਨੂੰ ਵੀ ਬਦਲ ਸਕਦੇ ਹੋ ।



ਕਦਮ 2

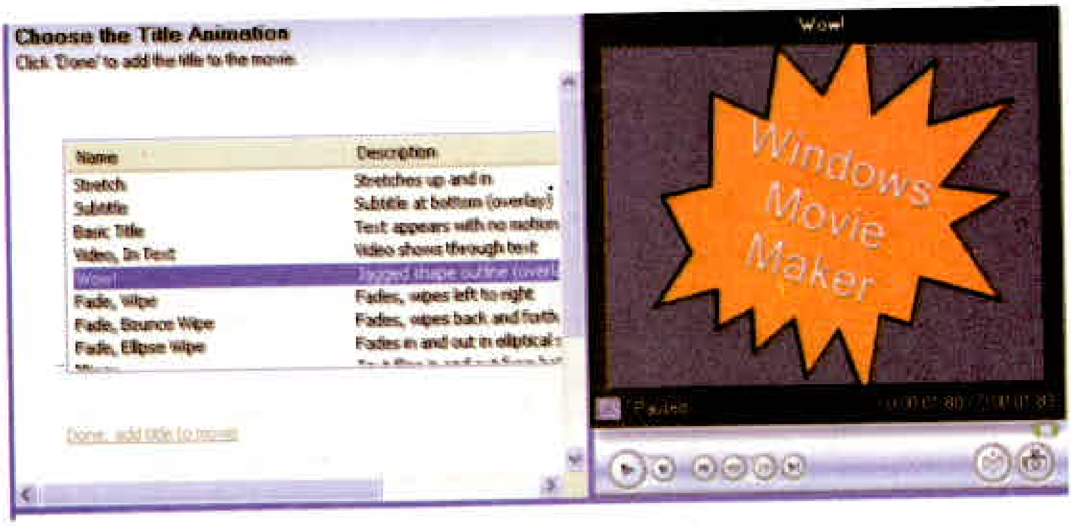
ਟਾਈਟਲ ਵਿੰਡੋਜ਼ ਵਿਚ **GOD IS GREAT** ਲਿਖੋ।



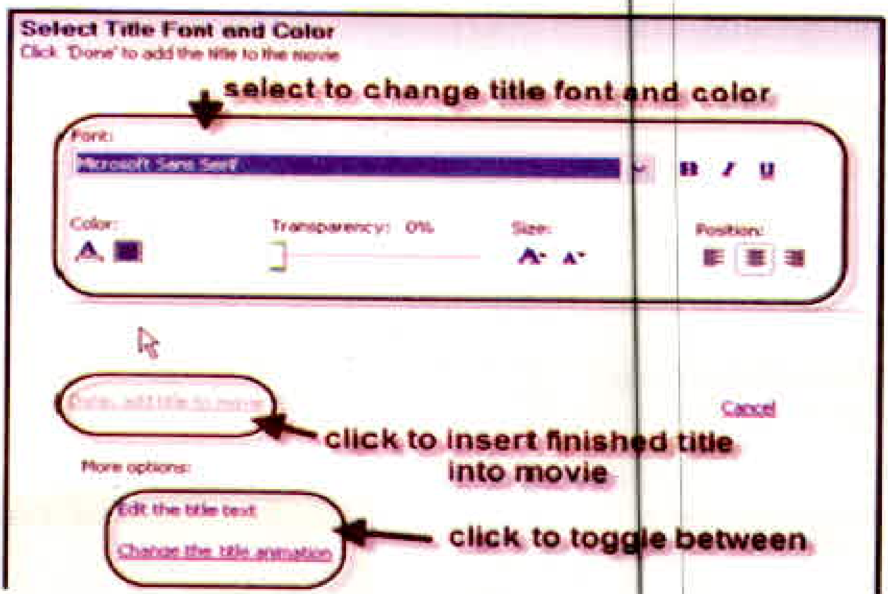
ਕਦਮ 3

change the title animation ਤੇ ਕਲਿੱਕ ਕਰੋ ਤੁਹਾਡਾ ਟਾਈਟਲ ਇਕ ਲਾਈਨ ਜਾਂ ਇਕ ਤੋਂ ਜ਼ਿਆਦਾ ਲਾਈਨਾਂ ਦਾ ਵੀ ਹੋ ਸਕਦਾ ਹੈ

"Choose the animation page" ਉੱਪਰ ਅਲੱਗ-ਅਲੱਗ ਐਨੀਮੇਸ਼ਨ ਨੂੰ ਦੇਖਣ ਲਈ ਪ੍ਰੀ-ਵਿਊ ਮੋਨੀਟਰ ਨੂੰ ਦੇਖੋ। ਜਦੋਂ ਤੁਸੀ ਆਪਣੀ ਐਨੀਮੇਸ਼ਨ ਲੱਭ ਲੈਂਦੇ ਹੋ ਤਾਂ "Change the text font and colour" ਤੇ ਕਲਿੱਕ ਕਰੋ।



ਸਲੈਕਟ ਟਾਈਟਲ ਫੌਂਟ ਅਤੇ ਕਲਰ ਪੇਜ ਉੱਪਰ ਤੁਸੀ ਵਰਤੇ ਜਾਣ ਵਾਲੇ ਟਾਈਟਲ ਲਗਾਉਣਾ ਚਾਹੁੰਦੇ ਹੋ ਤਾਂ ਚਮਕੀਲੀ ਤਸਵੀਰਾਂ (bright scenes) ਲਈ ਫੌਂਟ ਗਾੜ੍ਹੇ ਕਲਰ ਦੇ ਅਤੇ ਧੁੰਦਲੀਆਂ ਤਸਵੀਰਾਂ dark scenes ਲਈ ਹਲਕੇ ਕਲਰ ਦੀ ਵਰਤੋਂ ਕਰੋ। ਹੁਣ ਟਾਈਟਲ ਟੈਕਸਟ ਨੂੰ ਐਡਿਟ ਕਰੋ।



ਪ੍ਰੀ-ਵਿਊ ਮੂਵੀ ਟਾਈਟਲ (Preview Movie Title)

ਆਡੀਓ ਐਡ ਕਰਨਾ (Audio adding)

ਇਕ ਮੂਵੀ ਨੂੰ ਸੰਪੂਰਣ ਕਰਨ ਲਈ ਮਿਊਜ਼ਿਕ ਦੀ ਜਰੂਰਤ ਪੈਂਦੀ ਹੈ

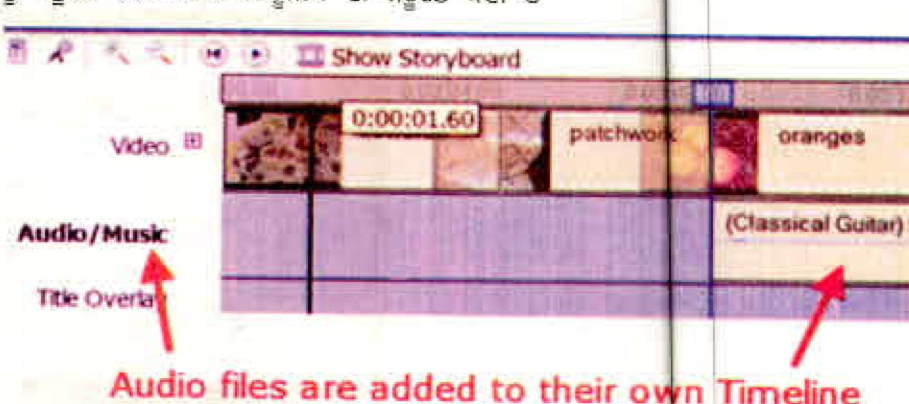


Fig : A

ਆਡੀਓ ਨੂੰ ਪਿਕਚਰ ਨਾਲ ਅਲਾਇੰਨ ਕਰਨਾ (aligning audio with picture)

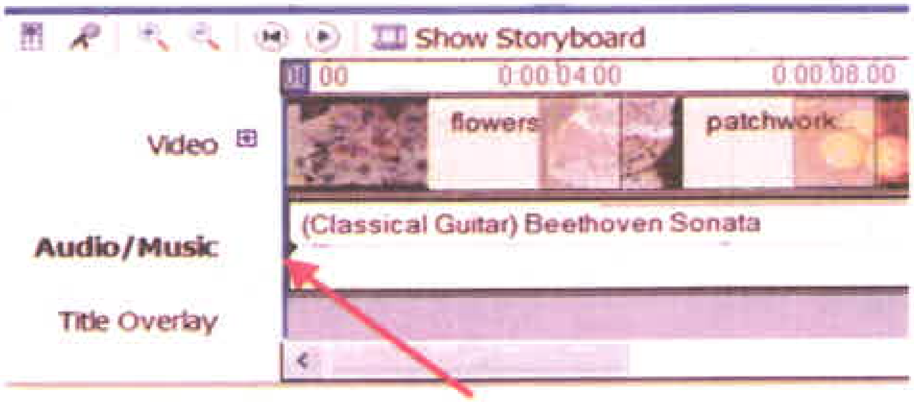
ਆਡੀਓ ਫਾਈਲਜ਼ ਨੂੰ ਖੱਬੇ ਪਾਸੇ ਲੈ ਕੇ ਜਾਉ ਤਾਂ ਜੋ ਉਹ ਪਹਿਲੀ ਪਿਕਚਰ ਦੇ ਨਾਲ ਜੁੜ ਜਾਣ

Drag audio icon to the left to align with start of the first picture.

Fig :B

ਆਡੀਓ ਕਲਿਪ ਦਾ ਟਾਈਮ ਲਾਈਨ ਵਿਊ (Time line view of audio clip)

ਟਾਈਮ ਲਾਈਨ ਤੁਹਾਨੂੰ ਇਹ ਦਰਸਾਉਂਦਾ ਹੈ ਕਿ ਕੋਈ item ਮੂਵੀ ਵਿਚ ਕਿੰਨੀ ਦੇਰ ਤੱਕ ਚਲ ਸਕਦਾ ਹੈ। ਇਸ ਮਿਸਾਲ ਵਿਚ ਅਤੇ ਦਿੱਤੀ ਗਈ ਪਿਕਚਰ ਵਿਚ, ਸਾਡੀ ਮਿਊਜ਼ਿਕ ਫਾਈਲਜ਼ ਲਗਭਗ 4-23 ਮਿੰਟ ਤੱਕ ਚਲੇਗੀ ਜੋ ਕਿ ਜਰੂਰਤ ਨਾਲੋ ਜ਼ਿਆਦਾ ਹੈ

Music ends at approximately 4:23 minutes

ਮਾਊਸ ਨੂੰ ਆਡੀਓ ਕਲਿਪ ਦੇ ਆਖਿਰ ਤੱਕ ਲੈ ਕੇ ਜਾਉ ਜਦੋ ਤੱਕ ਕਿ ਇਹ two headed arrow ਨਿਸ਼ਾਨ ਨਹੀਂ ਬਣ ਜਾਂਦਾ। ਇਸ ਸਿਰੇ ਨੂੰ ਖਿੱਚ ਕੇ ਛੋਟਾ ਕਰੋ।

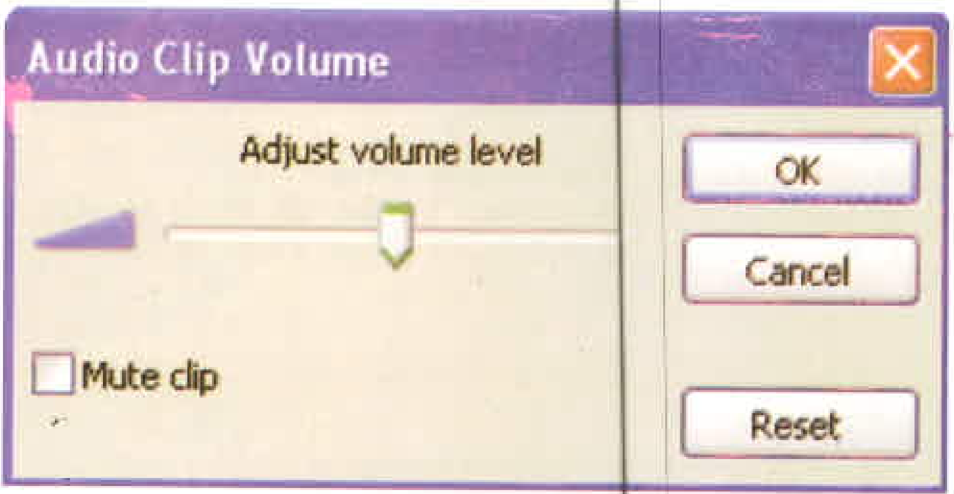
Music and pictures are lined up

ਨੋਟ (Note): ਤੁਸੀ ਆਪਣੀ ਮੂਵੀ ਵਿਚ ਕਿਸੇ ਵੀ ਸਮੇ ਮਿਊਜ਼ਿਕ ਨੂੰ ਚਲਾ ਸਕਦੇ ਹੋ। ਮਿਊਜ਼ਿਕ ਕਲਿਪ ਨੂੰ ਸ਼ੁਰੂਆਤ ਵਿਚ ਨਹੀ ਰਖਣਾ ਚਾਹੀਦਾ।

ਐਡਿਟਿੰਗ ਆਡੀਓ (Editing Audio)

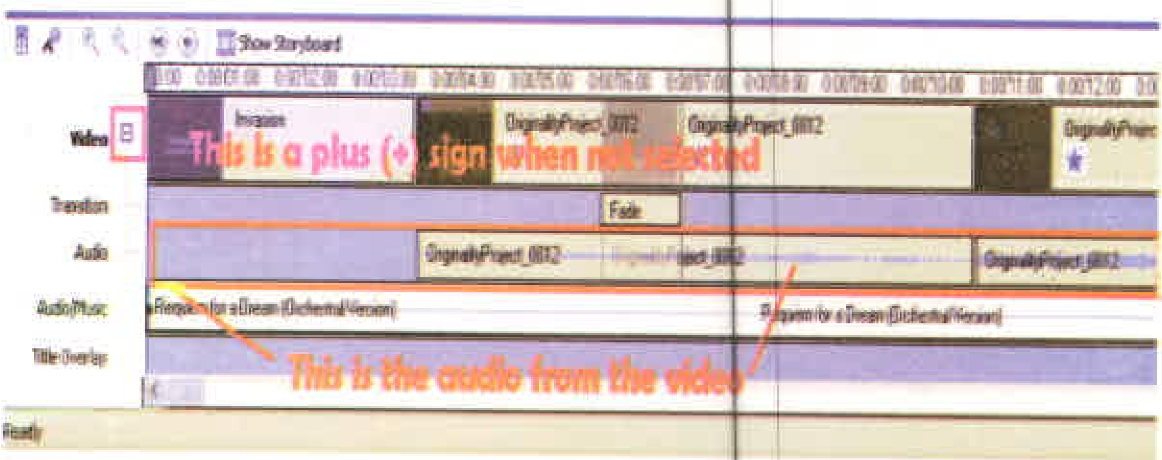
ਆਡੀਓ ਕਲਿਪ ਨੂੰ ਨਿਰਧਾਰਿਤ ਪੁਆਂਇੰਟ ਤੇ ਰੋਕਣ ਲਈ ਤੁਸੀ ਸਪਲਿਟ (split) ਬਟਨ ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਦੇ ਹੋ

- ਵੀਡੀਓ ਦੀ ਤਰ੍ਹਾਂ ਤੁਸੀ ਸੱਜਾ ਕਲਿੱਕ ਕਰ ਕੇ fade in & fade out ਦੀ ਵਰਤੋ ਕਰ ਸਕਦੇ ਹੋ।
- ਇਸ ਤਰ੍ਹਾਂ ਕਰਨ ਨਾਲ ਮਿਊਜ਼ਿਕ ਸ਼ੁਰੂ ਵਿਚ ਹੋਲੀ ਅਤੇ ਫਿਰ ਉੱਚਾ ਹੁੰਦਾ ਜਾਏਗਾ।
- ਆਡੀਓ ਆਵਾਜ਼ ਘੱਟ ਵੱਧ ਕਰਨ ਲਈ, ਸੱਜਾ ਕਲਿੱਕ ਕਰੋ ਅਤੇ volume ਸਿਲੈਕਟ ਕਰੋ।
- ਆਵਾਜ਼ ਨੂੰ ਸਹੀ ਕਰਨ ਲਈ ਵਿੰਡੋਜ਼ ਤੇ ਬਣੇ ਬਾਰ ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਦੇ ਹੋ। ਆਵਾਜ਼ ਨੂੰ mute ਵੀ ਕਰ ਸਕਦੇ ਹੋ।



ਵੀਡੀਓ ਦਾ ਆਡੀਓ ਘੱਟ ਜਾਂ ਵੱਧ ਕਰਨਾ (Adjusting audio of video)

ਕਈ ਵਾਰ ਤੁਸੀ ਆਡੀਓ ਨੂੰ ਘੱਟ ਜਾਂ ਵੱਧ ਕਰਨਾ ਚਾਹੁੰਦੇ ਹੋ ਤਾਂ ਅਜਿਹਾ ਕਰਨ ਲਈ, ਟਾਈਮ ਲਾਈਨ ਉੱਪਰ ਖੱਬੇ ਪਾਸੇ ਵਾਲੇ ਬਟਨ ਤੇ ਕਲਿੱਕ ਕਰੋ। ਜਿਵੇਂ ਉੱਪਰ ਪਿਕਚਰ ਵਿਚ ਦਰਸਾਇਆ ਗਿਆ ਹੈ। "Plus sign" ਨੂੰ ਕਲਿੱਕ ਕਰਨ ਨਾਲ ਦੋ ਨਵੇ ਟਰੈਕ ਹੇਠਾਂ ਵਲ ਵਧਣਗੇ ਅਤੇ ਇਹ minus sign ਵਿਚ ਬਦਲ ਜਾਣਗੇ। Transition track ਤੁਹਾਨੂੰ ਬਣਾਈ ਗਈ ਅਲੱਗ-ਅਲੱਗ transition ਦਿਖਾਏਗਾ।

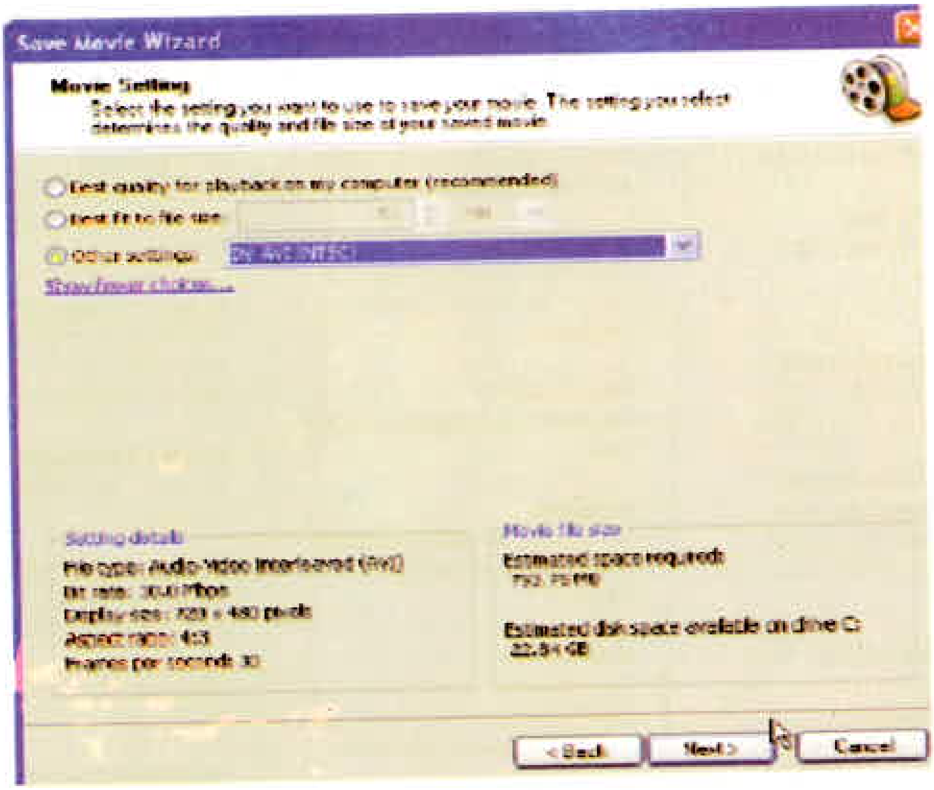


Rendering

ਜਦੋਂ ਤੁਹਾਡੀ ਮੂਵੀ ਪੂਰੀ ਤਰ੍ਹਾਂ ਐਡਿਟ ਹੋ ਜਾਂਦੀ ਹੈ ਅਤੇ ਖਤਮ ਹੋ ਜਾਂਦੀ ਹੈ, ਤਾਂ ਤੁਹਾਨੂੰ ਇਸ ਮੂਵੀ ਨੂੰ "viewable form" ਰਾਹੀਂ render ਕਰਨਾ ਪੈਂਦਾ ਹੈ। ਜਦੋਂ ਮੂਵੀ render ਕਰ ਲਈ ਜਾਂਦੀ ਹੈ ਤਾਂ ਇਸਨੂੰ ਅਸੀਂ ਵੰਡ ਸਕਦੇ ਹਾਂ। ਅਜਿਹਾ ਕਰਨ ਲਈ, ਫਾਈਲ ਮੀਨੂੰ ਤੇ ਜਾ ਕੇ "save movie file" ਤੇ ਕਲਿੱਕ ਕਰੋ ਤੇ ਤੁਸੀਂ ਇਕ ਵਿਜ਼ਾਰਡ (wizard) ਦੇਖੋਗੇ।

ਇਸ ਆਪਸ਼ਨ ਨੂੰ ਸਲੈਕਟ ਕਰਨ ਨਾਲ ਫਾਈਲ ਕੰਪਿਊਟਰ ਵਿਚ ਸੇਵ ਹੋ ਜਾਵੇਗੀ। **Next** ਕਲਿੱਕ ਕਰੋ, ਤੁਸੀ **file name step** ਤੇ ਪਹੁੰਚ ਜਾਵੋਗੇ। ਫਾਈਲ ਨੂੰ ਸੇਵ ਕਰਨ ਲਈ ਫੋਲਡਰ ਚੁਣੋ।

ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਵਿਚ ਇਕ ਮਹੱਤਵਪੂਰਣ ਕਦਮ ਹੈ। "best quality for play". ਇਹ ਆਪਸ਼ਨ ਵੀਡੀਓ ਫਾਈਲ ਨੂੰ high quality ਨਾਲ render ਕਰੇਗੀ।



ਜੇਕਰ ਤੁਹਾਡੀ ਫਾਈਲ 50 Mb ਤੋਂ ਘੱਟ ਹੈ ਤਾਂ ਤੁਸੀਂ ਇਸ ਫਾਈਲ ਨੂੰ ਛੋਟੀ ਫਾਈਲ ਬਣਾ ਕੇ ਸੇਵ ਕਰ ਸਕਦੇ ਹੋ। ਤੁਸੀਂ ਅੱਗੇ ਇਹ ਸਿਖੋਗੇ ਕਿ render ਮੂਵੀ ਦਾ ਸਾਈਜ਼ ਕੀ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ ।

Quick & Easy Guide to Windows Movie Maker

Create a New Project
File → New Project

Open a Project
File → Open Project → File Name → Open

Capture Entire Video from DV Camera
Set Camera to Play → File → Capture Video → DV Camera → Enter a File Name → Choose a Place to Save captured video → Capture Method select Capture the Entire Tape Automatically → Finish

Add a Clip to a Project
Collection → Contents → Select the Clip → Clip -->Add to Storyboard or Add to Timeline

Remove a Clip from a Project
Click the clip in the Storyboard/Timeline → Edit → Delete

Preview a Project
Add clips to storyboard/timeline → Rewind Storyboard → Play Storyboard

Trim a Clip
View → Timeline → Click the clip from the Collections/Contents pane → Clip → Add to Timeline, select the clip → click the playback indicator, drag to the point to trim the clip

Preview a Clip
Click clip from Contents Pane → Play → Play Clip

Split a Video or Audio Clip
In Contents Pane, click the clip → Play → Play Clip → Pause Clip (point you want to split the clip) → Clip → Split

Move a Clip on the Storyboard/Timeline
Select the clip from the storyboard/timeline → Edit → Cut → Click empty shell on the storyboard → Edit → Paste

Clear the Storyboard/Timeline
Edit → Clear Timeline or Clear Storyboard

Create Clips from Existing Video Clip
Select Video clip from Contents Pane → Tools → Create Clips

Switch the View of the Project
View → Storyboard
View → Timeline

Add Video Transitions
Select the clip on the storyboard/timeline → Tools → Video Transitions → select the video transition in the Contents Pane → Clip → Add to Timeline or Add to Storyboard

Remove Video Transition
On the storyboard, select the transition cell → Edit → Delete or On the timeline, click the transition → Edit → Delete

Add a Title or Credit
Tools → Titles and Credits → Add Title, Animations and Change Text Font and Color

Edit an Existing Title
Select the title on the storyboard/timeline → Edit → Edit Title → Type the new text, animation or text font and color

Remove the Title
Select the title on the storyboard/timeline → Edit → Delete

Add Narration
View -> Timeline → Move playback indicator on the timeline where you want to begin the audio narration

Adjust Audio Levels
Tools → Audio Levels → Increase Audio Level (Drag slider bar towards Audio/Music) or Decrease Audio Level (Drag slider bar towards Audio from video)

Add an Audio Effect
Select audio clip from audio or audio music track of the timeline → Clip → Audio → Fade In (to fade in the audio) or Clip → Audio → Fade Out (to fade out the audio)

Remove an Audio Effect
Select audio clip from audio or audio music track of the timeline → Clip → Audio → Clear select effect

Create a Collection
From Collections pane → Click the folder to add new collection → Tools → New Collection Folder and type a name for the collection

Delete a Collection
From Collections pane → Click the collection to be deleted → Edit → Delete

Delete a Clip from a Collection
From Collections pane → Click the collection with clips to be deleted → Switch to Contents pane → Click the clip → Edit → Delete

Copy a Clip in a Collection
From Contents pane → Clip the clip to be copied → Edit → Copy - Switch to Collections pane → Click the collection you want to copy the clips to → Click Contents pane → Edit → Paste

Change the Clip View
View → Thumbnails (view the title and bitmap image) or View → Details (view properties of each clip)

Arrange Clips in Contents Pane
From Collections pane -> Click the folder that contains the clips → View → Arrange Icons By → Choose the property to arrange the clips by

Play Video in Full-Screen Mode
Select video on the storyboard/timeline → View → Full Screen

Zoom In and Out on the Timeline
View → Timeline → Zoom In (more detailed view) or Zoom Out (less detailed view)

Fit the Timeline on the Screen
View → Zoom to Fit

Saving a Movie to a CD-R or CD-RW
File → Save Movie File → Recordable CD →Type Name for CD → Movie Setting(optional) → Save to CD

Record a Movie Back to Tape
Switch DV camera to Playback Mode (Rewind or Fast Forward Tape, if necessary) → File → Save Movie File → Click DV Camera → Follow Wizard → Click Finish when completed

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1. ਹੇਠਾਂ ਦਿਤੇ ਪ੍ਰਸ਼ਨਾਂ ਦੇ ਜਵਾਬ ਦਿਓ ?

1.1 ਸਟੋਰੀ ਬੋਰਡ ਅਤੇ ਟਾਈਮ ਲਾਈਨ ਵਿਚ ਕੀ ਅੰਤਰ ਹੈ ?
1.2 ਵਰਕਸਪੇਸ (workspace) ਕੀ ਹੁੰਦੀ ਹੈ ਅਤੇ ਇਸਦੇ ਕਿਹੜੇ-ਕਿਹੜੇ ਰੂਪ ਹੁੰਦੇ ਹਨ ?
1.3 ਜੇਕਰ ਤੁਸੀ ਟਾਸਕ ਬਟਨ ਉੱਪਰ, ਜੋ ਕਿ ਮੀਨੂੰ ਬਾਰ ਦੇ ਹੇਠਾਂ ਹੈ, ਕਲਿੱਕ ਕਰਦੇ ਹੋ ਤਾਂ ਕੀ ਹੋਵੇਗਾ ?
1.4 ਅਲੱਗ-ਅਲੱਗ ਕਲਿਪ ਨੂੰ ਇਕ ਮੂਵੀ ਵਿਚ ਲਿਆਉਣ ਦੇ ਕਿਹੜੇ-ਕਿਹੜੇ ਪੜਾਵ ਹਨ ?
1.5 ਮੋਨੀਟਰ ਉੱਪਰ ਸਪਲਿਟ ਬਟਨ ਦਾ ਕੀ ਉਪਯੋਗ ਹੈ ?

ਪ੍ਰਸ਼ਨ 2: ਖਾਲੀ ਸਥਾਨ ਭਰੋ

2.1 ਜਦੋਂ ਤੁਸੀ ਵੀਡੀਓ ਕਲਿਪ ਨੂੰ ਸਟੋਰੀ ਬੋਰਡ ਵਿਚ ਕਾਪੀ ਜਾਂ ਮੂਵ ਕਰਦੇ ਹੋ ਤਾਂ ਇਸਦੇ ਨਾਲ __________ ਵੀ ਕਾਪੀ ਹੋ ਜਾਂਦੇ ਹਨ ?
2.2 __________ ਦੇ ਉਪਯੋਗ ਨਾਲ ਸ੍ਰੋਤ ਸਾਮੱਗਰੀ (source material) ਦੀ ਸੂਚਨਾ ਸਮਾਪਤ ਹੋ ਜਾਂਦੀ ਹੈ ?
2.3 ਅਸੀ ਕਲਿਪ ਤੋਂ ਬਾਅਦ __________ ਦੇ ਉਪਯੋਗ ਨਾਲ ਅਗਲੇ ਕਲਿਪ (next clip) ਦੀ ਪਰਿਭਾਸ਼ਾ ਦੇ ਸਕਦੇ ਹਾਂ ?
2.4 ਅਸੀ __________ ਮੂਵੀ ਆਪਸ਼ਨ ਦੇ ਉਪਯੋਗ ਨਾਲ ਮੂਵੀ ਨੂੰ ਸੇਵ ਕਰ ਸਕਦੇ ਹਾਂ ?
2.5 ਕਲਿਪ ਦੀ ਸੋਧ ਲਈ __________ ਇਕ ਜ਼ਰੂਰੀ ਪ੍ਰਕਿਰਿਆ ਹੈ ?

ਪ੍ਰਸ਼ਨ 3: ਦਸੋ ਸਹੀ ਜਾਂ ਗਲਤ

3.1 ਕੀ ਅਸੀ ਐਡਵਾਂਸ ਆਡੀਓ ਇਫੈਕਟ (effect) ਨੂੰ ਟਾਈਮ-ਲਾਈਨ ਵਿਊ ਵਿਚ ਭਰ ਸਕਦੇ ਹਾਂ ?
3.2 ਸੇਵ ਮੂਵੀ ਨੂੰ ਵੈੱਬ ਪੇਜ ਜਾਂ ਈ-ਮੇਲ ਦੁਆਰਾ ਦੇਖਿਆ ਜਾਂ ਭੇਜਿਆ ਜਾ ਸਕਦਾ ਹੈ, ਪਰ ਇਸਨੂੰ ਸੋਧਿਆ ਨਹੀ ਜਾ ਸਕਦਾ ?
3.3 ਕੀ ਟਾਈਮ-ਲਾਈਨ ਵਿਚ ਜ਼ੂਮਿੰਗ (zooming) ਕਰਨ ਨਾਲ ਟਾਈਮ ਛੋਟੇ-ਛੋਟੇ ਭਾਗਾਂ ਵਿਚ ਦਿਖਾਈ ਦਿੰਦਾ ਹੈ, ਤਾਂ ਜੋ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਜ਼ਿਆਦਾ ਅਸਾਨੀ ਨਾਲ ਪੜ੍ਹਿਆ ਜਾ ਸਕੇ ?
3.4 ਕੀ ਟਰਾਂਜ਼ਿਸ਼ਨ (transition) ਨੂੰ ਕੋਲੈਕਸ਼ਨ-ਪੇਨ (collection pane) ਦੇ ਵੀਡੀਓ/ਆਡੀਓ ਟਰਾਂਜ਼ਿਸ਼ਨ ਫੋਲਡਰ ਵਿਚ ਰੱਖ ਸਕਦੇ ਹਾਂ ?
3.5 ਕੇਵਲ ਵੀਡੀਓ ਇਫੈਕਟਸ ਜੋ ਕਿ ਵੀਡੀਓ ਇਫੈਕਟਸ ਫੋਲਡਰ ਵਿਚ ਹੋਣ, ਨੂੰ ਜੋੜ ਸਕਦੇ ਹਾਂ ?

Answer to objective questions

Q	.1	.2	.3	.4	.5
Q2	Video Effects	Trimming	Title	Finish	Splitting
Q3	T	F	T	F	T

ਅਧਿਆਇ

ਨੈੱਟਵਰਕਿੰਗ ਸੰਕਲਪ (NETWORKING CONCEPTS)

6

6.0.0 ਭੂਮਿਕਾ (Introduction)

ਅੱਜ ਦੇ ਸ਼ਕਤੀਸ਼ਾਲੀ ਯੁੱਗ ਨੂੰ ਇਕ ਸਥਾਨ ਤੋਂ ਦੂਜੇ ਲਈ ਅਕਸਰ ਡਾਟਾ ਭੇਜਣ ਲਈ ਤੇਜ਼ ਸੰਚਾਰ (communication) ਚੈਨਲਾਂ ਦੀ ਲੋੜ ਪੈ ਰਹੀ ਹੈ। ਇਸ ਲਈ ਦੂਰੀ ਤੇ ਡਾਟਾ ਸੰਚਾਰ (communication) ਕਰਨਾ ਬਹੁਤ ਜਰੂਰੀ ਹੋ ਗਿਆ ਹੈ। ਡਾਟੇ ਨੂੰ ਤੇਜ਼ੀ ਨਾਲ ਇਕ ਸਥਾਨ ਤੋਂ ਦੂਜੇ ਤਕ ਭੇਜਣ ਲਈ ਨੈੱਟਵਰਕਿੰਗ ਦਾ ਸੰਕਲਪ ਸ਼ੁਰੂ ਕੀਤਾ ਗਿਆ ਹੈ। ਨੈੱਟਵਰਕਿੰਗ ਵਿਚ ਵਿਸ਼ਵ ਦੇ ਵੱਖ-ਵੱਖ ਭਾਗਾਂ ਤੋਂ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਇਕ ਦੂਜੇ ਨਾਲ ਜੋੜਿਆ ਗਿਆ ਹੈ ਤਾਂ ਜੋ ਉਹ ਇਕ ਦੂਜੇ ਨਾਲ ਡਾਟਾ ਸ਼ੇਅਰ (share) ਕਰ ਸਕਣ।

6.1.0 ਨੈੱਟਵਰਕ ਕੀ ਹੁੰਦਾ ਹੈ (What is Network)?

ਨੈੱਟਵਰਕ ਖੁਦ ਮੁਖਤਾਰ ਕੰਪਿਊਟਰਾਂ ਦਾ ਅੰਤਰ ਸੰਬੰਧਿਤ ਇਕਠ ਹੁੰਦਾ ਹੈ। ਦੋ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਅੰਤਰਜੁੜੇ ਕਿਹਾ ਜਾ ਸਕਦਾ ਹੈ ਜੇਕਰ ਉਹ ਸੂਚਨਾ (information) ਦਾ ਆਦਾਨ ਪ੍ਰਦਾਨ ਕਰਨ ਦੇ ਕਾਬਿਲ ਹੋਣ। ਸ਼ਬਦ ਖੁਦਮੁਖਤਾਰ ਦਾ ਭਾਵ ਹੈ ਕਿ ਨੈੱਟਵਰਕ ਤੇ ਕੰਪਿਊਟਰ ਦੂਜੇ ਨੂੰ ਸ਼ੁਰੂ, ਬੰਦ ਅਤੇ ਨਿਅੰਤਰਨ ਕਰ ਸਕਦੇ ਹਨ। ਇਸ ਲਈ ਨੈੱਟਵਰਕ ਰਾਹੀਂ ਕੰਪਿਊਟਰ ਯੂਜ਼ਰ (user) ਇਕ ਸਥਾਨ ਤੇ ਉਪਲਬਧ ਸੂਚਨਾ (information), ਸਾਜ਼ੋ ਸਮਾਨ, ਪ੍ਰੋਗਰਾਮ ਅਤੇ ਸੰਦੇਸ਼ ਵਰਤ ਸਕਦੇ ਹਨ।

6.1.1 ਨੈੱਟਵਰਕਿੰਗ ਲਈ ਲੋੜ (Need for Networking): ਨੈੱਟਵਰਕਿੰਗ ਨਿਮਨਲਿਖਤ ਕਾਰਨਾਂ ਕਰ ਕੇ ਜਰੂਰੀ ਹੁੰਦੀ ਹੈ :-

(ੳ) **ਕੰਪਿਊਟਰ ਫਾਈਲਾਂ ਦੀ ਸ਼ੇਅਰਿੰਗ ਲਈ :** ਨੈੱਟਵਰਕ ਯੂਜ਼ਰਸ (users) ਨੂੰ ਆਪਣੀਆਂ ਫਾਈਲਾਂ ਜਾਂ ਸੂਚਨਾਵਾਂ ਸ਼ੇਅਰ (share) ਕਰਨ ਲਈ ਇਕ ਮਹੱਤਵਪੂਰਨ ਸਾਧਨ ਪ੍ਰਦਾਨ ਕਰਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਇਕ ਸੰਗਠਨ ਵਿਚ ਇਕ ਫਾਈਲ ਨੂੰ ਵੱਖ-ਵੱਖ ਵਿਭਾਗਾਂ ਦੇ ਨਾਲ ਵੰਡਿਆ ਜਾਣਾ ਹੈ, ਅਸੀਂ ਇਸ ਫਾਈਲ ਨੂੰ ਨੈੱਟਵਰਕ ਪ੍ਰਣਾਲੀ ਤੇ ਰਖਾਂਗੇ। ਹੁਣ ਸਮੂਹ ਵਿਭਾਗ ਇਸ ਫਾਈਲ ਦੀ ਵਰਤੋਂ ਕਰ ਸਕਣਗੇ ।

(ਅ) **ਕੰਪਿਊਟਰ ਸੀਮਾ ਦੀ ਵੰਡ :-** ਕੁਝ ਕੰਪਿਊਟਰ ਡਿਵਾਇਸਿਸ (devices) ਬਹੁਤ ਹੀ ਮਹਿੰਗੇ ਹੁੰਦੇ ਹਨ। ਮਿਸਾਲ ਲਈ ਕਲਰ ਲੇਜ਼ਰ ਪ੍ਰਿੰਟਰ। ਨੈੱਟਵਰਕ ਰਾਹੀਂ ਅਸੀਂ ਇਨ੍ਹਾਂ ਯੰਤਰਾਂ ਨੂੰ ਵੰਡ ਸਕਦੇ ਹਾਂ।

(ੲ) **ਵੱਖ ਵੱਖ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਇਕ ਦੂਜੇ ਨਾਲ ਸੰਚਾਰ (communication) ਕਰਨ ਦੇ ਕਾਬਿਲ ਕਰਦੇ ਹਨ :-** ਨੈੱਟਵਰਕ ਰਾਹੀਂ ਵੱਖ ਵੱਖ ਕੰਪਿਊਟਰ ਇਕ ਦੂਜੇ ਨਾਲ ਸੰਚਾਰ (communication) ਕਰਦੇ ਹਨ। ਮਿਸਾਲ ਲਈ, ਉਪਰੇਟਿੰਗ ਸਿਸਟਮ ਵਿੰਡੋਜ਼ ਵਾਲਾ ਇਕ ਕੰਪਿਊਟਰ ਦੂਜੀ ਉਪਰੇਟਿੰਗ ਸਿਸਟਮ ਲਿਨੀਕਸ (Linux) ਦੇ ਨਾਲ ਸੰਚਾਰ (communication) ਕਰ ਸਕਦਾ ਹੈ।

(ਸ) **ਸੰਚਾਰ (communication) ਗਤੀ ਅਤੇ ਸ਼ੁਧਤਾ ਵਿਚ ਸੁਧਾਰ :-** ਇਹ ਸੰਭਵ ਹੈ ਕਿ ਡਾਕ ਨੂੰ ਇਕ ਥਾਂ ਤੋਂ ਦੂਜੀ ਥਾਂ ਤੇ ਪਹੁੰਚਣ ਲਈ ਅਨੇਕਾਂ ਦਿਨ ਲਗ ਜਾਂਦੇ ਹਨ। ਅਸੀਂ ਨੈੱਟਵਰਕਿੰਗ ਰਾਹੀਂ ਤੇਜ਼ੀ ਨਾਲ ਸੰਦੇਸ਼ ਭੇਜ ਸਕਦੇ ਹਾਂ।

(ਹ) **ਡਾਟਾ ਬਦਲੀ ਦੀ ਲਾਗਤ ਵਿਚ ਕਟੌਤੀ :-** ਡਾਟਾ ਬਦਲਣ ਲਈ ਕੰਪਿਊਟਰ ਬਹੁਤ ਸਸਤਾ ਢੰਗ ਹੈ। ਇਸ ਲਈ ਕੰਪਿਊਟਰ ਦੀ ਮਦਦ ਨਾਲ ਅਸੀਂ ਡਾਟਾ ਬਦਲੀ ਦੀ ਲਾਗਤ ਘਟਾ ਸਕਦੇ ਹਾਂ।

6.1.2 ਨੈੱਟਵਰਕਿੰਗ ਟਰਮੀਨੋਲੋਜੀ (Networking Terminology): ਨੈੱਟਵਰਕਿੰਗ ਦੇ ਵੇਰਵਿਆਂ ਵਿਚ ਜਾਣ ਤੋਂ ਪਹਿਲਾਂ ਆਓ ਨੈੱਟਵਰਕ ਦੀ ਬੁਨਿਆਦੀ ਟਰਮੀਨੋਲੋਜੀ (terminology) ਤੇ ਗੌਰ ਕਰੀਏ।

(ੳ) **ਨੋਡਸ (ਵਰਕ ਸਟੇਸ਼ਨ) Nodes (Workstations):** ਵੱਖ ਵੱਖ ਟਰਮੀਨਲ ਨੈੱਟਵਰਕ ਦੇ ਨਾਲ ਜੁੜੇ ਹੋਏ ਹੁੰਦੇ ਹਨ ਅਤੇ ਨੈੱਟਵਰਕ ਦਾ ਸਰੋਤ ਵੰਡਦੇ ਹਨ ਜਿਨ੍ਹਾਂ ਨੂੰ ਨੋਡਸ (nodes) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਜਦੋਂ ਅਸੀਂ ਇਕ ਕੰਪਿਊਟਰ ਨੂੰ ਨੈੱਟਵਰਕ ਦੇ ਨਾਲ ਜੋੜਦੇ ਹਾਂ ਤਾਂ ਇਹ ਨੈੱਟਵਰਕ ਦਾ ਵਰਕ-ਸਟੇਸ਼ਨ ਬਣ ਜਾਂਦਾ ਹੈ।

(ਅ) **ਸਰਵਰ (Server):** ਅਸੀਂ ਇਕ ਵਿਸ਼ੇਸ਼ ਨੋਡ (node) ਨੂੰ ਅਹੁਦਾ ਦਿੰਦੇ ਹਾਂ ਜੋ ਪ੍ਰਸਿੱਧ ਹੋ ਜਾਂਦਾ ਹੈ ਅਤੇ ਇਸਦਾ ਨਿਸ਼ਚਿਤ ਪਤਾ ਹੁੰਦਾ ਹੈ, ਜੋ ਸਮੂਚੇ ਰੂਪ ਵਿਚ ਨੈੱਟਵਰਕ ਨੂੰ ਸੇਵਾ ਪ੍ਰਦਾਨ ਕਰਦਾ ਹੈ। ਸੇਵਾ ਪ੍ਰਦਾਨ ਕਰਨ ਵਾਲੀ ਨੋਡ ਨੂੰ ਸਰਵਰ (server) ਕਿਹਾ ਜਾਂਦਾ ਹੈ।

(ੲ) **ਨੈੱਟਵਰਕ ਇੰਟਰਫੇਸ ਯੂਨਿਟ (Network Interface Unit):** ਦੁਭਾਸ਼ੀਆ (interpreter), ਜਿਹੜਾ ਸਰਵਰ (server) ਅਤੇ ਵੱਖ ਵੱਖ ਨੋਡਸ (nodes) ਦੇ ਦਰਮਿਆਨ ਸੰਚਾਰ (communication) ਪ੍ਰਦਾਨ ਕਰਨ ਵਿਚ

ਮਦਦ ਕਰਦਾ ਹੈ, ਨੂੰ ਨੈੱਟਵਰਕ ਇੰਟਰਫੇਸ ਯੂਨਿਟ (network interface unit) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਨੈੱਟਵਰਕ ਇੰਟਰਫੇਸ ਯੂਨਿਟ ਇਕ ਡਿਵਾਇਸ (device) ਹੈ ਜਿਹੜਾ ਸਰਵਰ (server) ਅਤੇ ਸਮੂਹ ਵਰਕਸਟੇਸ਼ਨਸ (work stations) ਦੇ ਦਰਮਿਆਨ ਸੰਬੰਧ ਜੋੜਨ ਲਈ ਜੁੜਿਆ ਹੁੰਦਾ ਹੈ। ਹਰ ਇਕ ਨੈੱਟਵਰਕ ਇੰਟਰਫੇਸ ਯੂਨਿਟ ਦਾ ਇਕ ਵਿਲੱਖਣ ਨੋਡ ਐਡਰੈਸ (unique node address) ਹੁੰਦਾ ਹੈ।

6.2.0 ਨੈਟਵਰਕ ਦੀਆਂ ਕਿਸਮਾਂ (Types of networking)

ਇਕ ਨੈਟਵਰਕ ਦਾ ਭਾਵ ਹੈ ਕੰਪਿਊਟਰਾਂ ਦਾ ਇਕ ਛੋਟਾ ਜਿਹਾ ਸਮੂਹ ਜੋ ਵੱਖ-ਵੱਖ ਕਿਸਮਾਂ (ਪੀ.ਸੀ., ਮਿੰਨੀ, ਮੇਨ-ਫਰੇਮ ਆਦਿ) ਦੇ ਕੁਝ ਸੈਂਕੜੇ ਕੰਪਿਊਟਰਾਂ ਦੀ ਇਕ ਕੜੀ ਨਾਲ ਜੁੜ ਕੇ ਸਾਰੀ ਦੁਨੀਆਂ ਵਿਚ ਫੈਲਿਆ ਹੁੰਦਾ ਹੈ। ਇਸ ਪ੍ਰਕਾਰ ਨੈਟਵਰਕ ਅਕਾਰ, ਪੇਚੀਦਗੀ ਅਤੇ ਭੂਗੋਲਿਕ (geographical) ਤੌਰ ਤੇ ਫੈਲਿਆ ਹੁੰਦਾ ਹੈ। ਬਹੁਤ ਸਾਰੇ ਕੰਪਿਊਟਰ ਭੂਗੋਲਿਕ (geographical) ਵਿਸਥਾਰ ਦੇ ਅਧਾਰ ਤੇ ਹੀ ਸ਼੍ਰੇਣੀਬੱਧ ਕੀਤੇ ਗਏ ਹੁੰਦੇ ਹਨ ਅਤੇ ਇਸ ਆਧਾਰ ਤੇ, ਤਿੰਨ ਕਿਸਮ ਦੇ ਨੈਟਵਰਕ ਹੋ ਸਕਦੇ ਹਨ:

- **ਲੋਕਲ ਏਰੀਆ ਨੈਟਵਰਕ (ਲੈਨ) [Local area Networks (LANs)]**
- **ਮੈਟਰੋਪੋਲੀਟਨ ਏਰੀਆ ਨੈਟਵਰਕ (ਮੈਨ) (MANs) [Metropolitan Area Networks (MANs)]**
- **ਵਾਈਡ ਏਰੀਆ ਨੈਟਵਰਕਸ (ਵੈਨ) (WANs) [Wide Area Networks (WANs)]**

ਲੋਕਲ ਏਰੀਆ ਨੈਟਵਰਕ (ਲੈਨ): ਲੈਨ (LAN) ਪ੍ਰਾਈਵੇਟ ਨੈਟਵਰਕ ਹੁੰਦੇ ਹਨ ਜਿੰਹੜੇ ਅਕਾਰ ਵਿਚ ਕੁੱਝ ਕਿਲੋਮੀਟਰ ਤੱਕ ਇੱਕ ਬਿਲਡਿੰਗ/ਕੈਂਪਸ ਵਿਚ ਹੁੰਦੇ ਹਨ। ਇਹ ਫੈਕਟਰੀਆਂ ਅਤੇ ਦਫਤਰਾਂ ਵਿਚ ਪੀ ਸੀ ਅਤੇ ਵਰਕਸਟੇਸ਼ਨ ਨੂੰ ਜੋੜ ਕੇ ਸੂਚਨਾ ਦਾ ਅਦਾਨ ਪ੍ਰਦਾਨ ਕਰਨ ਅਤੇ ਸਰੋਤਾਂ (ਮਿਸਾਲ ਲਈ ਪ੍ਰਿੰਟਰ) ਦੇ ਵੰਡ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ

ਮੈਟਰੋਪੋਲੀਟਨ ਏਰੀਆ ਨੈਟਵਰਕ (ਮੈਨ): ਮੈਨ (MAN) ਉਹ ਨੈਟਵਰਕ ਹਨ ਜਿਹੜੇ ਸਾਰੇ ਸ਼ਹਿਰ ਵਿਚ ਫੈਲੇ ਹੋਏ ਹੁੰਦੇ ਹਨ। ਮਿਸਾਲ ਲਈ ਕੇਬਲ ਟੀ ਵੀ ਨੈਟਵਰਕ ਜਿਹੜੇ ਸਾਰੇ ਸ਼ਹਿਰ ਵਿਚ ਫੈਲੇ ਹੋਏ ਹਨ। ਇਸਦਾ ਮਨੋਰਥ ਆਪਣੇ ਉਪਭੋਗਤਾਵਾਂ (users) ਵਿਚ ਹਾਰਡਵੇਅਰ ਅਤੇ ਸਾਫ਼ਟਵੇਅਰ ਵਸੀਲਿਆਂ ਦੀ ਵਰਤੋਂ ਕਰਨਾ ਹੁੰਦਾ ਹੈ।

ਵਾਈਡ ਏਰੀਆ ਨੈਟਵਰਕ (ਵੈਨ): ਸਮੁੱਚੇ ਦੇਸ਼ਾਂ ਵਿਚ ਫੈਲੇ ਨੈਟਵਰਕ ਨੂੰ ਵੈਨ (WAN) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਇਹ ਕੰਪਿਊਟਰਾਂ ਦਾ ਇਕ ਸਮੂਹ ਹੁੰਦਾ ਹੈ ਜਿਹੜਾ ਵੱਡੇ ਫਾਸਲੇ ਨਾਲ ਅਲੱਗ ਕੀਤਾ ਗਿਆ ਹੁੰਦਾ ਹੈ ਪਰ ਇਕਠਿਆਂ ਜੁੜਿਆ ਹੁੰਦਾ ਹੈ। ਇਹ ਇਥੋਂ ਤੱਕ ਕਿ ਲੈਨ (LAN) ਦਾ ਇਕ ਸਮੂਹ ਹੁੰਦਾ ਹੈ ਜੋ ਅਨੇਕਾਂ ਸਥਿਤੀਆਂ ਵਿਚ ਫੈਲੇ ਹੁੰਦੇ ਹਨ ਅਤੇ ਵੱਡਾ ਲੈਨ (LAN) ਦੇਖਣ ਲਈ ਇਕ ਦੂਜੇ ਨਾਲ ਜੁੜੇ ਹੁੰਦੇ ਹਨ। ਵੈਨ (WAN) ਲਿੰਕ ਕੰਪਿਊਟਰ ਘੱਟ ਕੀਮਤਾਂ ਅਤੇ ਉੱਚ ਗਤੀਆਂ ਤੇ ਸੂਚਨਾ ਦਾ ਕੁਸ਼ਲ ਆਦਾਨ ਪ੍ਰਦਾਨ ਕਰਨ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ।

ਲੈਨ , ਮੈਨ ਅਤੇ ਵੈਨ ਦੀ ਤੁਲਨਾ (Comparison of LAN, MAN & WAN)

	ਲੈਨ (LAN)	ਮੈਨ (MAN)	ਵੈਨ (WAN)
ਖੇਤਰ Area	ਇਹ ਆਰਿਜ ਖੇਤਰ ਦੇ ਅੰਦਰ ਹੁੰਦਾ ਹੈ	ਇਹ ਸ਼ਹਿਰ ਦੇ ਅੰਦਰ ਫੈਲਿਆ ਹੁੰਦਾ ਹੈ	ਇਹ ਸੰਸਾਰ ਵਿਚ ਫੈਲਿਆ ਹੁੰਦਾ ਹੈ
ਨੈਟਵਰਕ Network	ਇਹ ਨਿਜੀ-ਨੈਟਵਰਕ ਹੁੰਦਾ ਹੈ	ਆਪਣਾ ਨਿਜੀ ਜਾਂ ਕਿਰਾਏ ਦਾ ਨੈਟਵਰਕ ਹੁੰਦਾ ਹੈ	ਕਿਰਾਏ ਦਾ ਜਾਂ ਸੇਵਾਵਾਂ ਦੇਣ ਵਾਲੇ ਦਾ ਆਪਣਾ ਨਿਜੀ-ਨੈਟਵਰਕ ਹੁੰਦਾ ਹੈ
ਵਾਸਤਵਿਕ ਕੁਨੈਕਸ਼ਨ Physical connection	ਇਸ ਦੇ ਵਿਚ ਡਾਟਾ ਭੇਜਣ ਵਾਲੇ (sender) ਅਤੇ ਸਵੀਕਾਰ-ਕਰਤਾ (receiver) ਵਿਚਕਾਰ ਵਾਸਤਵਿਕ ਕੁਨੈਕਸ਼ਨ (connection) ਹੁੰਦਾ ਹੈ।	ਇਸ ਵਿਚ ਡਾਟਾ ਭੇਜਣ ਵਾਲੇ (sender) ਅਤੇ ਸਵੀਕਾਰ-ਕਰਤਾ (receiver) ਵਿਚਕਾਰ ਵਾਸਤਵਿਕ ਕੁਨੈਕਸ਼ਨ (connection) ਹੁੰਦਾ ਹੈ	ਇਸ ਵਿਚ ਡਾਟਾ ਭੇਜਣ ਵਾਲੇ (sender) ਅਤੇ ਸਵੀਕਾਰ-ਕਰਤਾ (receiver) ਵਿਚਕਾਰ ਵਾਸਤਵਿਕ ਕੁਨੈਕਸ਼ਨ (connection) ਹੋ ਵੀ ਸਕਦਾ ਹੈ ਤੇ ਨਹੀ ਵੀ।
ਲਾਗਤ Cost	ਲਾਗੂ-ਕਰਨ (execution) ਦੀ ਲਾਗਤ ਘੱਟ ਹੈ।	ਲਾਗੂ-ਕਰਨ (execution) ਦੀ ਲਾਗਤ ਘੱਟ ਹੈ ਪਰ ਲੈਨ ਨਾਲੋਂ ਵਧ ਹੁੰਦੀ ਹੈ	ਲਾਗੂ-ਕਰਨ (execution) ਦੀ ਲਾਗਤ ਬਹੁਤ ਜਿਆਦਾ ਹੁੰਦੀ ਹੈ
ਬੈਂਡਵਿਡਥ Bandwidth	10 Mbps ਤੋਂ ਤਕਰੀਬਨ 10 Gbps ਦੇ ਵਿਚ ਹੁੰਦੀ ਹੈ।	10 Mbps ਤੋਂ ਤਕਰੀਬਨ 1 Gbps ਦੇ ਵਿਚ ਹੁੰਦੀ ਹੈ।	15.4 Kbps ਤੋਂ ਤਕਰੀਬਨ 2.4 Gbps ਦੇ ਵਿਚ ਹੁੰਦੀ ਹੈ।
ਯੰਤਰ Devices	ਆਮ ਤੌਰ ਤੇ ਰਿਪੀਟਰਸ (Repeaters), ਹੱਬਸ (hubs), ਰੂਟਰਸ (Routers) ਇਸਤੇਮਾਲ ਹੁੰਦੇ ਹਨ।	ਆਮ ਤੌਰ ਤੇ ਰੂਟਰਸ (Routers), ਬਰਿਜਿਸ (bridges) ਇਸਤੇਮਾਲ ਹੁੰਦੇ ਹਨ।	ਆਮ ਤੌਰ ਤੇ ਵੈਨ ਸਵਿਚਸ (WAN switches), ਮਾਡਮਸ (modems) ਇਸਤੇਮਾਲ ਹੁੰਦੇ ਹਨ।
ਗਲਤੀ-ਦਰ Error rate	ਬਹੁਤ ਘੱਟ ਹੁੰਦੀ ਹੈ	ਲੈਨ ਤੋਂ ਥੋੜੀ ਜਿਆਦਾ ਹੁੰਦੀ ਹੈ	ਲੈਨ ਅਤੇ ਮੈਨ ਤੋਂ ਜਿਆਦਾ ਹੁੰਦੀ ਹੈ

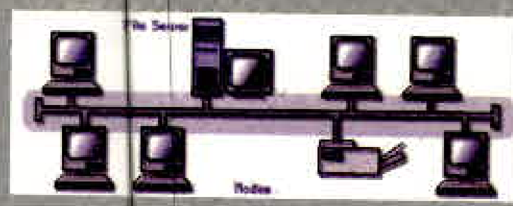
6.3.0 ਨੈਟਵਰਕ ਟੋਪੋਲੋਜੀ (Network Topologies)

6.3.1 ਟੋਪੋਲੋਜੀ ਕੀ ਹੁੰਦਾ ਹੈ (What is a Topology)? ਇਕ ਨੈੱਟਵਰਕ ਦੀ ਭੌਤਿਕ (physical) ਟੋਪੋਲੋਜੀ (topology) ਕੇਬਲਾਂ, ਕੰਪਿਊਟਰਾਂ ਅਤੇ ਅਜਿਹੇ ਹੋਰ ਪੈਰਿਫਰਲ (peripheral) ਦੇ ਸਮੂਹ ਦਾ ਹਵਾਲਾ ਦਿੰਦਾ ਹੈ। ਭੌਤਿਕ ਟੋਪੋਲੋਜੀ ਨੂੰ ਲੋਜੀਕਲ (Logical) ਟੋਪੋਲੋਜੀ ਦੇ ਨਾਲ ਉਲਝਾਉਣਾ ਨਹੀਂ ਚਾਹੀਦਾ ਜਿਹੜੀ ਵਰਕ-ਸਟੇਸ਼ਨਾਂ (workstations) ਦੇ ਦਰਮਿਆਨ ਸੂਚਨਾ (information) ਭੇਜਣ ਵਾਲੀ ਵਿਧੀ ਹੁੰਦੀ ਹੈ।

6.3.2 ਭੌਤਿਕ ਟੋਪੋਲੋਜੀ ਦੀਆਂ ਮੁੱਖ ਕਿਸਮਾਂ (Main Types of Physical Topologies)
ਨਿਮਨਲਿਖਤ ਭਾਗਾਂ ਰਾਹੀ ਨੈੱਟਵਰਕ ਅਤੇ ਹੋਰ ਸੰਬੰਧਤ ਵਿਸ਼ਿਆਂ ਵਿਚ ਵਰਤੇ ਜਾਣ ਵਾਲੇ ਭੌਤਿਕ ਟੋਪੋਲੋਜੀ (physical topology) ਬਾਰੇ ਵਿਚਾਰ ਵਟਾਂਦਰਾ ਕੀਤਾ ਗਿਆ ਹੈ।

- ਬਸ ਟੋਪੋਲੋਜੀ (Bus Topology)
- ਸਟਾਰ ਟੋਪੋਲੋਜੀ (Star Topology)
- ਰਿੰਗ ਟੋਪੋਲੋਜੀ (Ring Topology)
- ਟ੍ਰੀ ਟੋਪੋਲੋਜੀ (Tree Topology)

6.3.3 ਲੀਨੀਅਰ ਬੱਸ (Linear Bus): ਇਕ ਲੀਨੀਅਰ ਬਸ (Linear Bus) ਟੋਪੋਲੋਜੀ (topology) ਵਿਚ ਇਕ ਸਿਰੇ ਤੋਂ ਦੂਜੇ ਤਕ ਟਰਮੀਨੇਟਰ (terminator) ਦੇ ਨਾਲ ਮੇਨ ਕੇਬਲ ਜੁੜੀ ਹੁੰਦੀ ਹੈ। ਸਮੂਹ ਨੋਡਸ (ਫਾਈਲ ਸਰਵਰ, ਵਰਕ-ਸਟੇਸ਼ਨ ਅਤੇ ਸੀਮਾਵਾਂ) ਲੀਨੀਅਰ (Linear) ਕੇਬਲ ਨਾਲ ਜੁੜੇ ਹੁੰਦੇ ਹਨ। ਈਥਰਨੈਟ (ethernet) ਅਤੇ ਲੋਕਲ ਟਾਕ (local talk) ਨੈੱਟਵਰਕ ਲੀਨੀਅਰ (Linear) ਟੋਪੋਲੋਜੀ (topology) ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹਨ।



Linear Bus topology

ਲੀਨੀਅਰ ਬਸ ਟੋਪੋਲੋਜੀ ਦੇ ਲਾਭਾਂਸ਼ (Advantages of Linear Bus Topology):

- ਲੀਨੀਅਰ ਬਸ (Linear Bus) ਨਾਲ ਕੰਪਿਊਟਰ ਜਾਂ ਪੈਰਿਫਰਲ (peripheral) ਨੂੰ ਜੋੜਣ ਵਿਚ ਆਸਾਨੀ
- ਇਕ ਸਟਾਰ ਨਾਲੋਂ ਘੱਟ ਲੰਬਾਈ ਵਾਲੇ ਕੇਬਲ ਦੀ ਲੋੜ ।

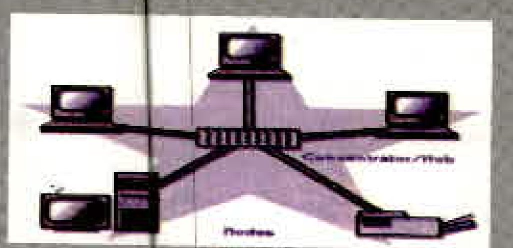
ਲੀਨੀਅਰ ਬਸ ਟੋਪੋਲੋਜੀ ਦੀਆਂ ਹਾਨੀਆਂ (Disadvantages of a Linear Bus Topology):

- ਜੇਕਰ ਮੁੱਖ ਕੇਬਲ ਵਿਚ ਖਰਾਬੀ ਹੋ ਜਾਂਦੀ ਹੈ ਤਾਂ ਸਮੁੱਚਾ ਨੈੱਟਵਰਕ ਬੰਦ ਹੋ ਜਾਂਦਾ ਹੈ।
- ਦੋਹਾਂ ਸਿਰਿਆਂ ਦੇ ਟੋਪੋਲੋਜੀ ਤੇ ਟਰਮੀਨੇਟਰਾਂ (terminators) ਦੀ ਲੋੜ।
- ਜੇਕਰ ਸਮੁੱਚਾ ਨੈਟਵਰਕ ਬੰਦ ਹੋ ਜਾਂਦਾ ਹੈ ਤਾਂ ਸਮੱਸਿਆ ਦੀ ਸ਼ਨਾਖਤ ਕਰਨੀ ਮੁਸ਼ਕਿਲ ਹੁੰਦੀ ਹੈ।

ਇਹ ਵੱਡੀ ਬਿਲਡਿੰਗ ਵਿਚ ਇੱਕਲੇ ਹਲ ਦੇ ਤੌਰ ਤੇ ਨਹੀ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ।

6.3.4 ਸਟਾਰ (Star) : ਸਟਾਰ ਟੋਪੋਲੋਜੀ (star topology) ਹਰੇਕ ਨੋਡ (ਫਾਈਲ ਸਰਵਰ, ਵਰਕ ਸਟੇਸ਼ਨ ਅਤੇ ਪੈਰਿਫਰਲ) ਨੂੰ ਇਕ ਸੈਂਟਰਲ ਨੈੱਟਵਰਕ ਹੱਬ (hub) ਜਾਂ ਕੋਸੈਂਨਟਰੇਟਰ (concentrator) ਦੇ ਨਾਲ ਜੋੜ੍ਹਣ ਲਈ ਡੀਜ਼ਾਈਨ ਕੀਤਾ ਗਿਆ ਹੈ।

ਸਟਾਰ ਨੈਟਵਰਕ ਤੇ ਡਾਟਾ ਆਪਣੀ ਮੰਜ਼ਿਲ ਤੇ ਪਹੁੰਚਣ ਤੋਂ ਪਹਿਲਾਂ ਹੱਬ (hub) ਜਾਂ ਕੋਸੈਂਨਟਰੇਟਰ (concentrator) ਰਾਹੀ ਗੁਜ਼ਰਦਾ ਹੈ ਹੱਬ (hub) ਜਾਂ ਕੋਸੈਂਨਟਰੇਟਰ (concentrator) ਨੈਟਵਰਕ ਦੇ ਸਾਰੇ ਕਾਰਜਾਂ ਦਾ ਪ੍ਰਬੰਧ ਅਤੇ ਕੰਟਰੋਲ ਕਰਦਾ ਹੈ। ਹੱਬ (hub) ਜਾਂ ਕੋਸੈਂਨਟਰੇਟਰ (concentrator) ਨੈਟਵਰਕ ਡਾਟਾ ਪ੍ਰਸਾਰ ਲਈ ਰਿਪੀਟਰ (repeater) ਦਾ ਕੰਮ ਵੀ ਕਰਦਾ ਹੈ। ਇਹ ਰੂਪ-ਰੇਖਾ ਲਚਕੀਲੇ ਕੇਬਲ ਨਾਲ ਸਾਂਝੀ ਹੁੰਦੀ ਹੈ। ਭਾਵੇਂ ਇਸ ਦੀ ਵਰਤੋਂ ਸਹਿਧੁਰੀ ਜਾਂ ਫਾਈਬਰ ਆਪਟਿਕ ਕੇਬਲ ਵਿਚ ਵੀ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਸਟਾਰ ਵਾਇਰਡ ਰਿੰਗ



Star topology

(star-wired ring) ਦੇ ਨਾਲ ਵਰਤੇ ਜਾਣ ਵਾਲੇ ਪ੍ਰੋਟੋਕੋਲ ਆਮ ਤੌਰ ਤੇ ਈਥਰਨੈਟ ਜਾਂ ਲੋਕਲ-ਟਾਕ (local talk) ਹੁੰਦੇ ਹਨ। ਟੋਕਨ ਰਿੰਗ (token ring) ਵਿਚ ਵੀ ਅਜਿਹੇ ਟੋਪੋਲੋਜੀ (topology) ਦੀ ਵਰਤੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਜਿਸ ਨੂੰ ਸਟਾਰ ਵਾਇਰਡ (star wired) ਕਿਹਾ ਜਾਂਦਾ ਹੈ।

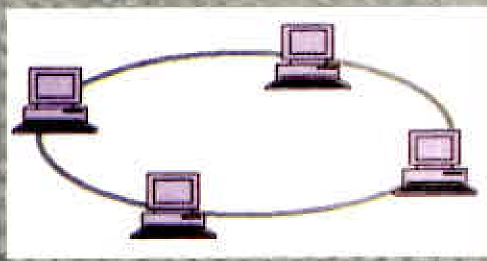
ਸਟਾਰ ਟੋਪੋਲੋਜੀ (star topology) ਦੇ ਲਾਭ

- ਸਥਾਪਤ ਕਰਨਾ ਅਤੇ ਤਾਰਾਂ ਜੋੜਨੀਆਂ ਆਸਾਨ
- ਡਿਵਾਇਸ ਜੋੜਦਿਆਂ ਜਾਂ ਹਟਾਉਂਦੀਆਂ ਨੈੱਟਵਰਕ ਵਿਚ ਕੋਈ ਵਿਘਨ ਨਹੀਂ।
- ਨੁਕਸ ਫੜਨੇ ਆਸਾਨ ਅਤੇ ਪੁਰਜੇ ਹਟਾਉਣੇ ਅਸਾਨ

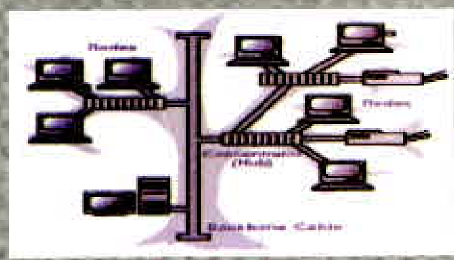
ਸਟਾਰ ਵਾਇਰਡ ਰਿੰਗ (Star-Wired Ring): ਸਟਾਰ ਵਾਇਰਡ ਰਿੰਗ ਟੋਪੋਲੋਜੀ ਸਟਾਰ ਟੋਪੋਲੋਜੀ ਦੀ ਤਰ੍ਹਾਂ ਜਾਪਦਾ (ਬਾਹਰੋਂ) ਹੈ। ਅੰਦਰੂਨੀ ਤੌਰ ਤੇ ਸਟਾਰ (star) ਵਾਇਰਡ ਰਿੰਗ ਵਿਚ ਐਮ ਏ ਯੂ (MAU/multi station access unit) ਇਕ ਸਰਕਲ ਜਾਂ ਰਿੰਗ (circle or ring) ਵਿਚ ਇਕ ਦੂਜੇ ਨਾਲ ਜੁੜੇ ਹੁੰਦੇ ਹਨ ਅਤੇ ਸੂਚਨਾ ਨੂੰ ਇਕ ਡਿਵਾਇਸ (device) ਤੋਂ ਦੂਜੇ ਤਕ ਪਹੁੰਚਣ ਦਾ ਸਾਧਨ ਪ੍ਰਦਾਨ ਕਰਦੀਆਂ ਹਨ। ਟੋਕਨ ਰਿੰਗ (token ring) ਵਿਭਾਗ ਵਲੋਂ ਸਟਾਰ (star) ਵਾਇਰਡ ਵਿਗਿਆਨ ਦੀ ਵਰਤੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।

ਸਟਾਰ ਟੋਪੋਲੋਜੀ (star topology) ਦੀਆਂ ਹਾਨੀਆਂ

- ਲੀਨੀਅਰ (linear) ਟੋਪੋਲੋਜੀ (topology) ਨਾਲੋਂ ਵਧੇਰੇ ਲੰਬੀ ਤਾਰ ਦੀ ਲੋੜ
- ਜੇਕਰ ਹੱਬ (hub) ਜਾਂ ਕੰਨਸੰਨਟਰੇਟਰ (concentrator) ਅਸਫ਼ਲ ਹੋ ਜਾਂਦਾ ਹੈ ਤਾਂ ਜੁੜੀਆਂ ਨੋਡਸ ਨਕਾਰਾ ਹੋ ਜਾਂਦੀਆਂ ਹਨ।



6.3.5 ਟ੍ਰੀ-ਟੋਪੋਲੋਜੀ (Tree topology): ਟ੍ਰੀ-ਟੋਪੋਲੋਜੀ ਵਿਚ ਲੀਨੀਅਰ ਬੱਸ (linear bus) ਅਤੇ ਸਟਾਰ ਟੋਪੋਲੋਜੀ (star topology) ਦੇ ਗੁਣ ਸਾਂਝੇ ਤੌਰ ਤੇ ਮੌਜੂਦ ਹੁੰਦੇ ਹਨ। ਇਸ ਵਿਚ ਸਟਾਰ-ਰੂਪ ਰੇਖਾ ਵਰਕਸਟੇਸ਼ਨਾਂ (workstation) ਦਾ ਇਕ ਸਮੂਹ ਸ਼ਾਮਲ ਹੁੰਦਾ ਹੈ ਜੋ ਲੀਨੀਅਰ ਬੱਸ (linear bus) ਬੈਕ-ਬੋਣ (backbone) ਕੇਬਲ ਟ੍ਰੀ-ਟੋਪੋਲੋਜੀ (Tree topology) ਦੇ ਨਾਲ ਜੁੜਿਆ ਹੁੰਦਾ ਹੈ। ਟ੍ਰੀ-ਟੋਪੋਲੋਜੀ (Tree topology) ਮੌਜੂਦਾ ਨੈੱਟਵਰਕ ਦੇ ਵਿਸਥਾਰ ਦੀ ਆਗਿਆ ਦਿੰਦੀ ਹੈ।



Tree Topology

ਟ੍ਰੀ ਟੋਪੋਲੋਜੀ ਦੇ ਲਾਭ (Advantages of a Tree Topology)

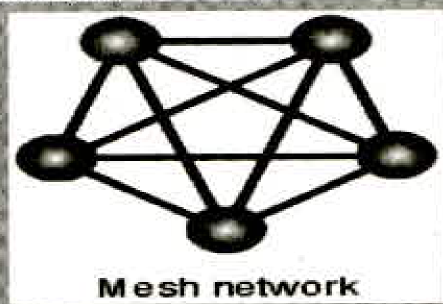
- ਵਿਅਕਤੀਗਤ ਭਾਗਾਂ ਲਈ ਪੁਆਂਇੰਟ-ਟੂ-ਪੁਆਂਇੰਟ ਵਾਇਰਿੰਗ
- ਅਨੇਕਾਂ ਹਾਰਡਵੇਅਰ ਅਤੇ ਸਾਫ਼ਟਵੇਅਰ ਵਿਕਰੇਤਾਵਾਂ ਵਲੋਂ ਸਮਰਥਨ।

ਟ੍ਰੀ ਟੋਪੋਲੋਜੀ ਦੀਆਂ ਹਾਨੀਆਂ (Disadvantages of Tree Topology)

- ਹਰ ਇਕ ਭਾਗ ਦੀ ਸਮੁੱਚੀ ਲੰਬਾਈ ਵਰਤੀ ਗਈ ਤਾਰ ਦੀ ਕਿਸਮ ਦੇ ਨਾਲ ਸੀਮਿਤ ਹੁੰਦੀ ਹੈ।
- ਜੇਕਰ ਬੈਕ-ਬੋਣ (backbone) ਰੇਖਾ ਟੁੱਟ ਜਾਂਦੀ ਹੈ ਤਾਂ ਸਮੁੱਚਾ ਭਾਗ ਹੇਠਾਂ ਆ ਜਾਂਦਾ ਹੈ।
- ਹੋਰ ਟੋਪੋਲੋਜੀ (topology) ਨਾਲੋਂ ਰੂਪ ਰੇਖਾ (configure) ਅਤੇ ਤਾਰਾਂ ਜੋੜਨੀਆਂ ਵਧੇਰੇ ਮੁਸ਼ਕਿਲ

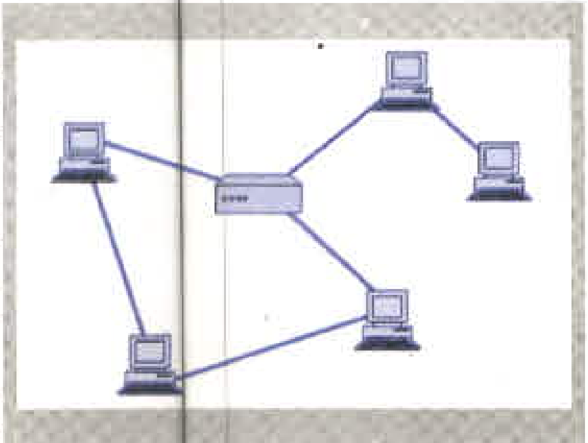
6.3.6 ਮੈਸ਼ ਟੋਪੋਲੋਜੀ (Mesh Topology):

ਨੈੱਟਵਰਕ ਟੋਪੋਲੋਜੀ (topology) ਜਿਸਦੇ ਭਾਗ ਹਰੇਕ ਦੂਜੇ ਭਾਗ ਦੇ ਨਾਲ ਸਿੱਧੇ ਤੌਰ ਤੇ ਜੁੜੇ ਹੁੰਦੇ ਹਨ। **ਮੈਸ਼** ਨੈੱਟਵਰਕ (Mesh Topology) ਇਕ ਲੋਕਲ ਏਰੀਆ ਨੈੱਟਵਰਕ (ਲੈਨ/LAN) ਹੁੰਦਾ ਹੈ ਜੋ ਇਕ ਜਾਂ ਦੋ ਕੁਨੈਕਸ਼ਨ ਪ੍ਰਬੰਧਾਂ ਨੂੰ ਪੂਰਨ **ਮੈਸ਼ ਟੋਪੋਲੋਜੀ** ਆਯੋਜਿਤ ਕਰਦਾ ਹੈ। ਪੂਰਨ **ਮੈਸ਼ ਟੋਪੋਲੋਜੀ (Mesh Topology)** ਵਿਚ ਹਰ ਇਕ ਨੋਡ (ਵਰਕ-ਸਟੇਸ਼ਨ ਜਾਂ ਹੋਰ ਯੰਤਰ **(device)** ਇਕ ਦੂਜੇ ਨਾਲ ਸਿੱਧੇ ਤੌਰ ਤੇ ਜੁੜੀ ਹੁੰਦੀ ਹੈ।



6.3.7 ਪਾਰਸ਼ਲ ਮੇਸ਼ ਟੋਪੋਲੋਜੀ (Partial Mesh Topology) ਵਿਚ ਨੋਡਸ (nodes) ਸਭਨਾਂ ਦੇ ਨਾਲ ਕੁਝ ਸਿੱਧੇ ਅਤੇ ਕੁਝ ਅਸਿੱਧੇ ਤੋਰ ਤੇ ਜੁੜੀਆਂ ਹੁੰਦੀਆਂ ਹਨ, ਜਿਸ ਨਾਲ ਉਹ ਵਧੇਰੇ ਡਾਟੇ ਦਾ ਵਟਾਂਦਰਾ ਕਰਦੀਆਂ ਹਨ।

ਮੇਸ਼ ਨੈੱਟਵਰਕ (Mesh network) ਇਹ ਡਾਟੇ ਦੇ ਵਟਾਂਦਰੇ ਲਈ ਭਰੋਸੇਮੰਦ ਅਤੇ ਅਨੇਕਾਂ ਤਰੀਕੇ ਦਿੰਦਾ ਹੈ। ਜੇਕਰ ਇਕ ਨੋਡ (node) ਸੰਚਾਲਨ (operate) ਨਾ ਕਰ ਸਕੇ, ਬਾਕੀ ਦੀਆਂ ਸਾਰੀਆਂ ਫਿਰ ਵੀ ਇਕ ਦੂਜੇ ਨਾਲ ਸੰਚਾਰ (communication) ਕਰ ਸਕਦੀਆਂ ਹਨ। ਮੇਸ਼ ਨੈੱਟਵਰਕ (Mesh network) ਉਦੋਂ ਚੰਗੀ ਤਰ੍ਹਾਂ ਕੰਮ ਕਰਦਾ ਹੈ ਜਦੋਂ ਨੋਡਸ (nodes) ਬਿਖਰੇ ਪੁਆਇੰਟਾਂ ਤੇ ਸਥਿਤ ਹੁੰਦੀਆਂ ਹਨ, ਜੋ ਸਾਂਝੀ ਲਾਈਨ ਦੇ ਨੇੜੇ ਨਹੀ ਹੁੰਦੀਆਂ।


Mesh tree

ਮੇਸ਼ ਟੋਪੋਲੋਜੀ (Mesh Topology) ਦੀ ਮੁੱਖ ਖਾਮੀ ਹੈ ਖਰਚਾ, ਕਿਉਂ ਜੋ ਬਹੁਤ ਸਾਰੀਆਂ ਕੇਬਲਾਂ ਅਤੇ ਕੁਨੈਕਸ਼ਨਾਂ ਦੀ ਲੋੜ ਹੁੰਦੀ ਹੈ। ਕਈ ਦ੍ਰਿਸ਼ਾਂ ਵਿਚ ਇਕ ਰਿੰਗ ਨੈੱਟਵਰਕ ਜਾਂ ਸਟਾਰ ਨੈੱਟਵਰਕ, ਮੇਸ਼ ਨੈੱਟਵਰਕ ਨਾਲੋ ਵਧੇਰੇ ਲਾਭਦਾਇਕ ਸਿੱਧ ਹੁੰਦਾ ਹੈ। ਜੇਕਰ ਸਮੂਹ ਨੋਡਸ (nodes) ਸਾਂਝੀ ਲਾਈਨ ਦੇ ਨੇੜੇ ਸਥਿਤ ਹੁੰਦੀਆਂ ਹਨ ਤਾਂ ਬਸ ਨੈੱਟਵਰਕ ਟੋਪੋਲੋਜੀ (bus network Topology) ਲਾਗਤ ਦੀ ਟਰਮੀਨੋਲੋਜੀ ਵਿਚ ਉੱਤਮ ਵਿੱਕਲਪ ਹੁੰਦਾ ਹੈ।

6.4.0 ਪ੍ਰੋਟੋਕੋਲਜ਼ (Protocols)

ਇਕ ਪ੍ਰੋਟੋਕੋਲ (Protocols) ਨਿਯਮਾਂ ਜਾਂ ਮਾਪਦੰਡਾਂ ਦਾ ਸੰਪੂਰਨ ਸੈਟ ਹੋ ਸਕਦਾ ਹੈ ਜੋ ਵਿਭਿੰਨ ਯੰਤਰਾਂ ਨੂੰ ਵਾਰਤਾਲਾਪ ਕਰਨ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਮਨੁੱਖੀ ਸੰਚਾਰ (communication) ਭਾਸ਼ਾ, ਵਿਆਕਰਣ ਅਤੇ ਸਭਿਆਚਾਰਕ ਨਿਯਮਾਂ ਤੋਂ ਬਿਨਾਂ ਅਸੰਭਵ ਹੈ। ਨਿਰਧਾਰਿਤ ਨਿਯਮਾਂ ਅਤੇ ਮਾਪਦੰਡਾਂ ਦੇ ਹੇਠ ਲਿਖੀਆਂ ਕੁੱਝ ਮਿਸਾਲਾਂ ਹਨ ਜਿਸ ਵਿਚ ਨੈੱਟਵਰਕ ਸਰਵਿਸ ਯੂਜ਼ਰ (user) ਅਤੇ ਸਰਵਿਸ ਪ੍ਰੋਵਾਇਡਰ (service provider) ਸੰਚਾਰ (communication) ਕਰ ਸਕਦੇ ਹਨ :-

- ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਦੀ ਭੌਤਿਕ ਤੌਰ ਤੇ ਸਥਾਪਨਾ ਕਿਵੇਂ ਹੋਈ
- ਨੈੱਟਵਰਕ ਵਜੂਦ ਕਿਵੇਂ ਇਕ ਦੂਜੇ ਨਾਲ ਸੰਪਰਕ ਕਰਦੇ ਹਨ
- ਨੈੱਟਵਰਕ ਵਜੂਦ ਕਿਵੇਂ ਜਾਣਦੇ ਹਨ ਕਿ ਕਿੰਨਾ ਡਾਟਾ ਸੰਚਾਰ (communication) ਕੀਤਾ ਜਾਣਾ ਹੈ
- ਨੈੱਟਵਰਕ ਵਜੂਦ ਕਿਵੇਂ ਗਰੰਟੀ ਦਿੰਦੇ ਹਨ ਕਿ ਸੰਦੇਸ਼ ਸਹੀ ਪੁੱਜ ਗਿਆ ਹੈ
- ਇਛੁੱਕ (intended) ਪ੍ਰਾਪਤ-ਕਰਤਾ (recipient) ਜਾਂ ਪ੍ਰਾਪਤ-ਕਰਤਾਵਾਂ (recipients) ਨੂੰ ਸੰਦੇਸ਼ ਕਿਵੇਂ ਜਾਂਦਾ ਹੈ
- ਬਿਟਸ (bits) ਦਾ ਪ੍ਰਸਾਰਣ ਮਾਧਿਅਮ ਰਾਹੀਂ ਕਿਵੇਂ ਹੁੰਦਾ ਹੈ
- ਅਲੱਗ ਭਾਸ਼ਾਵਾਂ ਨਾਲ ਨੈੱਟਵਰਕ ਯੰਤਰ ਕਿਵੇਂ ਸੰਚਾਰ (communication) ਕਰਦੇ ਹਨ
- ਇਕ ਨੈੱਟਵਰਕ ਯੰਤਰ ਨੂੰ ਕਿਵੇਂ ਪਤਾ ਲਗਦਾ ਹੈ ਕਿ ਡਾਟਾ-ਸੰਚਾਰ (communication) ਕਦੋਂ ਕੀਤਾ ਜਾਣਾ ਹੈ

6.4.1 ਯੂਨੀਫਾਰਮ ਰਿਸੋਰਸ ਲੋਕੇਟਰ (ਯੂ ਆਰ ਐਲ) [Uniform Resource Locator (URL)]

ਯੂ ਆਰ ਐਲ (URL) ਇਕ web site ਦਾ ਐਡਰੇਸ ਹੁੰਦਾ ਹੈ ਜਿਸ ਤੇ ਇੰਟਰਨੈੱਟ ਰਾਹੀਂ ਪਹੁੰਚਿਆ ਜਾਂਦਾ ਹੈ । web site ਦੀਆਂ ਕਿਸਮਾਂ ਇੰਟਰਨੈੱਟ ਐਪਲੀਕੇਸ਼ਨ ਪ੍ਰੋਟੋਕੋਲ ਤੇ ਨਿਰਭਰ ਕਰਦੀਆਂ ਹਨ।

ਯੂ ਆਰ ਐਲ (URL) ਦੀ ਮਿਸਾਲ ਹੈ **http://www.wwe.org/**

ਇਸ ਯੂ ਆਰ ਐਲ (URL) ਦੇ ਅਨੁਸਾਰ, ਇਕ ਵੈੱਬ ਪੰਨੇ ਨੂੰ HTTP (ਪ੍ਰੋਟੋਕੋਲ) ਦੀ ਵਰਤੋਂ ਨਾਲ ਪਹੁੰਚ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਜਿਹੜਾ ਕਿ www.wwe.org ਨਾਮਕ ਕੰਪਿਊਟਰ ਤੇ ਸਥਿਤ ਹੁੰਦਾ ਹੈ।

6.4.2 ਫਾਈਲ ਟਰਾਂਸਫਰ ਅਤੇ ਹਾਈਪਰਟੈਕਸਟ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ (File Transfer Protocol & Hypertext Transfer Protocol)

ਇਨ੍ਹਾਂ ਨੂੰ ਹੇਠਾਂ ਬਿਆਨ ਕੀਤਾ ਗਿਆ ਹੈ:

6.4.2.1 ਫਾਈਲ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ (ਐਫ ਟੀ ਪੀ) [File Transfer Protocol (FTP)]

ਇਕ ਸਟੈਂਡਰਡ ਇੰਟਰਨੈੱਟ ਪ੍ਰੋਟੋਕੋਲ, ਇੰਟਰਨੈੱਟ ਤੇ ਕੰਪਿਊਟਰਾਂ ਦੇ ਦਰਮਿਆਨ ਫਾਈਲਾਂ ਦੇ ਵਿਚਾਰ ਵਟਾਂਦਰੇ ਦਾ ਸਾਧਾਰਣ ਢੰਗ ਹੁੰਦਾ ਹੈ। ਐਫ ਟੀ ਪੀ (FTP) ਇਕ ਐਪਲੀਕੇਸ਼ਨ ਪ੍ਰੋਟੋਕੋਲ ਹੈ ਜਿਹੜਾ ਇੰਟਰਨੈੱਟ ਦੇ ਟੀ ਸੀ ਪੀ/ਆਈ ਪੀ ਪ੍ਰੋਟੋਕੋਲ (TCP/IP Protocol)ਦੀ ਵਰਤੋਂ ਕਰਦਾ ਹੈ। ਇਹ ਆਮ ਤੌਰ ਤੇ ਸਰਵਰ ਤੋਂ ਤੁਹਾਡੇ ਕੰਪਿਊਟਰ ਵਿਚ ਪ੍ਰੋਗਰਾਮ ਅਤੇ ਹੋਰ ਫਾਈਲਾਂ ਨੂੰ ਡਾਊਨਲੋਡ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

ਬੁਨਿਆਦੀ ਐਫ ਟੀ ਪੀ (FTP) ਸਹੂਲਤ ਆਮ ਤੌਰ ਤੇ ਢੁੱਕਵੇਂ ਪ੍ਰੋਗਰਾਮਾਂ ਰਾਹੀਂ ਪ੍ਰਾਪਤ ਹੁੰਦੀ ਹੈ, ਜਿਹੜੀ TCP/IP ਦੇ ਨਾਲ ਆਉਂਦੀ ਹੈ।

6.4.2.2 ਹਾਈਪਰਟੈਕਸਟ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ / Hypertext Transfer Protocol (HTTP)

ਹਾਈਪਰਟੈਕਸਟ ਟਰਾਂਸਫਰ ਪ੍ਰੋਟੋਕੋਲ (HTTP) ਸੰਚਾਰ (communication) ਪ੍ਰੋਟੋਕੋਲ ਹੈ ਜੋ ਕਿ ਸੂਚਨਾ ਨੂੰ ਇੰਟਰਾਨੈੱਟ (intranet) ਅਤੇ www (world wide web) ਤੇ ਪਹੁੰਚਾਂਦਾ ਹੈ। ਇਸ ਦਾ ਅਸਲੀ ਕੰਮ ਹਾਈਪਰਟੈਕਸਟ ਪੇਜ (Hypertext page) ਨੂੰ ਪ੍ਰਦਰਸ਼ਿਤ ਕਰਨਾ ਜਾਂ ਦਰਸਾਉਣਾ ਹੁੰਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਜਦੋਂ ਤੁਸੀਂ ਬਰਾਊਜ਼ਰ (browser) ਵਿਚ ਯੂ ਆਰ ਐਲ (URL) ਲਿਖਦੇ ਹੋ, ਇਹ ਅਸਲ ਵਿਚ ਵੈੱਬ ਸਰਵਰ (ਇਕ ਕੰਪਿਊਟਰ ਜਿੱਥੇ ਵੈਬ ਪੇਜ਼ ਰਹਿੰਦੇ ਹਨ) ਨੂੰ HTTP ਕਮਾਂਡ ਭੇਜਦਾ ਹੈ, ਇਸਨੂੰ HTML ਵਿਚ ਵੈੱਬ ਪੇਜ ਲਿਆਉਣ ਜਾਂ ਭੇਜਣ ਦੀ ਹਿਦਾਇਤ ਦਿੰਦਾ ਹੈ, ਜਿਹੜਾ ਇਸ ਗਲ ਨੂੰ ਨਿਪਟਾਉਂਦਾ ਹੈ ਕਿ ਵੈੱਬ ਪੇਜ (web pages) ਦੀ ਰੂਪਰੇਖਾ ਕੀ ਹੋਵੇਗੀ ਅਤੇ ਇਹ ਕਿਵੇਂ ਪ੍ਰਦਰਸ਼ਿਤ ਹੋਣਗੇ। ਵੈਬ (web) ਤੇ ਹਰ ਇਕ ਪੰਨੇ ਦਾ ਨਾਮ/ਪਤਾ ਹੁੰਦਾ ਹੈ, ਪਤੇ ਦੀ ਅਗਵਾਈ http:// ਵਲੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ, ਜਿਵੇਂ :- http://www.cdac.org ਹੈ। ਇਸ ਪਤੇ/ਨਾਮ ਨੂੰ ਯੂ ਆਰ ਐਲ (ਯੂਨੀਫਾਰਮ ਰਿਸੋਰਸ ਲੋਕੇਟਰ /URL uniform resource locator) ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਯੂ ਆਰ ਐਲ (URL) ਦੀ ਆਮ ਰੂਪ ਰੇਖਾ ਨਿਮਨ ਅਨੁਸਾਰ ਹੁੰਦੀ ਹੈ:-

1. ਸਰੋਤ (source) ਦੀ ਕਿਸਮ	(ਮਿਸਾਲ ਲਈ : http:// ਜਾਂ ftp://)
2. ਸਰਵਰ (server) ਦਾ ਨਾਮ	(ਮਿਸਾਲ ਲਈ : cdac.org)
3. ਸਰਵਰ ਤੇ ਵਿਸ਼ੇਸ਼ ਫਾਈਲ ਦਾ ਨਾਮ	(ਮਿਸਾਲ ਲਈ : office03.html) ਇਹ ਵਿਕਲਪੀ ਹੈ।

6.4.2.3 ਪ੍ਰਾਈਵੇਟ ਨੈੱਟਵਰਕਸ (Private Networks)

ਕੁਝ ਨੈੱਟਵਰਕ ਨਿਸ਼ਚਿਤ ਤੌਰ ਤੇ ਬਣਾਏ ਜਾਂਦੇ ਹਨ ਅਤੇ ਮੁਕੰਮਲ ਤੌਰ ਤੇ ਵਿਅਕਤੀਗਤ ਸੰਗਠਨਾਂ ਵਲੋਂ ਵਰਤੇ ਜਾਂਦੇ ਹਨ। ਉਨ੍ਹਾਂ ਨੂੰ ਪ੍ਰਾਈਵੇਟ ਨੈੱਟਵਰਕਸ (Private Networks) ਕਹਿੰਦੇ ਹਨ। ਸੰਚਾਰ (communication) ਉਪਕਰਣ ਜਿਹੜੇ ਨੈੱਟਵਰਕ ਬਣਾਉਂਦੇ ਹਨ ਸੰਗਠਨਾਂ ਦੇ ਨਾਮ ਤੇ ਖਰੀਦੇ ਅਤੇ ਪਟੇ (lease) ਤੇ ਲਏ ਜਾਂਦੇ ਹਨ। ਪਹਿਲਾਂ ਵੈਨ (WAN) ਜੋ ਡੀਜ਼ਾਈਨ ਕੀਤਾ ਗਿਆ ਅਤੇ ਵਿਕਸਿਤ ਕੀਤਾ ਗਿਆ ਉਹ ਪ੍ਰਾਈਵੇਟ ਨੈੱਟਵਰਕ (Private Networks) ਦਾ ਨਾਮ ਸੰਯੁਕਤ ਰਾਜ ਨੈੱਟਵਰਕ ਅਰਪਾਨੈਟ (ARPANET) ਸੀ। ਇਸਦਾ ਮਨੋਰਥ ਦਿਲਚਸਪ ਅਤੇ ਖਾਸ ਕੰਪਿਊਟਰ ਸਹੂਲਤਾਂ ਦੇਣੀਆਂ ਸਨ ਜੋ ਉਹ ਕੁਝ ਕੰਪਿਊਟਰ ਸੈਂਟਰਾਂ ਵਿਚ ਉਪਲਬਧ ਸਨ ਅਤੇ ਉਪਭੋਗਤਾਵਾਂ ਦੇ ਵੱਡੇ ਵਰਗ ਲਈ ਪਹੁੰਚਣਯੋਗ ਸਨ।

6.5.0 ਚੈਨਲ ਦੀ ਵਿਚਾਰਧਾਰਾ (Concept of Channel)

6.5.1 ਸੰਚਾਰ ਚੈਨਲ ਕੀ ਹੁੰਦਾ ਹੈ (What is Communication Channel)?

ਜਦੋਂ ਅਸੀਂ ਇਕ ਪੀ ਸੀ (PC) ਤੋਂ ਦੂਜੇ ਨੂੰ ਸੰਦੇਸ਼/ਡਾਟਾ ਭੇਜਦੇ ਹਾਂ ਅਸੀ ਪਹਿਲਾਂ ਡਾਟਾ ਨੂੰ ਇਲੈਕਟ੍ਰੀਕਲ/ਇਲੈਕਟਰੋਮੈਗਨੇਟਿਕ ਊਰਜਾ (electrical/electromagnetic energy) ਵਿਚ ਬਦਲਦੇ ਹਾਂ ਜੋ ਕਿ ਯੰਤਰਾਂ (mechanism) ਨਾਲ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਇਹ ਫਿਰ ਸੰਚਾਰ (communication) ਚੈਨਲ ਰਾਹੀਂ ਲੰਘਾਇਆ ਜਾਂਦਾ ਹੈ। ਇਹ ਚੈਨਲ ਬਿਜਲੀ ਦੀਆਂ ਕੇਬਲਾਂ ਜਾਂ ਉੱਚ ਊਰਜਾ ਇਲੈਕਟਰੋਮੈਗਨੈਟਿਕ ਬੀਮ ਦੇ ਰੂਪ ਵਿਚ ਹੋ ਸਕਦਾ ਹੈ। ਇਹ ਚੈਨਲ ਇਲੈਕਟਰੋਮੈਗਨੈਟਿਕ ਊਰਜਾ ਨੂੰ ਰਸੀਵਰ ਅੰਤ (receiver end) ਕੋਲ ਭੇਜਦਾ ਹੈ ਜਿਹੜਾ ਫਿਰ ਸੰਦੇਸ਼ ਦੇਣ ਲਈ ਬਦਲਿਆ ਜਾਂਦਾ ਹੈ।

ਆਦਰਸ਼ ਸੰਚਾਰ ਚੈਨਲ ਦੇ ਗੁਣ (Characteristics of an Ideal Communication Channel)

(ੳ) ਇਹ ਘਟੋ ਘਟ ਲਾਗਤ ਨਾਲ ਟਰਾਂਸਮੀਟਰ ਤੋਂ ਇਲੈਕਟਰੋਮੈਗਨੈਟਿਕ ਊਰਜਾ (electromagnetic energy) ਦੀ ਵੱਧ ਤੋਂ ਵੱਧ ਮਾਤਰਾ ਨੂੰ ਰਸੀਵਰ (receiver) ਤਕ ਭੇਜਣ ਦੇ ਕਾਬਿਲ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ।

(ਅ) ਇਸ ਵਲੋਂ ਰਸਤੇ ਵਿਚ ਜ਼ਿਆਦਾ ਸ਼ੋਰ ਨਹੀਂ ਕੀਤਾ ਜਾਣਾ ਚਾਹੀਦਾ ਤਾਂ ਜੋ ਰਸੀਵਰ (receiver) ਸੰਦੇਸ਼ ਨੂੰ ਸਹੀ ਤਰ੍ਹਾਂ ਨਾਲ ਸਮਝ ਸਕੇ।

(ੲ) ਟਰਾਂਸਮੀਟਰ (transmitter), ਜਿਥੇ ਭੇਜਣ ਵਾਲਾ ਸਥਿਤ ਹੈ ਅਤੇ ਰਸੀਵਰ (receiver) ਜਿਥੇ ਸਿਗਨਲ ਪ੍ਰਾਪਤ ਹੁੰਦਾ ਹੈ, ਦੋਹਾਂ ਦੇ ਦਰਮਿਆਨ ਦੂਰੀ ਦੀ ਕੋਈ ਬੰਦਸ਼ ਨਹੀਂ ਹੋਣੀ ਚਾਹੀਦੀ।

ਡਿਜ਼ੀਟਲ ਚੈਨਲ ਸਮਰਥਾ (Digital Channel Capacity)

ਇਹ ਅਨੇਕਾਂ ਡਿਜ਼ੀਟ ਮੁੱਲਾਂ ਦੀ ਗਿਣਤੀ ਹੁੰਦੀ ਹੈ ਜੋ ਚੈਨਲ ਵਲੋਂ ਇਕ ਸੈਕਿੰਡ ਵਿਚ ਭੇਜੀ ਜਾਂਦੀ ਹੈ। ਮਾਪ ਬਿਟਸ-ਪ੍ਰਤੀ-ਸੈਕਿੰਡ (ਬੀ ਪੀ ਐਸ) [Bits per second (bps)] ਹੁੰਦਾ ਹੈ। ਇਸ ਨੂੰ ਚੈਨਲ ਦੀ ਬਿਟ (bit) ਦਰ ਵੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਨੈੱਟਵਰਕਿੰਗ ਲਈ ਬਿਟ (bit) ਰੇਟ ਪ੍ਰਤਿ ਸੈਕਿੰਡ ਕਿਲੋਬਿਟਸ (kilo bits) ਵਿਚ ਜਾਂ ਪ੍ਰਤਿ ਸੈਕਿੰਡ ਮੇਗਾਬਿਟਸ (Mega bits) ਪ੍ਰਤਿ ਸੈਕਿੰਡ ਜਾਂ ਐਮ-ਬੀ-ਪੀ-ਐਸ (Mbps) ਬਿਟਸ ਦੇ ਮਿਲੀਅਨ (million) ਵਿਚ ਹੁੰਦਾ ਹੈ।

ਬਿਟ (bit) ਦਰ ਦੀ ਅਵਧੀ ਬਿਟ (bit) ਦਰ ਦਾ ਫੈਸਲਾ ਕਰਦੀ ਹੈ। ਜਿੰਨੀ ਛੋਟੀ ਇਕ ਬਿਟ (bit) ਸਿਗਨਲ ਦੀ ਅਵਧੀ ਹੋਵੇਗੀ ਉਨੀ ਵੱਡੀ ਹੀ ਸਿਗਨਲ ਦੀ ਬੀ ਪੀ ਐਸ (bps) ਦਰ ਹੋਵੇਗੀ।

ਬੋਡ (Baud)

ਡਾਟਾ ਸੰਚਾਰ (communication) ਗਤੀ ਨੂੰ ਮਾਪਣ ਦੀ ਇਕਾਈ। ਇਹ ਇਕ ਅੱਗੇ ਲਿਜਾਣ ਵਾਲੇ ਯੰਤਰ ਦੀ ਸਮਰਥਾ ਨੂੰ ਬਿਆਨ ਕਰਦੀ ਹੈ। ਆਮ ਵਰਤੋਂ ਵਿਚ, **ਬੋਡ** (baud) ਬਿਟਸ (bit) ਪ੍ਰਤਿ ਸੈਕਿੰਡ ਦੇ ਨਾਲ ਮੇਲ ਖਾਂਦਾ ਹੈ।

ਬੈਂਡ-ਵਿਡਥ (Bandwidth)

ਬਿਜਲੀ ਸੰਚਾਰ (communication) ਵਿਚ, ਬੈਂਡਵਿਡਥ (bandwidth) ਰਫ਼ਤਾਰ ਦੇ ਰੇਂਜ ਦੀ ਚੌੜਾਈ ਹੁੰਦੀ ਹੈ, ਜਿਹੜੀ ਕਿ ਇਲੈਕਟਰੋਨਿਕ ਸਿਗਨਲ ਵਲੋਂ ਇਕ ਦਿਤੇ ਗਏ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਦੇ ਵਰਤਣ ਲਈ ਹੁੰਦੀ ਹੈ। ਇਸ ਵਰਤੋਂ ਵਿਚ ਬੈਂਡਵਿਡਥ (bandwidth) ਨੂੰ ਉੱਚ-ਰਫ਼ਤਾਰ ਸਿਗਨਲ ਭਾਗ ਅਤੇ ਘੱਟ ਰਫ਼ਤਾਰ ਸਿਗਨਲ ਭਾਗ ਦੇ ਦਰਮਿਆਨ ਅੰਤਰ ਦੀ ਟਰਮੀਨੋਲੋਜੀ ਵਿਚ ਪ੍ਰਗਟਾਇਆ ਜਾਂਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ ਇਕ ਸਿਗਨਲ ਦੀ ਰਫ਼ਤਾਰ ਹਰਟਜ਼ (Hz) ਵਿਚ ਮਾਪੀ ਜਾਂਦੀ ਹੈ, ਦਿਤਾ ਗਿਆ ਬੈਂਡਵਿਡਥ (bandwidth) ਉੱਚ ਰਫ਼ਤਾਰ ਦੇ ਦਰਮਿਆਨ ਹਰਟਜ (Hz) ਵਿਚ ਅੰਤਰ ਹੁੰਦਾ ਹੈ ਜੋ ਸਿਗਨਲ ਵਲੋਂ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਘੱਟ ਰਫ਼ਤਾਰ ਜਿਹੜੀ ਇਸ ਵਲੋਂ ਵਰਤੀ ਜਾਂਦੀ ਹੈ। ਇਕ ਵਿਸ਼ੇਸ਼ ਆਵਾਜ਼ ਸਿਗਨਲ ਦੀ ਬੈਂਡਵਿਡਥ (bandwidth) ਲਗਭਗ ਤਿੰਨ ਕਿਲੋ ਹਰਟਜ਼ (3kHz) ਹੁੰਦੀ ਹੈ; ਇਕ ਐਨਾਲੋਗ (analogue) ਟੈਲੀਵਿਜ਼ਨ (ਟੀਵੀ) ਬਰਾਡਕਾਸਟ (broadcast) ਵੀਡੀਓ ਸਿਗਨਲ ਦੀ ਬੈਂਡਵਿਡਥ (bandwidth) 6 ਮੈਗਾ ਹਰਟਜ਼ (6MHz) ਹੁੰਦੀ ਹੈ, ਆਵਾਜ਼ ਅਤੇ ਸਿਗਨਲ ਤੋਂ 2000 ਗੁਣਾ ਚੌੜੀ ਹੁੰਦੀ ਹੈ।

ਕਿਸੇ ਸੰਚਾਰ (communication) ਪ੍ਰਣਾਲੀ ਦਾ ਬੈਂਡਵਿਡਥ (bandwidth) ਜਿਨ੍ਹਾਂ ਚੋੜਾ ਹੋਵੇਗਾ ਉਨਾ ਹੀ ਵਧੇਰੇ ਇਹ ਦਿਤੇ ਟਾਈਮ ਵਿਚ ਸੰਚਾਰ (communication) ਕਰ ਸਕੇਗੀ। ਇਸਨੂੰ ਕਿਲੋਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (Kbps) ਵਿਚ ਪ੍ਰਗਟਾਇਆ ਜਾਂਦਾ ਹੈ। ਇਸ ਪ੍ਰਕਾਰ ਇਹ ਦਰ ਨਿਸ਼ਚਿਤ ਕਰਦੀ ਹੈ, ਜਿਸ ਨਾਲ ਸੂਚਨਾ ਨੂੰ, ਸੂਚਨਾ ਸੰਚਾਰ (communication) ਪ੍ਰਣਾਲੀ ਰਾਹੀਂ ਭੇਜੀ ਜਾਂਦਾ ਹੈ।

ਬੈਂਡਵਿਡਥ ਪ੍ਰਭਾਵ (Bandwidth domain)

ਲੈਨ (LAN) ਵਿਚ ਇਹ ਯੰਤਰਾਂ (device) ਦਾ ਇਕ ਸੈੱਟ ਅਰਥਾਤ ਸਮੂਹ ਹੈ ਜੋ ਬੈਂਡਵਿਡਥ (bandwidth) ਦੀ ਵੰਡ (share) ਕਰਦਾ ਹੈ।

6.5.2 ਡਾਟਾ ਬਦਲੀ ਦਰ (Data Transfer Rate)

ਕੰਪਿਊਟਰ ਨੈਟਵਰਕ ਵਿਚ, ਬੈਂਡਵਿਡਥ (bandwidth) ਅਕਸਰ ਡਾਟਾ ਬਦਲੀ ਦਰ ਲਈ ਵਰਤੀ ਜਾਂਦੀ ਹੈ। ਡਾਟਾ ਦੀ ਮਾਤਰਾ ਜਿਹੜੀ ਇਕ ਮਿਥੇ ਸਮੇਂ (ਆਮ ਤੌਰ ਤੇ ਇਕ ਸੈਕਿੰਡ) ਵਿਚ ਇਕ ਪੁਆਇੰਟ ਤੋਂ ਦੂਜੇ ਨੂੰ ਜਾਂਦੀ ਹੈ। ਇਸ ਪ੍ਰਕਾਰ ਦੇ ਬੈਂਡਵਿਡਥ (bandwidth) ਨੂੰ ਵਧੇਰੇ ਕਰਕੇ ਬਿਟਸ ਡਾਟਾ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (bits per second/bps) ਦੇ ਵਿਚ ਪ੍ਰਗਟਾਇਆ ਜਾਂਦਾ ਹੈ। ਕਈ ਵਾਰ ਇਸਨੂੰ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (bits per second/bps) ਬਾਈਟਸ (bytes) ਵਜੋਂ ਪ੍ਰਗਟ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਇਕ ਮਾਡਮ ਜਿਹੜਾ 57,600 ਬੀ ਪੀ ਐਸ (bps) ਤੇ ਕੰਮ ਕਰਦਾ ਹੈ ਦਾ ਇਕ ਮਾਡਮ ਦਾ ਦੁਗਣਾ ਬੈਂਡਵਿਡਥ (bandwidth) ਹੁੰਦਾ ਹੈ ਜਿਹੜਾ 28,000 ਬੀ ਪੀ ਐਸ (bps) ਤੇ ਕੰਮ ਕਰਦਾ ਹੈ। ਆਮ ਤੌਰ ਤੇ ਉੱਚ ਬੈਂਡਵਿਡਥ (bandwidth) ਦੇ ਨਾਲ ਇੰਝ ਉਹ ਹੁੰਦਾ ਹੈ ਜਿਹੜਾ ਵੀਡੀਓ ਪੇਸ਼ਕਸ਼ ਵਿਚ ਪ੍ਰਤਿਬਿੰਬਾਂ ਦੀ ਅਨੁਕ੍ਰਮ ਨੂੰ ਕਾਇਮ ਰਖਣ ਲਈ ਕਾਫ਼ੀ ਸੂਚਨਾ ਲਿਜਾਂਦਾ ਹੈ।

ਦੂਜੇ ਪਾਸੇ ਡਾਟਾ ਬਦਲੀ ਪ੍ਰਤੀ ਬਿਟਸ (bits) ਵਿਚ ਪ੍ਰਗਟਾਈ ਜਾਂਦੀ ਹੈ। ਬਿਟ-ਦਰਾਂ (bit rates) ਵਿਚ ਸੰਖੇਪ ਟਰਮੀਨੋਲੋਜੀ ਨਿਮਨ ਅਨੁਸਾਰ ਹੈ:-

- 1 ਕੇ ਬੀ ਪੀ ਐਸ - 1,000 ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ — 1 kbps = 1,000 bits per second
- 1 ਐਮ ਬੀ ਪੀ ਐਸ - 1,000,000 ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ — 1 Mbps = 1,000,000 bits per second.
- 1 ਜੀ ਬੀ ਪੀ ਐਸ - 1,000,000,000 ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ — 1 Gbps = 1,000,000,000 bits per second

ਕੇ ਬੀ ਪੀ ਐਸ (ਕਿਲੋਬਾਈਟਸ/ਸੈਕਿੰਡ) ਭਾਵ **ਹਜ਼ਾਰਾਂ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ**
kbps (kilobits/sec) means **thousands of bits per second**

ਐਮ ਬੀ ਪੀ ਐਸ (ਮੈਗਾਬਾਈਟਸ/ਸੈਕਿੰਡ) ਭਾਵ **ਮਿਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ**
mbps or Mbps (megabits/sec) means **millions of bits per second**

ਜੀ ਬੀ ਪੀ ਐਸ (ਗੀਗਾਬਾਈਟਸ/ਸੈਕਿੰਡ) ਭਾਵ **ਬਿਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ** ਜਿਥੇ 'ਬਿਲੀਅਨ'=1,000,000,000.00
gbps or Gbps (gigabits/sec) means **billions of bits per second** (where "billion"= 1,000,000,000.00")

ਟੀ ਬੀ ਪੀ ਐਸ (ਟੇਰਾਬਿਟਸ/ਸੈਕਿੰਡ) ਭਾਵ **ਟ੍ਰਿਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ** (ਜਿਵੇਂ ਟੇਰਾਬਿਟਸ ਰੂਟਰ'/ਟੇਰਾਬਿਟ ਸਪੀਡ)
tbps (terabits/sec) means **trillions of bits per second** (as in "terabit router" or "terabit speeds")

ਪੀ ਬੀ ਪੀ ਐਸ (ਪੇਟਾ ਬਿਟਸ/ਸੈਕਿੰਡ) ਭਾਵ **ਕੁਆਡਰੀਲਿਆਨ (quadrillions) ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ**
pbps (petabits/sec) means **quadrillions of bits per second**.

ਬਦਲਵੇਂ ਰੂਪ (Alternate forms):

- **ਬੀ ਬੀ ਪੀ ਐਸ** - ਬਿਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (ਜਿਥੇ ਬੀ - 1,000,000,000)
  bbps = billion bits per second (where b = 1 000 000 000)
- **ਐਮ ਬੀ ਪੀ ਐਸ** - ਮਿਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (ਜਿਥੇ ਐਮ - 1,000,000)
  mbps = million bits per second (where m = 1 000 000)
- **ਕੇ ਬੀ ਪੀ ਐਸ** - ਹਜ਼ਾਰਾਂ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (ਜਿਥੇ ਕੇ - 1,000)
  kbps = thousand bits per second (where k = 1 000)
- **ਟੀ ਬੀ ਪੀ ਐਸ** - ਟਰੀਲੀਅਨ ਬਿਟਸ ਪ੍ਰਤਿ ਸੈਕਿੰਡ (ਜਿਥੇ ਟੀ - 1,000,000,000,000) (ਸਿਧਾਂਤਕ ਤੌਰ ਤੇ)
  tbps = trillion bits per second (where t = 1 000 000 000 000) [theoretical]

ਬਿਟਸ ਅਤੇ ਬਾਈਟਸ (Bits and Bytes):

1 **ਬਾਈਟ** - 8 **ਬਿਟਸ** ;ਕੇ ਬੀ ਪੀ ਐਸ * 0,1220703125 - ਕੇ ਬੀ/ਸੈਕਿੰਡ
bits and Bytes: 1 Byte = 8 bits; kbps* 0.1220703125 = KB/s

ਜਿਵੇਂ ਕਿ, ਇਕ ਬਾਈਟ ਵਿਚ 8 ਬਿਟਸ ਹੁੰਦੀਆਂ ਹਨ, ਬਾਈਟਸ ਵਿਚ ਦਿਤੇ ਗਏ ਮੁੱਲ ਚੋਂ ਬਿਟ ਦਰ (ਗਤੀ) ਲੈਣ ਲਈ, ਤੁਹਾਨੂੰ ਕੁੱਲ ਬਾਈਟਸ ਨੂੰ 8 ਨਾਲ ਗੁਣਾ ਕਰਨਾ ਚਾਹੀਦਾ ਹੈ। ਬਿਟ ਦਰਾਂ ਤੋਂ ਕੇ ਬੀ/ਸੈਕਿੰਡ ਮੁੱਲ ਲੈਣ ਲਈ ਤੁਹਾਨੂੰ ਕੁਲ ਬਿਟਾਂ ਨੂੰ ਤਕਸੀਮ ਕਰਨੀ ਚਾਹੀਦੀ ਹੈ, ਤਦ 1024 ਨਾਲ ਤਕਸੀਮ ਕਰੋ।

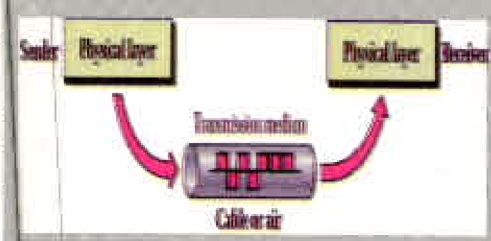
ਕੇ ਬੀ/ਸੈਕਿੰਡ ਨੂੰ ਕੇ ਬੀ ਪੀ ਐਸ (ਬਾਈਟ ਮੁੱਲ ਤੋਂ ਬਿਟ ਦਰ) ਤੇ ਬਦਲਣ ਲਈ ਸਮਾਨਤਾ ਬੁਨਿਆਦੀ ਤੌਰ ਤੇ ਨਿਮਨ ਅਨੁਸਾਰ ਹੁੰਦੀ ਹੈ:

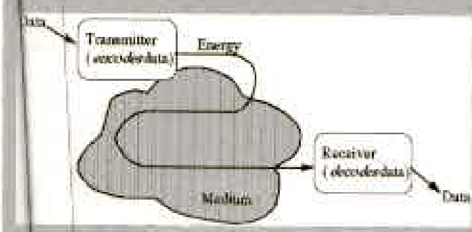
<k> ਕਿਲੋਬਾਈਟਸ * 1024 = <t> ਕੁੱਲ ਬਾਈਟਸ	<K> Kilobytes * 1,024 = <t> total Bytes
<t> ਕੁੱਲ ਬਾਈਟਸ * 8 = <b> ਬਿਟਸ	<t> total Bytes * 8 = <b> bits
<b> ਬਿਟਸ / 1000 = <k> ਕਿਲੋ ਬਾਈਟਸ	<b> bits / 1,000 = <k> kilobits

6.6.0 ਸੰਚਾਰ ਮਾਧਿਅਮ (Transmission Media)

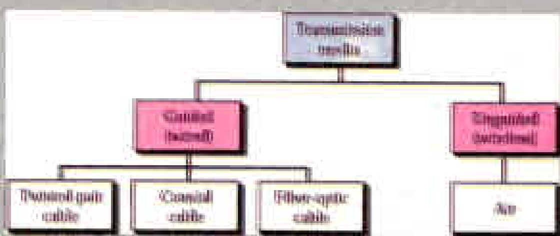
ਕੰਪਿਊਟਰ ਅਤੇ ਹੋਰ ਟੈਲੀਕਮਿਊਨੀਕੇਸ਼ਨ ਡਿਵਾਇਸ ਡਾਟਾ ਦੇਣ ਲਈ ਸਿਗਨਲਾਂ ਦੀ ਵਰਤੋਂ ਕਰਦੇ ਹਨ। ਇਹ ਸਿਗਨਲ ਇਲੈਕਟਰੋਮੈਗਨੈਟਿਕ ਊਰਜਾ ਦੇ ਰੂਪ ਵਿਚ ਡਿਵਾਇਸ ਤੋਂ ਡਿਵਾਇਸ ਤਕ ਸੰਚਾਰ (communication) ਕਰਦੇ ਹਨ ਜਿਹੜੇ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਰਾਹੀਂ ਚਲਦੇ ਹਨ।

ਇਕ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ **(transmission medium)** ਕੋਈ ਵੀ ਸਮਗਰੀ ਪਦਾਰਥ ਹੁੰਦਾ ਹੈ ਜਿਹੜਾ ਲਹਿਰਾਂ ਜਾਂ ਊਰਜਾ ਨੂੰ ਫੈਲਾਉਂਦਾ ਹੈ। ਡਾਟਾ ਤਾਂਬੇ ਦੀਆਂ ਤਾਰਾਂ, ਫਾਈਬਰ ਆਪਟਿਕ ਤਾਰ, ਰੇਡੀਓ ਅਤੇ ਮਾਈਕਰੋਵੇਵਜ਼ ਵਿਚ ਸੰਚਾਰ (communication) ਹੁੰਦਾ ਹੈ। ਸ਼ਬਦ 'ਮਾਧਿਅਮ' ਭੌਤਿਕ ਕੁਨੈਕਟਰਾਂ, ਤਾਰਾਂ ਜਾਂ ਡਿਵਾਇਸ ਜਿਹੜੇ ਚੀਜਾਂ ਨੂੰ ਇਕਠਿਆਂ ਜੋੜਦੇ ਹਨ, ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।





- ਮਾਧਿਅਮ ਨੂੰ ਨਿਮਨ ਅਨੁਸਾਰ ਸ਼੍ਰੇਣੀਬੱਧ ਕੀਤਾ ਗਿਆ ਹੈ:



6.6.1 ਗਾਈਡਿਡ ਮੀਡੀਆ (Guided media)

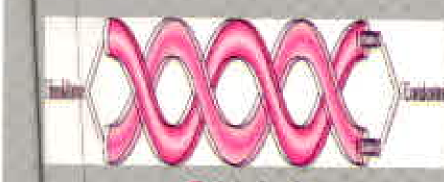
ਗਾਈਡਿਡ ਮਿਡੀਆ ਵਿਚ ਸਿਗਨਲ ਤਾਰਾਂ ਜਾਂ ਲਹਿਰਾਂ (ਫਾਈਬਰ ਆਪਟਿਕਸ) ਰਾਹੀਂ ਭੇਜੇ ਜਾਂਦੇ ਹਨ।

ਕੋ-ਐਕਸ਼ਿਅਲ ਕੇਬਲ (Coaxial Cable) ਠੋਸ ਤਾਰ ਕੋਰ (core) ਦੇ ਨਾਲ ਭਰੇ ਹੁੰਦੇ ਹਨ ਜਿਸਦੇ ਦੁਆਲੇ ਇਕ ਜਾਂ ਵਧੇਰੇ ਧਾਤੂ ਦੀ ਪਤਲੀ ਪਰਤ (foil) ਜਾਂ ਤਾਰਾਂ ਦੀ ਸ਼ੀਲਡ/ਢਾਲ (shield) ਹੁੰਦੀਆਂ ਹਨ ਹਰੇਕ ਪਲਾਸਟਿਕ ਇੰਨਸੁਲੇਟਰ (isulator) ਦੀ ਕਿਸੇ ਕਿਸਮ ਦੁਆਰਾ ਵੱਖ ਹੁੰਦੀਆਂ ਹਨ। ਅੰਦਰੂਨੀ ਕੋਰ (core) ਸਿਗਨਲ (signal) ਲਿਜਾਂਦੀ ਹੈ ਅਤੇ ਸ਼ੀਲਡ/ਢਾਲ (shield) ਗਰਾਉਂਡ (ground) ਪ੍ਰਦਾਨ ਕਰਦੀ ਹੈ।

Coaxial Cable

ਟਵਿਸਟਿਡ ਪੇਅਰ (Twisted Pair)

ਟਵਿਸਟਿਡ ਪੇਅਰ (Twisted Pair) ਵਾਲੀ ਕੇਬਲਿੰਗ ਵਿਚ ਦੋਵੇਂ ਤਾਰਾਂ ਵਿਚ ਇਕ ਦੂਜੇ ਤੇ ਲਿਪਟਿਆਂ ਹੁੰਦੀਆਂ ਹਨ। ਹਰੇਕ ਜੋੜੇ ਵਿਚ ਇਕ ਤਾਰ ਹੁੰਦੀ ਹੈ ਜੋ ਕਿ ਪੋਜ਼ੀਟਿਵ (positive) ਡਾਟਾ ਸਿਗਨਲ ਦੇ ਨਾਲ ਅਤੇ ਇਕ ਤਾਰ ਨੈਗੇਟਿਵ (Negative) ਡਾਟਾ ਸਿਗਨਲ ਨਾਲ ਜੋੜੀ ਹੁੰਦੀ ਹੈ। ਇਹ ਕੋਰ ਲਚਕੀਲੀ ਹੁੰਦੀ ਹੈ ਅਤੇ ਇਛੁੱਕ ਡਿਵਾਇਸ ਪ੍ਰਤਿ ਅਸਾਨੀ ਨਾਲ ਜੁੜ ਸਕਦੀ ਹੈ।


Twisted pair

6.6.2 ਅਨਗਾਈਡਿਡ ਮੀਡੀਆ (Unguided media)

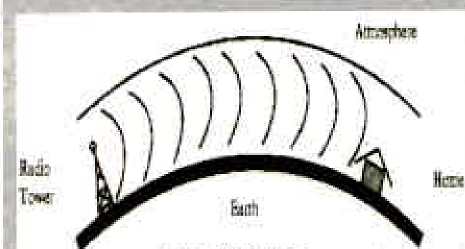
ਅਨਗਾਈਡਿਡ ਮੀਡੀਆ (Unguided media) ਮਾਧਿਅਮ ਵਿਚ, ਸਿਗਨਲ ਸਮੂਹ ਦਿਸ਼ਾਵਾਂ ਵਿਚ ਸੁਤੰਤਰਤਾ ਨਾਲ ਭੇਜੇ ਜਾਂਦੇ ਹਨ ਜਿਵੇ ਰੇਡੀਓ ਜਾਂ ਮਾਈਕਰੋਵੇਵ ਰੇਡੀਏਸ਼ਨ। ਇਹ ਬਹੁਤ ਉੱਚ ਰਫ਼ਤਾਰ ਵਾਲੇ ਸਿਗਨਲ ਵਰਤਦੇ ਹਨ। ਜੋ ਪੁਲਾੜ ਰਾਹੀ ਡਾਟਾ ਸੰਚਾਰ (communication) ਕਰਦੇ ਹਨ। ਡਾਟਾ ਰਿਪੀਟਰ ਸਟੇਸ਼ਨਾਂ ਰਾਹੀਂ- ਗਰਾਉਂਡ (ground) ਰੂਟ ਵਜੋਂ ਸੰਚਾਰ (communication) ਹੁੰਦਾ ਹੈ ਜੋ ਆਮ ਤੌਰ ਤੇ 25 ਮੀਲ ਦੀ ਦੂਰੀ ਤੇ ਹੁੰਦਾ ਹੈ। ਹਰੇਕ ਸਟੇਸ਼ਨ ਡਾਟਾ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ, ਇਸ ਨੂੰ ਐਂਪਲੀਫਾਈ (amplify) ਕਰਦਾ ਹੈ ਅਤੇ ਪੁਨਰ ਸੰਚਾਰ (communication) ਕਰਦਾ ਹੈ।

ਅਨਗਾਈਡਿਡ ਮੀਡੀਆ (Unguided media) ਸੰਚਾਰ ਮਾਧਿਅਮ (communication media) ਇਕ ਡਾਟਾ ਸਿਗਨਲ ਹੈ ਜੋ ਹਵਾ ਰਾਹੀ ਵਹਿੰਦਾ ਹੈ। ਉਹ ਕਿਸੇ ਚੈਨਲ ਵਲੋਂ ਅਪਣਾਏ ਜਾਣ ਜਾਂ ਮਾਰਗ ਦਰਸ਼ਨ ਲਈ ਪਾਬੰਦ ਨਹੀ ਹੁੰਦੇ। ਉਹ ਤਰੰਗ ਪ੍ਰਸਾਰ ਦੀ ਕਿਸਮ ਦੁਆਰਾ ਸ਼੍ਰੇਣੀਬਧ ਕੀਤੇ ਜਾਂਦੇ ਹਨ।

6.6.3 ਆਰ ਐਫ ਪ੍ਰਸਾਰ (RF Propagation)

ਆਰ ਐਫ (ਰੇਡਿਓ ਫਰੀਕੁਏਂਸੀ)/RF (radio frequency) ਪ੍ਰਸਾਰ ਦੀਆਂ 3 ਕਿਸਮਾਂ ਹਨ :

- **ਜ਼ਮੀਨੀ ਤਰੰਗ (Ground Wave)**
- **ਆਈਨੋਸਫੈਰਿਕ (Ionospheric)**
- **ਦਿੱਖ ਪ੍ਰਸਾਰ ਦੀ ਰੇਖਾ [Line of sight (LOS)]**



Ground Wave Propagation

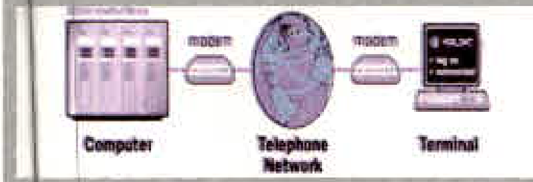
ਜ਼ਮੀਨੀ ਤਰੰਗ ਪ੍ਰਸਾਰ: ਇਹ ਧਰਤੀ ਦੀ ਵਕਰ ਰੇਖਾ ਨੂੰ ਅਪਣਾਉਂਦਾ ਹੈ।

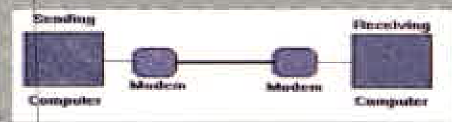
ਮਿਸਾਲ ਲਈ

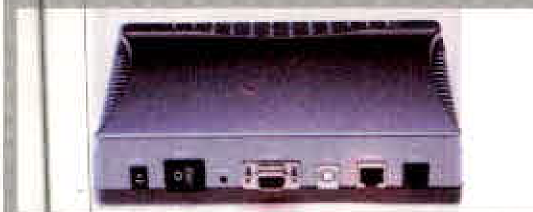
- **ਜ਼ਮੀਨੀ ਤਰੰਗ** ਦਾ ਪ੍ਰਸਾਰ **< 2 MHz** ਹੈ ਅਤੇ ਇਹ ਤਰੰਗਾ AM radio ਵਿਚ ਵਰਤਿਆ ਜਾਂਦੀਆਂ ਹਨ ।
- **ਆਈਨੋਸਫੈਰਿਕ (Ionospheric)** ਤਰੰਗਾ ਦਾ ਪ੍ਰਸਾਰ **2 - 30 MHz** ਹੈ ਅਤੇ ਇਹ International broadcast ਵਿਚ ਵਰਤਿਆ ਜਾਂਦਿਆਂ ਹਨ ।
- **ਦਿੱਖ ਪ੍ਰਸਾਰ ਦੀ ਰੇਖਾ [Line of sight (LOS)] : LOS** ਤਰੰਗਾ ਦਾ ਪ੍ਰਸਾਰ **>30 MHz** ਹੈ ਅਤੇ ਇਹ Mobile phones, Satellite systems , cordless telephones ਵਿਚ ਵਰਤਿਆਂ ਜਾਂਦੀਆਂ ਹਨ ।

6.7.0 ਨੈਟਵਰਕਿੰਗ ਡਿਵਾਇਸ: ਹੱਬ ਅਤੇ ਮੋਡਮ (Networking Devices: Hub & Modems)

ਮੋਡਮਸ (Modems): ਮਾਡਮ **(modem)** ਇਕ ਅਜਿਹਾ ਡਿਵਾਇਸ ਹੈ ਜੋ ਨੈਟਵਰਕ ਤੇ ਡਾਟਾ ਸੰਚਾਰ **(communication)** ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਕੰਪਿਊਟਰ ਦੂਹਰੇ ਕੋਡ (binary code) ਵਿਚ ਮਾਡਮ ਨੂੰ ਡਾਟਾ ਭੇਜਦੇ ਹਨ, ਜਿਹੜਾ ਦੋਹਰੇ ਕੋਡ ਡਾਟਾ ਨੂੰ ਐਨਾਲੋਗ **(analogue)** ਨੈਟਵਰਕ ਵਿਚ ਬਦਲਦਾ ਹੈ। ਇਹ ਡਾਟਾ ਫਿਰ ਟੈਲੀਫੋਨ ਨੈਟਵਰਕ ਦੇ ਨਾਲ ਨਾਲ ਸਫ਼ਰ ਕਰਦਾ ਹੈ। ਜਦੋਂ ਡਾਟਾ ਮੰਜ਼ਿਲ ਤੇ ਪਹੁੰਚ ਜਾਂਦਾ ਹੈ, ਉਸ ਕੰਪਿਊਟਰ ਨਾਲ ਜੁੜਿਆ ਮਾਡਮ ਐਨਾਲੋਗ **(modem analogue)** ਸਿਗਨਲ ਨੂੰ ਦੁਬਾਰਾ ਦੂਹਰੇ ਕੋਡ ਡਾਟਾ ਵਿਚ ਬਦਲਦਾ ਹੈ ਜਿਹੜਾ ਕੰਪਿਊਟਰ ਵਲੋਂ ਪੜ੍ਹਿਆ ਜਾ ਸਕਦਾ ਹੈ। ਇਸ ਪ੍ਰਕਾਰ ਮਾਡਮ ਨੈਟਵਰਕ **(modem Network)** ਤੇ ਡਾਟਾ ਦਾ ਸੰਚਾਰ **(communication)** ਕਰਦਾ ਹੈ।






Modem Back View

Modem Front View

ਮਾਡਮ **(modem)** ਦੋ ਪ੍ਰਕਾਰ ਦੇ ਹੁੰਦੇ ਹਨ, ਬਾਹਰੀ ਅਤੇ
ਅੰਦਰਲੇ। ਬਾਹਰੀ ਮਾਡਮ **(modem)** ... ਪਿਊਟਰ ਅਤੇ
ਟੈਲੀਫੋਨ ਲਾਈਨ ਦੇ ਨਾਲ ਤਾਰਾਂ ਅਤੇ ... ੀ ਦੇ ਸਾਕੇਟ
ਨਾਲ ਜੋੜਿਆ ਜਾਂਦਾ ਹੈ ਜਦਕਿ ਅੰਦਰੂ ... ਪਹਿਲਾਂ ਹੀ
ਕੰਪਿਊਟਰ ਦੇ ਵਿਚ ਬਣਿਆ ਹੁੰ ... ਟੈਲੀਫੋਨ ਦਾ
ਬਿਲਿੰਗ ਮੀਟਰ ਉਦੋਂ ਚਲਣਾ ... ਾ ਹੈ ਜਦੋਂ
ਯੂਜ਼ਰ **(user)** ਇੰਟਰਨੈੱਟ ਦੇ ... ਹੈ। ਇਸ
ਲਈ ਚੰਗੀ ਗਤੀ ਵਾਲਾ ਇ ... **em)** ਘੱਟ
ਟੈਲੀਫੋਨ ਬਿਲਾਂ ਨੂੰ ਯਕੀਨੀ ... 00 **bps**
ਦੀ ਗਤੀ ਵਾਲੇ ਮਾਡਮ ... ਸ਼ ਕੀਤੀ
ਜਾਂਦੀ ਹੈ। ਪਰੰਤੂ ਤੁਸੀਂ ... ਵੀ ਵਰਤੋਂ
ਕਰ ਸਕਦੇ ਹੋ।

ਜਦੋਂ ਫੋਨ ਲਾਈਨਾਂ ਤੇ ਡਾਟਾ ... ਤਾਂ ਇਹ ਡਾਟਾ ਵੱਖ ਵੱਖ ਦਰਾਂ ਤੇ ਪ੍ਰਸਾਰ ਕਰਦਾ ਹੈ। ... ਭੇਜਣ ਵਾਲੇ ਕੰਪਿਊਟਰ ਅਤੇ ਭੇਜਣ ਵਾਲੇ ਮਾਡਮ **(modem)** ਦੇ ਦਰਮਿਆਨ **transfer rate** ਹੁੰਦੀ ਹੈ। ਦੂਜਾ, ਫੋਨ ਲਾਈਨਾਂ ਤੇ ਦੋਹਾਂ ਮਾਡਮਾਂ **(modems)** ਦੇ ਦਰਮਿਆਨ transfer rate ਹੁੰਦੀ ਹੈ। ਤੀਜਾ, ਪ੍ਰਾਪਤ ਕਰਨ ਯੋਗ ਮਾਡਮ **(modem)** ਅਤੇ ਪ੍ਰਾਪਤ ਕਰਨ ਯੋਗ ਕੰਪਿਊਟਰ ਦੇ ਦਰਮਿਆਨ **transfer rate** ਹੁੰਦੀ ਹੈ।

ਹੱਬ (Hub): ਨੈਟਵਰਕਿੰਗ ਹੱਬ **(hub)** ਇਕ ਡਿਵਾਇਸ **(device)** ਹੈ ਜੋ ਤੁਹਾਨੂੰ ਬਹੁਪੱਖੀ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਇਕਹਰੇ ਨੈਟਵਰਕ ਡਿਵਾਇਸ ਦੇ ਨਾਲ ਜੁੜਣ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ। ਸਾਧਾਰਣ ਰੂਪ ਵਿਚ, ਹੱਬ **(hub)** ਇਕ ਪੋਰਟ ਤੋਂ ਪ੍ਰਾਪਤ ਹੋਏ ਡਾਟਾ ਪੈਕੇਟ ਨੂੰ ਡੁਪਲੀਕੇਟਿੰਗ ਕਰਨ ਦਾ ਕੰਮ ਕਰਦਾ ਹੈ ਅਤੇ ਇਸਨੂੰ ਸਮੂਹ ਪੋਰਟਾਂ ਤੇ ਉਪਲਬਧ ਕਰਾਉਂਦਾ ਹੈ। ਇਸ ਲਈ ਹੱਬ **(hub)** ਨਾਲ ਜੁੜੇ ਸਮੂਹ ਯੰਤਰਾਂ ਦੇ ਦਰਮਿਆਨ ਡਾਟਾ ਵੰਡ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ।

ਪੈਸਿਵ ਹੱਬ (passive hub): ਕੇਵਲ ਡਾਟਾ ਪੈਕੇਟਾਂ ਨੂੰ ਇਸ ਰਾਹੀ ਪ੍ਰਸਾਰ ਹੋਣ ਦਿੰਦੀ ਹੈ, ਪ੍ਰਬੰਧਯੋਗ ਹੱਬ **(hub)** ਡਾਟਾ ਦੀ ਨਿਗਰਾਨੀ ਕਰਦੀ ਹੈ ਅਤੇ ਪੋਰਟਾਂ ਨੂੰ ਵਿਅਕਤੀਗਤ ਤੌਰ ਤੇ ਸਮਰੂਪ ਕਰਦੀ ਹੈ।

o

ਸੰਖੇਪ ਰਚਨਾ (Summary)

ਨੈਟਵਰਕਿੰਗ ਸੰਕਲਪ (Networking Concepts)

ਡਾਟਾ ਸੰਚਾਰ ਇਕ ਡਿਵਾਈਸ ਤੋਂ ਦੂਜੇ ਤਕ ਡਾਟਾ ਨੂੰ ਭੇਜਣ ਲਈ ਇਹ ਸੰਚਾਰ ਮਾਧਿਅਮ ਦੇ ਕਿਸੇ ਰੂਪ ਦੁਆਰਾ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ। ਡਾਟਾ ਸੰਚਾਰ ਸਿਸਟਮ ਡਾਟਾ ਨੂੰ ਸਹੀ ਜਗ੍ਹ ਤੇ ਬਿਲਕੁਲ ਠੀਕ ਅਤੇ ਸਮੇ ਅੰਦਰ ਭੇਜਣ ਦੇ ਸਮਰਥ ਹੋਣੀ ਚਾਹੀਦੀ ਹੈ ਪੰਜ ਮਾਧਿਅਮ ਜਿਹੜੇ ਡਾਟਾ ਸੰਚਾਰ ਸਿਸਟਮ ਬਣਾਉਂਦੇ ਹਨ ਵਿਚ ਸ਼ਾਮਲ ਹਨ, message,sender,receiver,medium & Protocol. ਡਾਟਾ ਟਾਈਪ ਤਿੰਨ ਢੰਗਾਂ ਨਾਲ ਦੋ ਡਿਵਾਈਸਿਸ ਵਿਚ ਵਾਪਰ ਸਕਦਾ ਹੈ। ਸਧਾਰਣ, ਹਾਫ਼ ਡੂਪਲੈਕਸ, ਫੁਲ ਡੁਪਲੈਕਸ (simplex, half duplex or full duplex). ਇਕ ਨੈਟਵਰਕ ਸੰਚਾਰ ਉਪਕਰਣਾਂ ਦਾ ਡਿਵਾਈਸ ਹੁੰਦਾ ਹੈ ਜੋ ਮਾਧਿਅਮ ਲਿੰਕਾਂ ਦੁਆਰਾ ਜੁੜਿਆ ਹੁੰਦਾ ਹੈ। ਪੁਆਇੰਟ-ਟੂ-ਪੁਆਇੰਟ : ਇਸ ਕੁਨੈਕਸ਼ਨ ਵਿਚ ਕੇਵਲ ਦੋ ਡਿਵਾਈਸ ਇਕ ਐਸਾਈਨ ਲਿੰਕ ਨਾਲ ਜੁੜੇ ਹੁੰਦੇ ਹਨ। ਮਲਟੀ-ਪੁਆਇੰਟ ਕੁਨੈਕਸ਼ਨ ਵਿਚ ਤਿੰਨ ਜਾਂ ਵਧੇਰੇ ਡਿਵਾਈਸ ਇਕ ਲਿੰਕ ਦੀ ਵੰਡ ਕਰਦੇ ਹਨ। ਟੋਪੋਲੋਜੀ ਵਲੋ ਇਕ ਨੈਟਵਰਕ ਦੇ ਭੌਤਿਕਲ ਅਤੇ ਲੋਜਿਕਲ ਪ੍ਰਬੰਧ ਦਾ ਹਵਾਲਾ ਦਿਤਾ ਜਾਂਦਾ ਹੈ। ਡਿਵਾਈਸ ਮੈਸ਼, ਰਿੰਗ (ring) ਜਾਂ ਸਟਾਰ (star) ਟੋਪੋਲੋਜੀ ਵਿਚ ਵਿਵਸਥਿਤ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ। ਇਕ ਨੈਟਵਰਕ ਨੂੰ LAN ਦੀ ਤਰ੍ਹਾਂ ਸ਼੍ਰੇਣੀਬੱਧ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ ਜਾਂ WAN ਵੀ ਕਿਹਾ ਜਾਂ ਸਕਦਾ ਹੈ। LAN ਇਕ ਡਾਟਾ ਸੰਚਾਰ ਸਿਸਟਮ ਹੈ ਜੋ ਇਕ ਬਿਲਡਿੰਗ, ਪਲਾਂਟ ਜਾਂ ਕੈਮਪਸ ਦੇ ਵਿਚ ਹੁੰਦੀ ਹੈ ਜਾਂ ਨੇੜੇ ਦੀ ਬਿਲਡਿੰਗ ਦੇ ਦਰਮਿਆਨ ਜਦ ਕਿ WAN ਡਾਟਾ ਸੰਚਾਰ ਸਿਸਟਮ ਹੁੰਦਾ ਹੈ ਜੋ ਰਾਜਾਂ, ਦੇਸ਼ਾਂ ਜਾਂ ਸਮੁੱਚੇ ਵਿਸ਼ਵ ਵਿਚ ਹੁੰਦਾ ਹੈ। ਇੰਟਰਨੈੱਟ ਬਹੁ-ਨੈੱਟਵਰਕਾਂ ਦਾ ਇਕ ਨੈਟਵਰਕ ਹੁੰਦਾ ਹੈ ਅਤੇ ਇਹ ਬਹੁਤ ਸਾਰੇ ਨੈਟਵਰਕਾਂ ਦਾ ਸੰਗ੍ਰਹਿ ਹੁੰਦਾ ਹੈ। ਪ੍ਰੋਟੋਕੋਲ ਨਿਯਮਾਂ ਦਾ ਇਕ ਸੈੱਟ ਹੁੰਦਾ ਹੈ ਜੋ ਡਾਟਾ ਸੰਚਾਰ ਦਾ ਪ੍ਰਬੰਧ ਕਰਦਾ ਹੈ। ਪ੍ਰੋਟੋਕੋਲ ਦੇ ਮਹਤੱਵਪੂਰਨ ਅੰਸ਼ ਹਨ ਵਾਕ ਸਟਰਕਚਰ , ਸਟਰਿੰਗ ਬਣਤਰ ਅਤੇ ਟਾਈਮ। ਸਟੈਂਡਰਡ ਜ਼ਰੂਰੀ ਹੁੰਦੇ ਹਨ ਜੋ ਇਹ ਯਕੀਨੀ ਕਰਦੇ ਹਨ ਕਿ ਵੱਖ ਵੱਖ ਨਿਰਮਾਤਾਵਾਂ ਦੀਆਂ ਉਪਜਾਂ ਇਕਠੀਆਂ ਹੀ ਕੰਮ ਕਰ ਸਕਣ। ਸੰਚਾਰ ਮਾਧਿਅਮ guided or unguided ਟਾਈਪ ਦੇ ਹੁੰਦੇ ਹਨ। ਪ੍ਰਮੁੱਖ guided ਵਾਲੇ ਮਾਧਿਅਮ ਹਨ twisted pair, co-axial ਕੇਬਲ ਅਤੇ fibre optics. Unguided ਵਾਲੇ ਮਾਧਿਅਮ ਵਿਚ ਰੇਡੀਓ ਮਾਇਕਰੋਵੇਵ, ਇਨਫਰਾਰੇਡ ਅਤੇ ਲੇਜ਼ਰ ਸ਼ਾਮਲ ਹਨ ਜੋ ਹਵਾ ਰਾਹੀ ਹੁੰਦੇ ਹਨ। ਸੰਚਾਰ ਚੈਨਲ ਵਾਹਨਾਂ ਨੂੰ ਇਕ ਮੰਜ਼ਿਲ ਦੇ ਸੋਰਸ ਤੋਂ ਡਾਟਾ ਪ੍ਰਾਪਤ ਕਰਨ ਲਈ ਸਾਧਨ ਪ੍ਰਦਾਨ ਕਰਦੇ ਹਨ। ਭਾਵੇਂ ਭੌਤਿਕ ਚੈਨਲਾਂ ਵਿਚ ਉਹ ਸਾਂਝੇ ਅੰਸ਼ ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ ਜਿਹੜੇ ਸਿਗਨਲਾਂ ਦੇ ਪ੍ਰਵਾਹ ਆਵਾਜ਼ ਅਤੇ ਡਾਟਾ ਸਿਗਨਲਾਂ ਨੂੰ ਭੇਜਦੇ ਹਨ ਪਰ ਇਸ ਵਿਚ ਵੀ ਅੜਚਨਾਂ ਹੁੰਦੀਆਂ ਹਨ। ਵਿਕਰੇਤਾਂ ਕੋਲ ਅਨੇਕਾਂ ਤਕਨੀਕਾਂ ਹਨ ਜੋ ਇਨ੍ਹਾਂ ਸੀਮਾਵਾਂ ਤੇ ਕਾਬੂ ਪਾਉਂਦੀਆਂ ਹਨ। ਪਰੰਤੂ ਜਿਵੇਂ ਹੀ ਤਕਨਾਲੋਜੀ ਦੇ ਵਿਚ ਸਮੇਂ ਦੇ ਨਾਲ ਸੁਧਾਰ ਆਉਂਦਾ ਗਿਆ ਵਧੇਰੇ ਵਿਕਲਪ ਵੀ ਉਪਲਬਧ ਹੁੰਦੇ ਗਏ। ਸੰਚਾਰ ਚੈਨਲਾਂ ਦੇ ਬੈਂਡ-ਵਿਡਥ ਅਤੇ ਡਾਟਾ ਰੇਟ ਮਹਤਵਪੂਰਨ connectivity characteristics ਹੁੰਦੇ ਹਨ। ਉਹ ਅਕਸਰ ਸਿਸਟਮ ਦੇ ਦਰਮਿਆਨ ਅੰਤਰ ਪੈਦਾ ਕਰਦੇ ਹਨ ਜਿਹੜੇ ਯੂਜ਼ਰ ਦੀਆਂ ਲੋੜਾਂ ਦੀ ਪੂਰਤੀ ਕਰਦੇ ਹਨ। ਇਕ ਚੈਨਲ ਦੇ ਬੁਨਿਆਦੀ ਗੁਣਾਂ ਵਿਚ ਸੁਧਾਰ ਲਿਆਉਣ ਲਈ ਐਨਾਲਾਗ ਅਤੇ ਡਿਜ਼ੀਟਲ ਸਿਗਨਲ ਮਾਡੂਲੇਸ਼ਨ ਤਕਨੀਕਾਂ ਦੀ ਕਿਸਮ ਉਪਲਬਧ ਹੈ। ਪਰੰਤੂ ਇਹ ਤਕਨੀਕਾਂ ਦੀ ਸੀਮਾ ਹੁੰਦੀ ਹੈ ਜਿਹੜੀ ਸਮਝੀ ਜਾਣੀ ਚਾਹੀਦੀ ਹੈ। ਤੁਸੀਂ ਚੈਨਲ ਗੁਣਾਂ ਅਤੇ ਤਕਨੀਕਾਂ ਦੇ ਸੁਮੇਲ ਦੀ ਚੋਣ ਕਰ ਸਕਦੇ ਹੋ ਜਿਹੜੇ ਸਿਸਟਮ ਕਾਰਗੁਜ਼ਾਰੀ ਨੂੰ ਆਸ਼ਾ ਅਨੁਕੂਲ ਬਣਾਉਂਦੇ ਹਨ।

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1 ਬਹੁ ਪਸੰਦੀ ਪ੍ਰਸ਼ਨ

1. ਦੋ ਸਥਿਤੀਆਂ ਦੇ ਦਰਮਿਆਨ ਡਾਟਾ ਬਦਲਣ ਵਾਲੇ ਮਾਧਿਅਮ ਨੂੰ ਕੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
(ੳ) ਨੈਟਵਰਕ (ਅ) ਹਾਰਡ ਡਿਸਕ (ੲ) ਸੀ.ਡੀ.ਰੋਮ (ਸ) ਇਨ੍ਹਾਂ ਸਭ ਦਾ

2. HTTP ਪ੍ਰੋਟੋਕੋਲ ਕਿਸ ਵਾਸਤੇ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ?
(ੳ) ਈ. ਮੇਲ (ਅ) ਨੈਟਵਰਕ ਦੁਆਲੇ ਖ਼ਬਰਾਂ ਭੇਜਣ ਦਾ
(ੲ) www ਤੇ ਪੰਨੇ ਦੇਣ ਲਈ (ਸ) ਇਨ੍ਹਾਂ ਸਾਰਿਆਂ ਦਾ

3. ਨਿਮਨਲਿਖਿਤ ਵਿਚੋਂ ਕਿਹੜਾ ਕੰਪਿਊਟਰ ਨੈਟਵਰਕਾਂ ਦਾ ਨਿਸ਼ਾਨਾ ਨਹੀ ?
(ੳ) ਸਰੋਤ ਵੰਡ (ਅ) ਉੱਚ ਭਰੋਸੇ ਦਾ ਇਕਤਾ (ੲ) ਸਕੇਲੇਬਿਲਟੀ (ਸ) ਉਪਰੋਕਤ ਚੋਂ ਕੋਈ ਨਹੀ

4. ਇਕ distributed network configuration ਜਿਸ ਵਿਚ ਸਮੂਹ ਡਾਟਾ/ਸੂਚਨਾ ਇਕ ਕੇਂਦਰੀ ਕੰਪਿਊਟਰ ਵਿਚੋਂ ਗੁਜ਼ਰਦੀ ਹੈ ?
(ੳ) ਬੱਸ ਨੈਟਵਰਕ (ਅ) ਰਿੰਗ ਨੈਟਵਰਕ
(ੲ) ਸਟਾਰ ਨੈ. ਵਰਕ (ਸ) ਪੁਆਂਇਟ-ਟੂ-ਪੁਆਂਇਟ ਨੈੱਟਵਰਕ

5. PC ਦੀ ਮੁਢਲੀ ਮੈਮਰੀ ਜਾਂ ਡਿਸਕ ਤੋਂ ਕਿਸੇ ਦੂਜੇ ਕੰਪਿਊਟਰ ਨੂੰ ਫਾਈਲ ਭੇਜਣ ਨੂੰ ਕੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
(ੳ) ਅਪਲੋਡਿੰਗ (ਅ) ਡਾਊਨ ਲੋਡਿੰਗ (ੲ) ਲੋਗ-ਇਨ (ਸ) ਹੈਂਗ-ਆਨ

6. ਉਸ ਡਿਵਾਇਸ ਦਾ ਨਾਮ ਦੱਸੋ ਜਿਹੜਾ ਟੈਲੀਫੋਨ ਲਾਈਨਾਂ ਰਾਹੀ ਦੋ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਜੋੜ੍ਹਦਾ ਹੈ ?
(ੳ) ਟੇਪ (ਅ) ਮਾਡਮ (ੲ) ਬੱਸ (ਸ) ਕੇਬਲ

7. ਸੀ ਪੀ ਯੂ ਤੋਂ ਇਕ ਕੰਪਿਊਟਰ ਦੇ ਸੀਮਾਂ ਯੰਤਰਾਂ ਲਈ ਡਾਟਾ ਦੀ ਬਦਲੀ ਪ੍ਰਾਪਤ ਜਿਸ ਰਾਹੀ ਹੁੰਦੀ ਹੈ
(ੳ) ਮਾਡਮ (ਅ) ਕੰਪਿਊਟਰ ਪੋਰਟ (ੲ) ਇੰਟਰਫੇਸ (ਸ) ਬਫ਼ਰ ਮੈਮੋਰੀ

8. ਇਨ੍ਹਾਂ ਨਿਮਨਲਿਖਤ ਵਿਚੋਂ ਕਿਹੜਾ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਨਹੀ ?
(ੳ) ਟੈਲੀਫੋਨ ਲਾਈਨਾਂ (ਅ) ਕੋ-ਐਕਸ਼ਿਅਲ ਕੇਬਲ (ੲ) ਮਾਡਮ
(ਸ) ਮੁੜੀ ਹੋਈ ਕੇ-ਐਕਸਿਅਲ ਕੇਬਲ

9. ਟਰਮੀਨਲਾਂ ਦੀ ਵਰਤੋਂ ਜਿਸ ਲਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ?
(ੳ) ਯੂਜ਼ਰ ਤੋਂ ਡਾਟਾ ਪ੍ਰਵੇਸ਼ ਕਰਨ ਲਈ (ਅ) ਸਕੂਲ ਮੁਖੀ ਨੂੰ ਸੂਚਨਾ ਪ੍ਰਦਾਨ ਕਰਨ ਲਈ
(ੲ) ਨੈਟਵਰਕ ਵਿਚ ਡਾਟਾ ਵੰਡਣ ਲਈ (ਸ) ਇਨ੍ਹਾਂ ਵਿਚੋਂ ਕੋਈ ਵੀ ਨਹੀ

10. ਇੰਟਰਨੈਟ ਤੇ ਪਹੁੰਚ ਕਰਨ ਲਈ PC ਤੇ ਲਗਾਇਆ ਜਾਣ ਵਾਲਾ ਸਾਫਟਵੇਅਰ ਕੀ ਹੁੰਦਾ ਹੈ
(ੳ) www (ਅ) TCP/IP (ੲ) Browser (ਸ) URL

ਪ੍ਰਸ਼ਨ 5 ਹੇਠ ਲਿਖੇ ਪ੍ਰਸ਼ਨਾਂ (3 ਲਾਈਨਾਂ ਤੋਂ ਵੱਧ ਨਾ ਹੋਣ) ਦੇ ਉੱਤਰ ਦਿਓ

1. ਨੈਟਵਰਕ ਕੀ ਹੁੰਦਾ ਹੈ ?
2. ਕੰਪਿਊਟਰ ਨੂੰ ਅਨਾਲੋਗ ਟੈਲੀਫੋਨ ਲਾਈਨ ਨਾਲ ਜੋੜਣ ਵਾਲੇ ਡਿਵਾਇਸ ਦਾ ਨਾਮ ਦੱਸੋ ? ਇਸਦੀਆਂ ਵੱਖ ਵੱਖ ਕਿਸਮਾਂ ਕਿਹੜੀਆਂ ਹਨ ?
3. ਸੰਚਾਰ (communication) ਚੈਨਲ ਦੀ ਪਰਿਭਾਸ਼ਾ ਦੱਸੋ ?
4. ਨੈਟਵਰਕ ਟੋਪੋਲੋਜੀ (topology) ਕੀ ਹੁੰਦਾ ਹੈ ?
5. URL ਕੀ ਹੁੰਦਾ ਹੈ ? www ਕੀ ਹੈ ?
6. ਨੈਟਵਰਕਿੰਗ ਕੀ ਹੁੰਦੀ ਹੈ ?
7. FTP ਕੀ ਹੈ ਅਤੇ ਇਸਦੇ ਲਾਭ ਕੀ ਹਨ ?
8. ਨੈਟਵਰਕ ਹੱਬ ਕੀ ਹੁੰਦਾ ਹੈ ਅਤੇ ਇਸਦੇ ਲਾਭ ਕੀ ਹਨ ?
9. ਡਾਟਾ ਬਦਲੀ ਮੁੱਲ ਸ਼ਬਦ ਦੀ ਪਰਿਭਾਸ਼ਾ ਦਿਓ, ਇਸਦੀਆਂ ਇਕਾਇਆਂ ਕੀ ਹਨ ?
10. ਤੁਸੀਂ ਸ਼ਬਦ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਤੋਂ ਕੀ ਸਮਝਦੇ ਹੋ ?

ਪ੍ਰਸ਼ਨ 6 ਦੱਸੋ ਸਹੀ ਹੈ ਜਾਂ ਗਲਤ

1. ਇਕ ਨੋਡ (ਵਰਕਸਟੇਸ਼ਨ) ਇਕ ਕੰਪਿਊਟਰ ਹੈ ਜੋ ਇਕ ਨੈਟਵਰਕ ਨਾਲ ਜੁੜਿਆ ਹੋਇਆ ਹੈ ?
2. MAN ਨੈਟਵਰਕ ਉਹ ਹਨ ਜੋ ਇਕ ਸ਼ਹਿਰ ਵਿਚ ਫੈਲੇ ਹੁੰਦੇ ਹਨ ?
3. ਦਰ ਜਿਸ ਤੇ ਲੈਨ ਡਾਟਾ ਨੂੰ ਬਦਲਦਾ ਹੈ ਗੀਗਾ-ਬਿਟਸ ਪ੍ਰਤਿਸੈਕਿੰਡ (Giga bits per second) ਹੁੰਦਾ ਹੈ ?
4. ਨੈਟਵਰਕ ਇੰਟਰਫੇਸ ਇਕਾਈ ਇਕ ਡਿਵਾਇਸ ਹੈ ਜੋ ਸਰਵਰ ਨਾਲ ਅਤੇ ਸਮੂਹ ਵਰਕ ਸਟੇਸ਼ਨਾਂ ਨਾਲ ਉਨ੍ਹਾਂ ਦਰਮਿਆਨ ਕੁਨੈਕਸ਼ਨ ਜੋੜਣ ਲਈ ਹੁੰਦੇ ਹਨ ?
5. ਨੈਟਵਰਕ ਸੰਕਲਪ ਵਿਚ ਯੂ ਆਰ ਐਲ ਦਾ ਭਾਵ ਯੂਨੀਫਾਰਮ ਰਿਸੋਰਸ ਲੋਕੇਟਰ ਹੁੰਦਾ ਹੈ ?
6. 1 Gbps 1,000,000,00 ਬਿਟ ਪ੍ਰਤੀ ਸੈਕਿੰਡ ਦੇ ਬਰਾਬਰ ਹੁੰਦਾ ਹੈ ?
7. ਹਵਾ ਵਿਚ ਗੈਰ ਮਾਰਗ ਦਰਸ਼ਕ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਹੈ ?
8. ਡਿਵਾਇਸ ਜਿਹੜਾ ਤੁਹਾਨੂੰ ਬਹੁ ਮੁਖੀ ਕੰਪਿਊਟਰਾਂ ਨੂੰ ਇਕਹਰੇ ਨੈਟਵਰਕ ਨਾਲ ਜੋੜਣ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ ਹੱਬ ਹੁੰਦਾ ਹੈ ?
9. ਬੈਂਡਵਿਡਥ ਮਾਧਿਅਮ ਦੀ ਸਮਰਥਾ ਹੈ ਜਿਸ ਵਿਚ ਸਿਗਨਲ ਦਾ ਸੰਚਾਰ (communication) ਹੁੰਦਾ ਹੈ ?
10. ਜੇਕਰ ਕਿਸੇ ਫਾਈਲ ਦੇ ਨੈਟਵਰਕ ਵਿਚ ਦੋ ਜਾਂ ਵਧੇਰੇ ਕੰਪਿਊਟਰ ਵਿਚ ਕਾਪੀਆਂ ਹੋਣ ਅਤੇ ਜੇਕਰ ਕੋਈ ਇਕ ਉਪਲਬਧ ਨਾ ਹੋਵੇ ਤਦ ਹੋਰ ਕਾਪੀਆਂ ਵਰਤੀਆਂ ਜਾ ਸਕਦੀਆਂ ਹਨ ?

Answer to Objective Questions

Q	.1	.2	.3	.4	.5	.6	.7	.8	.9	.10
Q1	D	D	D	a	a	b	b	c	a	c
Q2	13	5	3	8	2	4	11	6	15	9
Q3	network	Topology	Star topology	Arpanet	Baud	-	-	-	-	-
Q6	True	True	False	True	True	False	True	True	True	True

ਪ੍ਰਸ਼ਨ 2 ਦਿਤੀ ਗਈ ਸੂਚੀ ਚੋਂ ਨਿਮਨਲਿਖਿਤ ਚੋ ਇਕ ਸ਼ਬਦ ਲਿਖੋ

1. ਵਾਇਰਲੈਸ ਲੈਨ	2. ਸਰਵਰ	3. ਲੈਨ	4. ਡਾਟਾ ਵੰਡ
5. ਸੰਚਾਰ (communication) ਚੈਨਲ	6. ਇਲੈਕਟਰੋਨਿਕ ਮੇਲ	7. ਵੈਨ	8. ਇੰਟਰਨੈੱਟ ਸਥਾਨ ਵਿਗਿਆਨ
9. ਓਪਟੀਕਲ ਫਾਈਬਰ	10. ਸਰਵਿਸ ਰੂਟ	11. FTP	12. HTP
13. ਨੈੱਟਵਰਕ ਸਥਾਨ ਵਿਗਿਆਨ	14. ਚੈਟ	15. ਵਿਭਾਗ	16. ਮੁੜ੍ਹੇ-ਜੋੜ੍ਹੇ ਕੇਬਲ

2.1 ਸਟਾਰ, ਰਿੰਗ ਅਤੇ ਬੱਸ ਨੈੱਟਵਰਕ ਇਸ ਦੀਆਂ ਮਿਸਾਲਾਂ ਹਨ ?
2.2 ਦੋ ਸਥਿਤੀਆਂ ਦੇ ਦਰਮਿਆਨ ਡਾਟਾ ਦੀ ਬਦਲੀ ਦੇ ਮਾਧਿਅਮ ਨੂੰ ਇਹ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
2.3 ਇਕ ਛੋਟਾ ਜਿਹਾ ਕੰਪਿਊਟਰ ਨੈੱਟਵਰਕ ਜਿਹੜਾ ਸਥਾਨਕ ਖੇਤਰ ਜਿਵੇਂ ਦਫਤਰ, ਇਮਾਰਤ ਤਕ ਹੀ ਸੀਮਿਤ ਹੁੰਦਾ ਹੈ ?
2.4 ਦੋ ਜਾਂ ਵਧੇਰੇ ਨੈੱਟਵਰਕ ਦੇ ਕੁਨੈਕਸ਼ਨਾਂ ਨੂੰ ਇਹ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
2.5 ਇਹ ਨੈੱਟਵਰਕ ਦੀ ਭਾਸ਼ਾ ਵਿਗਿਆਨ ਹੈ ?
2.6 ਇਹ ਨੈੱਟਵਰਕ ਦਾ ਪ੍ਰਯੋਗ ਹੈ ?
2.7 ਨਿਯਮ ਜਿਹੜਾ ਫਾਈਲ ਬਦਲੀ ਦਾ ਧਿਆਨ ਰੱਖਦਾ ਹੈ ?
2.8 ਤੁਸੀਂ ਇਸ ਰਾਹੀਂ ਕੰਪਿਊਟਰ ਤੇ ਸੰਦੇਸ਼ ਭੇਜ ਅਤੇ ਪ੍ਰਾਪਤ ਕਰ ਸਕਦੇ ਹੋ ?
2.9 ਰੂਲਾਂ ਦਾ ਸੈੱਟ ਜੋ ਪ੍ਰਦਾਨ ਕੀਤੀ ਜਾਂਦੀ ਸੇਵਾ ਦਾ ਪ੍ਰਬੰਧ ਕਰਦਾ ਹੈ ?
2.10 ਕੇਬਲ ਜਿਹੜੀਆਂ ਪਲਾਸਟਿਕ ਜਾਂ ਸ਼ੀਸ਼ੇ ਦੀਆਂ ਬਣੀਆਂ ਹੁੰਦੀਆਂ ਹਨ ਅਤੇ ਜੋ ਮਨੁੱਖ ਦੇ ਵਾਲਾਂ ਜਿੰਨੀਆਂ ਘਣੀਆਂ ਹੁੰਦੀਆਂ ਹਨ ਡਾਟਾ ਸੰਚਾਰ (communication) ਲਈ ਵਰਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ?

ਪ੍ਰਸ਼ਨ 3 ਨਿਮਨਲਿਖਤ ਲਈ ਉਚਿਤ ਸ਼ਬਦ ਲਿਖੋ

1. ਸੰਚਾਰ (communication) ਪ੍ਰਣਾਲੀ ਦੇ ਸਾਧਨ ਦੁਆਰਾ ਜੋੜੇ ਗਏ ਕੰਪਿਊਟਰਾਂ ਦਾ ਸੰਗ੍ਰਹਿ ?
2. ਇਕ ਨੈੱਟਵਰਕ ਵਿਚ ਨੋਡਸ ਦੇ ਅੰਤਰ ਸੰਬੰਧ ਦਾ ਨਮੂਨਾ ?
3. ਟੋਪੋਲੋਜੀ (topology) ਜਿਸ ਵਿਚ ਕੇਂਦਰੀ ਨੋਡ ਸ਼ਾਮਲ ਹੁੰਦੀ ਹੈ ਜਿਸ ਨਾਲ ਹੋਰ ਨੋਡਸ ਹਰ ਇਕ ਮਾਰਗ ਦੁਆਰਾ ਜੁੜੀਆਂ ਹੁੰਦੀਆਂ ਹਨ ?
4. ਵਿਕਸਤ ਅਤੇ ਡੀਜ਼ਾਈਨ ਕੀਤੀ ਗਈ ਪਹਿਲੀ ਵੈਨ (WAN) ਪ੍ਰਾਈਵੇਟ ਨੈੱਟਵਰਕ ਸੀ ?
5. ਡਾਟਾ ਸੰਚਾਰ (communication) ਗਤੀ ਨੂੰ ਮਾਪਣ ਵਾਲੀ ਇਕ ਇਕਾਈ ?

ਪ੍ਰਸ਼ਨ 4 ਨਿਮਨ ਲਿਖਿਤ ਪ੍ਰਸ਼ਨਾਂ (6 ਲਾਈਨਾਂ ਤੋਂ ਵੱਧ ਨਾ ਹੋਣ) ਦੇ ਉੱਤਰ ਦਿਓ

1. ਨੈਟਵਰਕ ਦੀ ਲੋੜ ਕਿਉਂ ਹੁੰਦੀ ਹੈ ?
2. ਡਾਟਾ ਵੰਡ ਨੂੰ ਨੈੱਟਵਰਕਿੰਗ ਦਾ ਮਹਤਵਪੂਰਨ ਪ੍ਰਯੋਗ ਕਿਉ ਮੰਨਿਆ ਜਾਂਦਾ ਹੈ ?
3. ਕੇਬਲ ਟੀ ਵੀ ਨੈੱਟਵਰਕ ਲਈ ਕਿਹੜੀ ਕਿਸਮ ਦਾ ਨੈੱਟਵਰਕ ਲਾਭਦਾਇਕ ਹੁੰਦਾ ਹੈ ? ਇਹ ਵੈਨ (WAN) ਤੋਂ ਕਿਵੇਂ ਵੱਖ ਹੈ ?
4. ਗਾਈਡਿਡ ਮਿਡੀਆ ਅਤੇ ਅਨਗਾਈਡਿਡ ਮਿਡੀਆ ਡਾਟਾ ਸੰਚਾਰ (communication) ਮਾਧਿਅਮ ਕੀ ਹੁੰਦੇ ਹਨ ? ਹਰੇਕ ਦੀ ਮਿਸਾਲ ਦਿਓ ?
5. ਨੈੱਟਵਰਕਿੰਗ ਵਿਚ ਮਾਡਮ ਦਾ ਕੀ ਕੰਮ ਹੁੰਦਾ ਹੈ ?

ਅਧਿਆਇ

7

ਲਾਈਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨਜ਼
Library Functions

ਇਸ ਪਾਠ ਵਿੱਚ ਅਸੀਂ ਹੇਠ ਲਿਖੇ ਵਿਸ਼ਿਆਂ ਬਾਰੇ ਜਾਣਕਾਰੀ ਹਾਸਲ ਕਰਾਂਗੇ :

- ਫੰਕਸ਼ਨ ਬਾਰੇ ਜਾਣ-ਪਛਾਣ
- ਫੰਕਸ਼ਨ ਦਾ ਵਰਗੀਕਰਨ
- ਲਾਇਬ੍ਰੇਰੀ ਫੰਕਸ਼ਨ (ਇਨ ਬਿਲਟ ਫੰਕਸ਼ਨ)
- ਫੰਕਸ਼ਨਾਂ ਦੇ ਲਾਭ

ਫੰਕਸ਼ਨ

ਸੀ (C) ਭਾਸ਼ਾ ਵਿੱਚ ਬਣੇ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਕਈ ਸਟੇਟਮੈਂਟਾਂ ਹੁੰਦੀਆ ਹਨ। ਕਈ ਵਾਰ ਕੁਝ ਸਟੇਟਮੈਂਟਾਂ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਬਾਰ-ਬਾਰ ਆ ਕੇ ਉਸ ਨੂੰ ਵੱਡਾ ਅਤੇ ਗੁੰਝਲਦਾਰ ਬਣਾ ਦਿੰਦੀਆ ਹਨ। ਇਸ ਤਰ੍ਹਾ ਦੇ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਲਿਖਣਾ, ਸਮਝਣਾ ਅਤੇ ਉਸ ਵਿੱਚੋਂ ਗਲਤੀਆ ਲੱਭਣਾ ਔਖਾ ਹੋ ਜਾਂਦਾ ਹੈ। ਕਿਸੇ ਵੀ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਪ੍ਰਭਾਵਸ਼ਾਲੀ ਤਰੀਕੇ ਨਾਲ ਲਾਗੂ ਕਰਨ ਲਈ ਸਾਨੂੰ ਵੱਡੇ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਛੋਟੇ-ਛੋਟੇ ਭਾਗਾਂ ਵਿੱਚ ਵੰਡ ਲੈਣਾ ਚਾਹੀਦਾ ਹੈ। ਪ੍ਰੋਗਰਾਮ ਦੇ ਇਹਨਾਂ ਛੋਟੇ-ਛੋਟੇ ਭਾਗਾਂ ਨੂੰ ਫੰਕਸ਼ਨ ਕਹਿੰਦੇ ਹਨ। ਇਸ ਤੋਂ ਇਲਾਵਾ ਫੰਕਸ਼ਨਾਂ ਦੀ ਸਹਾਇਤਾ ਨਾਲ ਅਸੀਂ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਸੌਖੇ ਸਮਝ ਸਕਦੇ ਹਾਂ। ਸੀ (C) ਭਾਸਾ ਵਿੱਚ ਪ੍ਰੋਗਰਾਮਰ ਇਹਨਾਂ ਫੰਕਸ਼ਨਾਂ ਨੂੰ ਵੱਡਾ ਪ੍ਰੋਗਰਾਮ ਬਣਾਉਣ ਲਈ ਮੁੱਢਲੀ ਇਕਾਈ ਵਜੋਂ ਵਰਤਦਾ ਹੈ। ਫੰਕਸ਼ਨਾਂ ਨੂੰ ਮੁੱਖ ਪ੍ਰੋਗਰਾਮ ਨਾਲ ਮਿਲਾ ਕੇ ਇੱਕ ਪੂਰਾ ਪ੍ਰੋਗਰਾਮ ਬਣਾਇਆ ਜਾਂਦਾ ਹੈ। ਇੱਕ ਵੱਡੇ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਛੋਟੇ ਹਿੱਸਿਆ ਭਾਵ ਫੰਕਸ਼ਨਾਂ ਵਿੱਚ ਵੰਡਣ ਨੂੰ ਮਾਡੂਲਰ ਪ੍ਰੋਗਰਾਮਿੰਗ ਕਹਿੰਦੇ ਹਨ।

ਫੰਕਸ਼ਨ ਆਪਣੇ ਆਪ ਵਿੱਚ ਸਟੇਟਮੈਂਟਾਂ ਦਾ ਸਮੂਹ ਹੈ ਜੋ ਇੱਕ ਖਾਸ ਕੰਮ ਕਰਦਾ ਹੈ। ਇੱਕ ਫੰਕਸ਼ਨ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਕਾਫੀ ਗਿਣਤੀ ਵਿੱਚ ਬਾਰ-ਬਾਰ ਆਉਣ ਵਾਲੀਆਂ ਸਟੇਟਮੈਂਟਾਂ ਨੂੰ ਇੱਕ ਇਕਾਈ ਵਜੋਂ ਪੈਦਾ ਕਰਦਾ ਹੈ ਅਤੇ ਉਹਨਾਂ ਦੀ ਇੱਕ ਨਾਮ ਦੁਆਰਾ ਪਛਾਣ ਦਿਵਾਉਂਦਾ ਹੈ ਜਿਸ ਨੂੰ ਫੰਕਸ਼ਨ ਦਾ ਨਾਮ ਕਹਿੰਦੇ ਹਨ। ਇੱਕ ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਇੱਕ ਜਾਂ ਜ਼ਿਆਦਾ ਫੰਕਸ਼ਨ ਹੋ ਸਕਦੇ ਹਨ ਪ੍ਰੰਤੂ ਮੁੱਖ ਫੰਕਸ਼ਨ ਹਰ ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ। ਬਾਕੀ ਸਾਰੇ ਫੰਕਸ਼ਨ ਮੁੱਖ ਫੰਕਸ਼ਨ ਦੇ ਸਹਾਇਕ ਹੁੰਦੇ ਹਨ।

ਇੱਕ ਪ੍ਰੋਗਰਾਮਰ ਫੰਕਸ਼ਨ ਨੂੰ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਜਿੱਥੇ ਵੀ ਜ਼ਰੂਰਤ ਹੋਵੇ ਬੁਲਾ ਸਕਦਾ ਹੈ। ਭਾਵ ਇੱਕੋ ਤਰ੍ਹਾਂ ਦੀਆਂ ਸਟੇਟਮੈਂਟਾਂ ਬਾਰ-ਬਾਰ ਲਿਖਣ ਦੀ ਲੋੜ ਨਹੀਂ ਹੁੰਦੀ।

ਫੰਕਸ਼ਨਾਂ ਦਾ ਵਰਗੀਕਰਨ

ਆਮ ਤੌਰ ਤੇ ਫੰਕਸ਼ਨ ਦੋ ਤਰ੍ਹਾਂ ਦੇ ਹੁੰਦੇ ਹਨ

1 ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ

2 ਯੂਜ਼ਰ ਡਿਫਾਈਨ ਫੰਕਸ਼ਨ

1 ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ:-

ਜਿਵੇਂ ਕਿ ਨਾਮ ਤੋਂ ਪਤਾ ਚਲਦਾ ਹੈ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਆਮ ਵਰਤੋਂ ਵਾਲੇ ਫੰਕਸ਼ਨਾਂ ਦਾ ਇਕੱਠ ਅਤੇ ਭੰਡਾਰ ਹੈ। ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਵਿੱਚ ਡਿਸਕ ਸਥਿਤ ਹੁੰਦੇ ਹਨ ਅਤੇ ਇਹ ਸਾਡੇ ਲਈ ਕੰਪਾਇਲਰ ਲਿਖਣ ਵਾਲੇ ਲੋਕਾਂ ਦੁਆਰਾ ਲਿਖੇ ਗਏ ਹੁੰਦੇ ਹਨ।

ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਪੁਨਰ ਨਿਰਧਾਰਿਤ ਅਤੇ ਪੁਨਰ ਸੰਗ੍ਰਹਿਤ ਫੰਕਸ਼ਨ ਹਨ ਜਿਹੜੇ ਕਿਸੇ ਖਾਸ ਕੰਮ ਨੂੰ ਕਰਨ ਲਈ ਬਣਾਏ ਜਾਂਦੇ ਹਨ। ਇਹਨਾਂ ਨੂੰ ਬਿਲਿਟ ਇੰਨ ਫੰਕਸ਼ਨ ਵੀ ਕਹਿੰਦੇ ਹਨ ਇਹ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਕਿਸੇ ਖਾਸ ਲਾਇਬਰੇਰੀ ਫਾਈਲ ਵਿੱਚ ਰੱਖੇ ਜਾਂਦੇ ਹਨ। ਜੇਕਰ ਅਸੀਂ ਕਿਸੇ ਖਾਸ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਵਰਤਨਾ ਚਾਹੁੰਦੇ ਹੋਈਏ ਤਾਂ ਸਾਨੂੰ ਉਸ ਨਾਲ ਸਬੰਧਿਤ ਲਾਇਬਰੇਰੀ ਫਾਈਲ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਸ਼ਾਮਿਲ ਕਰਨਾ ਪੈਂਦਾ ਹੈ ਉਦਾਹਰਣ ਵਜੋਂ ਜੇਕਰ ਅਸੀਂ scanf( ), printf( ) ਵਰਤਣਾ ਚਾਹੁੰਦੇ ਹਾਂ, ਸਾਨੂੰ stdio.h ਫਾਈਲ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਸ਼ਾਮਿਲ ਕਰਨਾ ਪਵੇਗਾ। ਇਸ ਲਈ ਕਿਸੇ ਵੀ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਵਰਤਨ ਤੋਂ ਪਹਿਲਾਂ ਉਸ ਨਾਲ ਸਬੰਧਿਤ ਹੈਡਰ ਫਾਈਲ ਨੂੰ ਸ਼ਾਮਿਲ ਕਰਨਾ ਹੋਵੇਗਾ। ਇਸ ਹੈਡਰ ਫਾਈਲ ਵਿੱਚ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਨਾਲ ਸਬੰਧਿਤ ਸਾਰੀ ਜਾਣਕਾਰੀ ਸ਼ਾਮਿਲ ਹੁੰਦੀ ਹੈ। ਇੱਕ ਵਾਰ ਹੈਡਰ ਫਾਈਲ ਸ਼ਾਮਿਲ ਹੋਣ ਤੋਂ ਬਾਅਦ ਲਾਇਬਰੇਰੀ ਵਿੱਚ ਮੌਜੂਦ ਕੋਈ ਵੀ ਫੰਕਸ਼ਨ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਸ਼ਾਮਿਲ ਕਰ ਸਕਦੇ ਹਾਂ। ਅਸੀਂ ਕਹਿ ਸਕਦੇ ਹਾਂ ਕਿ ਸਾਰੇ ਮੌਜੂਦਾ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਇੱਕ ਖਾਸ ਫਾਈਲ ਵਿੱਚ ਮੌਜੂਦ ਹਨ। ਇਸ ਲਈ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨਾਂ ਦੇ ਸਹੀ ਢੰਗ ਨਾਲ ਕੰਮ ਕਰਨ ਲਈ ਪ੍ਰੋਗਰਾਮ ਦੇ ਸ਼ੁਰੂ ਵਿੱਚ ਹੀ ਹੈਡਰ ਫਾਈਲ ਨੂੰ ਸ਼ਾਮਿਲ ਕਰਨਾ ਜ਼ਰੂਰੀ ਹੋ ਜਾਂਦਾ ਹੈ।

ਸੀ-ਭਾਸ਼ਾ ਵਿੱਚ ਵੱਖ-ਵੱਖ ਪ੍ਰੋਗਰਾਮ ਇੱਕ ਸਾਂਝਾ ਫੰਕਸ਼ਨ ਵਰਤ ਸਕਦੇ ਹਨ। ਅਸੀਂ ਇਹਨਾਂ ਫੰਕਸ਼ਨਾਂ ਦੀ ਵਰਤੋਂ ਆਪਣੇ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਜ਼ਰੂਰਤ ਅਨੁਸਾਰ ਕਰ ਸਕਦੇ ਹਾਂ।

ਸੀ-ਭਾਸ਼ਾ ਵਿੱਚ ਕਾਫ਼ੀ ਗਿਣਤੀ ਵਿੱਚ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਮੌਜੂਦ ਹਨ। ਕੁਝ ਮਹੱਤਵਪੂਰਨ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਅਤੇ ਉਹਨਾਂ ਨਾਲ ਸਬੰਧਿਤ ਹੈਡਰ ਫਾਇਲਾਂ ਹੇਠਾਂ ਦਿੱਤੀਆਂ ਹਨ।

1 ਸਟੈਂਡਰਡ ਇਨਪੁੱਟ ਅਤੇ ਆਊਟ ਪੁੱਟ ਔਪਰੇਸ਼ਨਜ਼ ਨਾਲ ਸਬੰਧਿਤ ਸਾਰੇ ਹੈਡਰ ਫੰਕਸ਼ਨ stdio.h ਇਹਨਾਂ ਵਿੱਚੋਂ ਕੁਝ ਹੇਠ ਲਿਖੇ ਹਨ:-
scanf( ), printf( ), getchar( ), fclose( ), fopen( ),

2 ਫਾਈਲ ctype.h ਵਿੱਚ tolower(c), toupper(c), toascii(c), isupper(c) ਆਦਿ ਫੰਕਸ਼ਨਜ਼ ਮੌਜੂਦ ਹਨ।

3 ਸਟਰਿੰਗ ਨਾਲ ਸਬੰਧਿਤ ਫੰਕਸ਼ਨਜ਼ strlen( ),strcmp(),strcpy(), strcat() ਆਦਿ ਫੰਕਸ਼ਨਜ਼ string.h ਫਾਈਲ ਵਿੱਚ ਮੌਜੂਦ ਹਨ।

4 ਮੈਥ ਨਾਲ ਸਬੰਧਿਤ ਫੰਕਸ਼ਨ ਜਿਵੇਂ ਕਿ cos( ), fmod( ),sqrt(),pow() ,exp( ) math.h ਫਾਈਲ ਵਿੱਚ ਮੌਜੂਦ ਹਨ।

ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ

Strlen( )

ਇਹ ਫੰਕਸ਼ਨ ਕਿਸੇ ਵੀ ਫੰਕਸ਼ਨ ਦੀ ਲੰਬਾਈ ਮਾਪਣ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਇਹ ਕਿਸੇ ਫੰਕਸ਼ਨ ਵਿੱਚ ਮੌਜੂਦ ਸ਼ਬਦਾਂ ਦੀ ਗਿਣਤੀ ਕਰਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

```
main( )
{
   char a[ ]= "sehij" ;
   int len ;
   len= strlen(a) ;
   printf("\n string = %s  length = %d  ", a, len);
 }
```

Output will
String = sehij   length=5

Strcpy( )

ਇਹ ਫੰਕਸ਼ਨ ਕਾਪੀ ਕੀਤੇ ਕਨਟੈਂਟ ਨੂੰ ਇੱਕ ਸਟਰਿੰਗ ਤੋਂ ਦੂਸਰੇ ਸਟਰਿੰਗ ਤੱਕ ਲੈ ਕੇ ਜਾਂਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

```
main( )
{
  char source[ ] = "satik" ;
   char target[ ] ;
   strcpy(target , source) ;
   printf("\n source= %s" ,source) ;
   printf("\n target=%s" ,target) ;
}
```

Output will be following
Source string = satik
Target string = satik

Strcat( )

ਇਹ ਫੰਕਸ਼ਨ ਦੋ ਸਟਰਿੰਗਜ ਨੂੰ ਆਪਸ ਵਿੱਚ ਜੋੜਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

Strcat(string1, string2)

String1 ਅਤੇ and String2 ਕਰੈਕਟਰ ਐਰੇ ਹਨ। ਜਦੋਂ strcat ਫੰਕਸ਼ਨ ਨੂੰ ਰਨ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਤਾਂ string2 , string1 ਨਾਲ ਜੁੜ ਜਾਂਦਾ ਹੈ। ਜਦਕਿ ਵਿੱਚ ਕੋਈ ਫਰਕ ਨਹੀਂ ਪੈਂਦਾ । ਜਿਵੇਂ ਕਿ

```
main()
{
  char source[ ]= "GHS" ;
  Char target[ ]= "Tapa" ;
  strcat(target,source);
  printf("\n source string = %s", source);
  printf("\n target string =  %s", target);
 }
```

```
Output
source string =GHS
target string = GHStapa
```

ਨੋਟ : ਟਾਰਗਿਟ ਸਟਰਿੰਗ ਇਨਾਂ ਵੱਡਾ ਹੋਣਾ ਚਾਹੀਦਾ ਹੈ ਕਿ ਫਾਈਨਲ ਸਟਰਿੰਗ ਨੂੰ ਸਟੋਰ ਕਰ ਸਕੇ ।

Strcmp( )

ਇਹ ਫੰਕਸ਼ਨ ਦੋ ਸਟਰਿੰਗਾਂ ਦਾ ਮੁਕਾਬਲਾ ਕਰਕੇ ਇਹ ਪਤਾ ਲਗਾਉਂਦਾ ਹੈ ਕਿ ਉਹ ਇੱਕੋ ਜਿਹੇ ਹਨ ਕਿ ਵੱਖਰੇ ਦੋ ਸਟਰਿੰਗਾਂ ਨੂੰ ਵਿਸ਼ੇਸ਼ਤਾ ਅਨੁਸਾਰ ਉਸ ਵੇਲੇ ਤੱਕ ਮਿਲਾ ਕਿ ਵੇਖਿਆ ਜਾਂਦਾ ਹੈ ਜਦੋਂ ਤੱਕ ਉਹਨਾਂ ਦਾ ਮਿਲਾਣ ਹੋ ਜਾਵੇ ਜਾ ਇੱਕ ਸਟਰਿੰਗ ਖ਼ਤਮ ਹੋ ਜਾਵੇ। ਜੇਕਰ ਦੋ ਸਟਰਿੰਗ ਇੱਕੋ ਜਿਹੇ ਹੋਣ ਤਾਂ ਇਹ ਫੰਕਸ਼ਨ ਜੀਰੋ ਰਿਟਰਨ ਕਰੇਗਾ। ਪ੍ਰੰਤੂ ਜੇਕਰ ਇਹ ਦੋਵੇਂ ਇੱਕੋ ਜਿਹੇ ਨਹੀਂ ਹਨ ਤਾਂ ਫੰਕਸ਼ਨ ਕੁਝ ਗਣਿਤਕ ਅੰਤਰ ਦਿਖਾਵੇਗਾ। ਜੋ ਕਿ ਅਸਕਾਈ (ASCII) ਕੀਮਤ ਅਤੇ ਨਾ ਮੈਚ ਕਰਨ ਵਾਲੀਆਂ ਵਿਸ਼ੇਸ਼ਤਾਵਾਂ ਵਿਚਲਾ ਅੰਤਰ ਹਨ । ਉਦਾਹਰਣ ਲਈ

```
main()
{
  char string1[]= "harpreet" ;
  char string2[]= "ghs tapa" ;
  int i,j,k;

 i=strcmp(string1, "harpreet");
 j=strcmp(string2,string1) ;

 printf("\n %d %d" , i , j );
```

OUTPUT

0 1

getchar( )

ਇਸ ਵਿੱਚ ਸਟੈਂਟਡ ਇਨਪੁੱਟ ਡੀਵਾਇਸ ਇੱਕ ਕਰੈਕਟਰ ਇਨਪੁੱਟ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

```
#include<stdio.h>
main( )
{
   Char ans;
    Printf("Like to know my name ? \n") ;
    Printf("Type Y for yes and  N for no : ") ;
    ans=getchar();
    if(ans=='y'||ans=='Y' \n);
    printf("\n\n Terni ");
    else
    printf("\n\n thanx") ;
}
```

Putchar ( )

getchar( ) ਫੰਕਸ਼ਨ ਦੀ ਤਰ੍ਹਾਂ ਇੱਕ ਹੋਰ ਫੰਕਸ਼ਨ putchar ( ) ਹੈ ਜੋ ਕਿ ਇੱਕ ਸਮੇਂ ਤੇ ਇੱਕ ਕਰੈਕਟਰ ਲਿਖਣ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।

```
ans='y' ;
putchar(ans);
```

will display the character Y on the screen.

```
Putchar("\n");
```

Would cause the cursor or screen to move to the next line.

Printf( )

ਡਾਟੇ ਨੂੰ ਸਟੈਂਡਰਡ ਆਊਟਪੁੱਟ ਯੰਤਰ ਤੱਕ ਭੇਜਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

```
Printf("GHS Tapa")
```

Output will be :- GHS Tapa

Scanf( )

ਇਹ ਫੰਕਸ਼ਨ ਸਟੈਂਡਰਡ ਇਨਪੁੱਟ ਡੀਵਾਇਸ ਤੋਂ ਡਾਟਾ ਪ੍ਰਾਪਤ ਕਰਦਾ ਹੈ। ਇਹ ਹੇਠ ਲਿਖੇ ਅਨੁਸਾਰ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ।

```
Scanf("%f ",&x);
```

Here x variable is assume as float.

```
Scanf("%d ", &x);      here x is integer
```

Cos( )

It return the cosine of a radian.

Exp ( )

exp(x) ਇਹ ਫੰਕਸ਼ਨ ਈ (e) ਦੀ ਘਾਤ ਜਿਵੇਂ ਕਿ x ਦੀ ਕੀਮਤ ਨਿਰਧਾਰਤ ਕਰਦਾ ਹੈ। ਜਿਥੇ ਕਿ e=2.7183 ਹੁੰਦੀ ਹੈ।

exp(0) output twill be 2.71

Pow( )

pow(a,b) ਦਾ ਭਾਵ ਹੈ a ਦੀ ਘਾਤ b.

e.g pow(2,2)

output will be 4

Sqrt( )

Sqrt(f) ਇਹ ਫੰਕਸ਼ਨ f ਦਾ ਵਰਗਮੂਲ ਪਤਾ ਕਰੇਗਾ।

e.g sqrt(36)

output will be 6

fmod( )

ਇੱਕ ਨੰਬਰ ਨੂੰ ਕਿਸੇ ਹੋਰ ਨੰਬਰ ਨਾਲ ਵੰਡਦੇ ਹਾਂ ਤਾਂ ਬਾਕੀ ਕੀ ਬਚੇਗਾ ਦੀ ਗਣਨਾ ਕਰੇਗਾ। ਜਿਵੇਂ ਕਿ

fmod(10,3)

Output will be 1

Fabs( )

ਇਹ ਫੰਕਸ਼ਨ ਦਿੱਤੀ ਹੋਈ ਕੀਮਤ ਦਾ ਨਿਰਪੇਖ ਮੁੱਲ ਪਤਾ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ

Fabs(-24) return 24 as output.

ceil( )

ਇਹ ਫੰਕਸ਼ਨ ਦਿੱਤੀ ਹੋਈ ਨੂਮੇਰਿਕ ਕੀਮਤ ਨੂੰ ਨੇੜਲੇ ਅਜਿਹੇ ਪੂਰਨ ਅੰਕ ਵਿੱਚ ਬਦਲਦਾ ਹੈ ਜੋ ਕਿ ਦਿੱਤੀ ਹੋਈ ਵੈਲਊ ਦੇ ਬਰਾਬਰ ਹੁੰਦਾ ਹੈ ਜਾਂ ਫਿਰ ਉਸ ਤੋਂ ਵੱਧ ਹੁੰਦਾ ਹੈ।

floor(y)

y ਨੂੰ ਨੇੜੇ ਦੇ ਹੇਠਲੇ ਪੂਰਨ ਅੰਕ ਵਿੱਚ ਬਦਲਿਆ ਜਾਂਦਾ ਹੈ।

islower( )

ਇਹ ਇੱਕ ਕੰਡੀਸ਼ਨਲ ਫੰਕਸ਼ਨ ਹੈ ਜੋ ਇਹ ਦਰਸਾਉਂਦਾ ਹੈ ਕਿ ਕਰੈਕਟਰ ਆਰਗੂਮੈਂਟ ਲੋਅਰ ਕੇਸ ਵਿੱਚ ਹੈ ਜਾਂ ਨਹੀਂ।

isupper( )

ਇਹ ਇੱਕ ਕੰਡੀਸ਼ਨਲ ਫੰਕਸ਼ਨ ਹੈ ਜੋ ਇਹ ਦਰਸਾਉਂਦਾ ਹੈ ਕਿ ਕਰੈਕਟਰ ਆਰਗੂਮੈਂਟ ਅਪਰ ਕੇਸ ਵਿੱਚ ਹੈ ਜਾਂ ਨਹੀਂ।

tolower( )

ਇਹ ਫੰਕਸ਼ਨ ਕਰੈਕਟਰ ਆਰਗੂਮੈਂਟ ਨੂੰ ਲੋਅਰ ਕੇਸ ਵਿੱਚ ਤਬਦੀਲ ਕਰ ਦਿੰਦਾ ਹੈ।

toupper( )

ਇਹ ਫੰਕਸ਼ਨ ਕਰੈਕਟਰ ਆਰਗੂਮੈਂਟ ਨੂੰ ਅਪਰ ਕੇਸ ਵਿੱਚ ਤਬਦੀਲ ਕਰ ਦਿੰਦਾ ਹੈ।

Example of above functions used in a program:-

```
#include<stdio.h>
#include<ctype.h>
main( )
{
 char alpha ;
 printf(" \n Enter an alphabet \n");
 alpha=getchar( );
 if(islower(alpha)) ;
 Putchar(toupper(alpha));
 else
 putchar(tolower(alpha)) ;
}
```

Output

Enter an alphabet

h

H
Enter an alphabet
Q
q

ਫੰਕਸ਼ਨਾਂ ਦੇ ਲਾਭ

ਫੰਕਸ਼ਨਾਂ ਦੇ ਲਾਭ ਹੇਠ ਲਿਖੇ ਹਨ :-

1 ਫੰਕਸ਼ਨਜ਼ ਦੀ ਸਹਾਇਤਾ ਨਾਲ ਅਸੀਂ ਗੁੰਝਲਦਾਰ ਪ੍ਰੋਗਰਾਮਾਂ ਨੂੰ ਸੌਖਿਆਂ ਹੀ ਲਿਖ ਸਕਦੇ ਹਾਂ।

2 ਫੰਕਸ਼ਨਜ਼ ਦੀ ਸਹਾਇਤਾ ਨਾਲ ਵੱਡੇ ਪ੍ਰੋਗਰਾਮਾਂ ਨੂੰ ਸੌਖਿਆਂ ਹੀ ਅਤੇ ਘੱਟ ਸਮੇਂ ਵਿੱਚ ਡਿਜ਼ਾਈਨ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ। ਇਸ ਤਰ੍ਹਾਂ ਪ੍ਰੋਗਰਾਮ ਦਾ ਅਕਾਰ ਛੋਟਾ ਹੋ ਜਾਂਦਾ ਹੈ।

3 ਹਰੇਕ ਪ੍ਰੋਗਰਾਮ ਨੂੰ ਸੌਖਿਆਂ ਹੀ ਸਮਝਿਆ ਜਾ ਸਕਦਾ ਹੈ।

4. ਇਕ ਫੰਕਸ਼ਨ ਨੂੰ ਵੱਖ-ਵੱਖ ਸਿਸਟਮਾਂ ਉੱਤੇ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ ਕਿਉਂਕਿ ਇਹ ਸਿਸਟਮ ਡਿਪੈਨਡੈਂਟ ਨਾ ਹੋ ਕੇ ਇੰਡੀਪੈਨਡੈਂਟ ਕੰਮ ਕਰਦੇ ਹਨ।

5. ਇੱਕੋ ਹੀ ਫੰਕਸ਼ਨ ਨੂੰ ਇੱਕ ਤੋਂ ਵੱਧ ਪ੍ਰੋਗਰਾਮਾਂ ਵਿੱਚ ਵਰਤਿਆ ਜਾ ਸਕਦਾ ਹੈ।

6. ਅਸੀਂ ਆਪਣਾ ਸਮਾਂ ਅਤੇ ਰੁਪਇਆ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਕਰ ਕੇ ਬਚਾ ਸਕਦੇ ਹਾਂ।

ਯਾਦ ਰੱਖਣ ਯੋਗ ਗੱਲਾਂ

- ਸੀ (C) ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਇੱਕ ਮੁੱਖ ਫੰਕਸ਼ਨ ਹੁੰਦਾ ਹੈ ਅਤੇ ਬਾਕੀ ਸਾਰੇ ਫੰਕਸ਼ਨ ਮੁਖ ਫੰਕਸ਼ਨ ਦੇ ਸਹਾਇਕ ਹੁੰਦੇ ਹਨ।
- ਆਮ ਤੌਰ ਤੇ ਫੰਕਸ਼ਨ ਦੋ ਤਰ੍ਹਾਂ ਦੇ ਹੁੰਦੇ ਹਨ - ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਅਤੇ ਯੂਜ਼ਰ ਡਿਫਾਇਨ ਫੰਕਸ਼ਨ।
- ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਵਰਤਨ ਲਈ ਉਸ ਨਾਲ਼ ਸੰਬੰਧਿਤ ਲਾਇਬਰੇਰੀ ਫਾਇਲ ਨੂੰ ਪ੍ਰੋਗਰਾਮ ਵਿੱਚ ਸ਼ਾਮਿਲ ਕਰਨਾ ਪੈਂਦਾ ਹੈ।
- ਸੀ (C) ਭਾਸ਼ਾ ਵਿੱਚ ਵੱਖ-ਵੱਖ ਪ੍ਰੋਗਰਾਮ ਇੱਕ ਸਾਂਝਾ ਫੰਕਸ਼ਨ ਵਰਤ ਸਰਦੇ ਹਨ।
- ਫੰਕਸ਼ਨ ਦੀ ਸਹਾਇਤਾ ਨਾਲ਼ ਅਸੀਂ ਗੁੰਝਲਦਾਰ ਪ੍ਰੋਗਰਾਮਾਂ ਨੂੰ ਸੌਖਿਆਂ ਹੀ ਲਿਖ ਸਕਦੇ ਹਾਂ।

ਅਭਿਆਸ

ਪ੍ਰ 1. - ਖਾਲੀ ਸਥਾਨ ਭਰੋ :-

1 .... ਸਟੇਟਮੈਂਟਾਂ ਦਾ ਇੱਕ ਸਮੂਹ ਹੈ ਜੋ ਕਿ ਇੱਕ ਖਾਸ ਕੰਮ ਕਰਦਾ ਹੈ।

2 ਆਮ ਤੌਰ ਤੇ ਫੰਕਸ਼ਨਾਂ ਨੂੰ....................ਭਾਗਾਂ ਵਿੱਚ ਵੰਡਿਆ ਜਾ ਸਕਦਾ ਹੈ।

3 .... ਫੰਕਸ਼ਨ ਆਰਗੂਮੈਂਟ ਵਿੱਚ ਸਥਿਤ ਕਰੈਕਟਰ ਦੀ ਕਿਸਮ ਨੂੰ ਲੋਅਰ ਕੇਸ ਵਿੱਚ ਬਦਲਦਾ ਹੈ।

4 ਸਟੈਂਡਰਡ ਇਨਪੁੱਟ ਆਉਟਪੁੱਟ ਫੰਕਸ਼ਨ .... ਹੈਡਰ ਫਾਈਲ ਵਿੱਚ ਹੁੰਦੇ ਹਨ।

5 ਸਾਰੇ ਮੌਜੂਦ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਇੱਕ ....ਫਾਈਲ ਵਿੱਚ ਹੁੰਦੇ ਹਨ।

ਪ੍ਰ. 2 ਮਿਲਾਣ ਕਰੋ

ਏ	ਬੀ
Sqrt( )	ਭਾਗਫਲ ਦੇ ਬਾਕੀ ਦੀ ਗਣਨਾ ਕਰਨਾ
toupper( )	ਨਿਰਪੇਖ ਮੁੱਲ ਪਤਾ ਕਰਨਾ
Fab( )	ਕਰੈਕਟਰ ਨੂੰ ਅਪਰ ਕੇਸ ਵਿੱਚ ਬਦਲਣਾ।
fmod( )	ਵਰਗਮੂਲ ਪਤਾ ਕਰਨ ਲਈ
Strcmp( )	ਦੋ ਸਟਰਿੰਗਾ ਦਾ ਮੁਕਾਬਲਾਂ ਕਰਨ ਲਈ

ਪ੍ਰ. 3 ਛੋਟੇ ਉੱਤਰਾਂ ਵਾਲੇ ਪ੍ਰਸਨ

1 ਫੰਕਸ਼ਨਾਂ ਦਾ ਵਰਗੀਕਰਨ ਕਿੰਨੇ ਭਾਗਾਂ ਵਿੱਚ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ ?

2 ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਦੀ ਵਿਆਖਿਆ ਕਰੋ ?

3 ਪ੍ਰੋਗਰਾਮ ਦੀ ਸ਼ੁਰੂਆਤ ਵਿੱਚ ਕਿਹੜੀਆਂ ਫਾਈਲਾਂ ਸ਼ਾਮਿਲ ਕੀਤੀਅਸ਼ ਜਾਂਦੀਆਂ ਹਨ ਅਤੇ ਕਿਉਂ ?

4 ਯੂਜ਼ਰ ਡਿਫਾਈਨ ਅਤੇ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਵਿੱਚ ਅੰਤਰ ਦੱਸੋ ?

5 Stringh ਫਾਈਲ ਵਿੱਚ ਮੌਜੂਦ ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨਾਂ ਦੇ ਨਾਂ ਦੱਸੋ ?

6 ਸੀ C ਭਾਸਾ ਵਿੱਚ ਵਰਤਿਆਂ ਜਾਣ ਵਾਲੀਆਂ ਕਿਸੇ ਚਾਰ ਹੈਡਰ ਫਾਈਲਾਂ ਦੇ ਨਾਂ ਲਿਖੋ ?

7 ਸਟਰਿੰਗ ਕਾਪੀ ਫੰਕਸ਼ਨ ਦੀ ਉਦਾਹਰਣ ਸਹਿਤ ਵਿਆਖਿਆ ਕਰੋ ?

8 ਟੂਲੋਅਰ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ ?

9 Strcat ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ ?

ਪ੍ਰ. 4 ਵੱਡੇ ਉੱਤਰਾਂ ਵਾਲੇ ਪ੍ਰਸਨ

1 ਫੰਕਸ਼ਨਾਂ ਦੇ ਮੁੱਖ ਲਾਭ ਕੀ ਹਨ ?

2 ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨਾਂ ਦੀ ਵਿਆਖਿਆ ਕਰੋ ਕਿਸੇ ਚਾਰ ਦੀ ਵਿਸਥਾਰ ਪੂਰਵਕ ਵਿਆਖਿਆ ਕਰੋ ?

3 ਲਾਇਬਰੇਰੀ ਫੰਕਸ਼ਨ ਦੀ ਵਰਤੋਂ ਕਰਕੇ ਕਿਸੇ ਅੰਕ ਦਾ ਵਰਗਮੂਲ ਪਤਾ ਕਰਨ ਲਈ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ ?

4 ਇੱਕ ਅਜਿਹਾ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ ਜੋ ਕਿਸੇ ਕਰੈਕਟਰ ਨੂੰ ਅਪਰ ਕੇਸ ਵਿੱਚ ਬਦਲ ਦੇਵੇ ?

5 ਇੱਕ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ ਜੋ ਸਟਰਿੰਗ Satik ਦੀ ਲੰਬਾਈ ਦੀ ਗਣਨਾ ਕਰੇ ?

6 ਹੇਠਾਂ ਦਿੱਤੀ ਸੀਰੀਜ ਦਾ ਜੋੜ ਪਤਾ ਕਰਨ ਲਈ ਪ੍ਰੋਗਰਾਮ ਲਿਖੋ–

$x^2 + x^4 + x^6 + \ldots\ldots\ldots\ldots + x^n$

****

ਅਧਿਆਇ

8

ਈ - ਗਵਰਨੈਂਸ
e - Governance

ਈ-ਗਵਰਨੈਂਸ ਇੰਟਰਨੈਟ ਅਤੇ ਵਲਡ-ਵਾਈਡ-ਵੈਬੱ ਨਾਲ ਸਬੰਧਤ ਇਕ ਤਕਨੀਕ ਹੈ ਜਿਸਦੇ ਨਾਲ ਅਸੀਂ ਰੋਜ਼ਾਨਾ ਦੇ ਲਈ ਜ਼ਰੂਰੀ ਸੂਚਨਾਵਾਂ ਨੂੰ ਕਿਤੇ ਵੀ ਤੇ ਕਦੇ ਵੀ ਦੇਖ ਸਕਦੇ ਹਾਂ ਅਤੇ ਇਸਤੇਮਾਲ ਵੀ ਕਰ ਸਕਦੇ ਹਾਂ। ਇਸ ਦੇ ਲਈ ਸਾਡੇ ਕੋਲ ਕੰਪਿਊਟਰ ਅਤੇ ਇੰਟਰਨੈੱਟ ਹੋਣਾ ਜ਼ਰੂਰੀ ਹੈ। ਇਹ ਇਕ ਗਲੋਬਲ (global) ਪ੍ਰਕ੍ਰਿਆ ਹੈ। ਇਸ ਦੇ ਨਾਲ ਸੂਚਨਾਵਾਂ ਨੂੰ ਆਪਸ ਵਿਚ ਅਦਾਨ ਪ੍ਰਦਾਨ ਵੀ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ। ਬਹੁਤ ਸਾਰੇ ਕੰਮ ਜੋ ਅਸੀਂ ਕਾਗਜਾਂ ਤੇ ਕਰਦੇ ਸੀ ਹੁਣ ਮਾਊਸ ਦੀ ਕਲਿੱਕ ਉੱਤੇ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ।

ਈ-ਗਵਰਨੈਂਸ ਵਿਚ ਨੈਟਵਰਕਿੰਗ ਦੀਆਂ ਤਕਨੀਕਾਂ ਦਾ ਪ੍ਰਯੋਗ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਜਿਸ ਨਾਲ ਅਸੀ ਸੂਚਨਾਵਾਂ ਨੂੰ ਸੁਚੱਜੇ ਢੰਗ ਨਾਲ ਕਿਤੇ ਵੀ ਅਤੇ ਕਿਸੇ ਸਮੇਂ ਵੀ ਪ੍ਰੋਸੈਸ ਕਰ ਸਕਦੇ ਹਾਂ। ਕਿਸੇ ਵੀ ਈ-ਗਵਰਨੈਂਸ ਦਾ ਵਾਤਾਵਰਣ ਨੈਟਵਰਕ ਅਤੇ ਇੰਟਰਨੈਟ ਅਧਾਰਿਤ ਹੋ ਸਕਦਾ ਹੈ। ਇਸ ਵਿਚ ਲਿਖਿਤ, ਮੌਖਿਕ, ਵੀਡੀਓ ਅਤੇ ਐਨੀਮੇਸ਼ਨ ਤਕਨੀਕਾਂ ਸ਼ਾਮਿਲ ਹਨ। ਇਹ ਸਿਖਣ ਦਾ ਤੇ ਸੇਵਾਵਾਂ ਪ੍ਰਦਾਨ ਕਰਨ ਦਾ ਬਹੁਤ ਵਧੀਆ ਢੰਗ ਹੈ।

ਭਾਰਤ ਵਿਚ ਈ-ਗਵਰਨੈਂਸ ਪਿਛਲੇ ਕੁਝ ਸਾਲਾਂ ਤੋਂ ਬਹੁਤ ਪ੍ਰਸਿਧ ਤੇ ਹਰਮਨ ਪਿਆਰੀ ਹੋਈ ਹੈ। ਸਰਕਾਰ ਦੇ ਬਹੁਤ ਸਾਰੇ ਵਿਭਾਗਾਂ ਨੇ ਈ-ਗਵਰਨੈਂਸ ਨੂੰ ਅਪਣਾਇਆ ਹੈ। ਇਸ ਦਾ ਮੁੱਖ ਕਾਰਨ ਇਹ ਹੈ ਕਿ ਇਸ ਦੇ ਨਾਲ ਜਿਥੇ ਕੰਮ ਵਿਚ ਵਧੇਰੇ ਪਾਰਦਰਸ਼ਿਤਾ ਆਉਂਦੀ ਹੈ ਉੱਥੇ ਲਾਗਤ ਵੀ ਕਈ ਗੁਣਾਂ ਘੱਟ ਜਾਂਦੀ ਹੈ ਤੇ ਸਮੇਂ ਦੀ ਵੀ ਬਹੁਤ ਬਚਤ ਹੁੰਦੀ ਹੈ। ਇਹ ਨਾਗਰਿਕਾਂ ਨੂੰ ਹੋਰ ਵੀ ਬਹੁਤ ਸਾਰੀਆਂ ਸੁਵਿਧਾਵਾਂ ਜਿਵੇਂ distance education ਅਤੇ telemedicine ਆਦਿ ਪ੍ਰਦਾਨ ਕਰਦੀ ਹੈ।

8.1 ਈ-ਗਵਰਨੈਂਸ ਦਾ ਪ੍ਰਯੋਗ (use of e-governance)

ਅੱਜ ਦੇ ਜਮਾਨੇ ਵਿਚ ਈ-ਗਵਰਨੈਂਸ ਦਾ ਪ੍ਰਯੋਗ ਹਰ ਫੀਲਡ ਵਿਚ ਹੋ ਰਿਹਾ ਹੈ ਜਿਵੇਂ ਕਿ ਸਿਖਿਆ ਪ੍ਰਣਾਲੀ, ਵਪਾਰ, ਮੈਡੀਕਲ ਸੁਵਿਧਾਵਾਂ ਆਦਿ। ਅੱਜ ਹਰ ਸ਼ਹਿਰ ਵਿਚ ਈ-ਗਵਰਨੈਂਸ ਸੈਂਟਰ, ਸੁਵਿਧਾ ਕੇਂਦਰ ਅਤੇ ਸੰਪਰਕ ਸੈਂਟਰ ਹਨ। ਇਸ ਤੋਂ ਇਲਾਵਾ ਈ-ਗਵਰਨੈਂਸ ਦੇ ਲਾਭ ਹੇਠਾਂ ਦਰਸਾਏ ਗਏ ਹਨ।8

8.1.1 ਘੱਟ ਖਰਚ (Low cost)

ਈ-ਗਵਰਨੈਂਸ ਵਿਚ ਸਿਰਫ ਸ਼ੁਰੂਆਤ ਦੇ ਖਰਚੇ ਵੱਧ ਹਨ। ਉਸ ਤੋਂ ਇਲਾਵਾ ਜਦੋਂ ਸਿਸਟਮ ਦਾ ਪੂਰਾ ਸੈੱਟ-ਅਪ ਸਥਾਪਿਤ ਹੋ ਜਾਂਦਾ ਹੈ ਫਿਰ ਇਸ ਦੇ ਰੋਜ਼ਾਨਾ ਹੋਣ ਵਾਲੇ ਖ਼ਰਚੇ ਬਹੁਤ ਘੱਟ ਹੁੰਦੇ ਹਨ। ਇਸ ਦੀ ਲਾਗਤ ਵਿਚ ਸਾਫਟਵੇਅਰ, ਨੈਟਵਰਕਿੰਗ ਅਤੇ ਇੰਟਰਨੈਟ ਦੀ ਲਾਗਤ ਮੁੱਖ ਹਨ। ਪਰ ਫਿਰ ਵੀ ਇਸ ਦਾ ਖ਼ਰਚ ਪ੍ਰੰਪਰਾਗਤ ਸਾਧਨਾਂ ਦੇ ਖਰਚ ਤੋਂ ਬਹੁਤ ਘੱਟ ਹੁੰਦਾ ਹੈ। ਇਸ ਦਾ ਮੁੱਖ ਕਾਰਨ man power ਦੀ ਲਾਗਤ ਦਾ ਘੱਟ ਹੋਣਾ ਹੈ।

8.1.2 ਤੇਜ਼ ਰਫਤਾਰ (fast speed)

ਈ-ਗਵਰਨੈਂਸ ਵਿਚ ਕੰਮ ਪ੍ਰੰਪਰਾਗਤ ਤਰੀਕਿਆਂ ਤੋਂ ਵੱਧ ਤੇਜ਼ੀ ਨਾਲ ਹੁੰਦੇ ਹਨ। ਜਿਵੇਂ ਅਸੀਂ ਕਿਸੇ ਰੇਲ ਦੀ ਟਿਕਟ ਪਹਿਲਾਂ ਸਟੇਸ਼ਨ ਤੇ ਜਾ ਕੇ ਬੁੱਕ ਕਰਵਾਉਂਦੇ ਸੀ ਪਰ ਹੁਣ ਸਭ ਕੁਝ ਰੇਲਵੇ ਦੀ ਵੈੱਬ-ਸਾਈਟ ਦੇ ਉੱਪਰ ਉਪਲਬਧ ਹੈ। ਅਸੀਂ ਇਹ ਸਭ ਕੁਝ ਆਪਣੇ ਕੰਪਿਊਟਰ ਤੋਂ ਘਰ ਬੈਠੇ ਹੀ ਕਰ ਸਕਦੇ ਹਾਂ। ਇਸੇ ਤਰ੍ਹਾਂ ਅਸੀਂ ਨੈਟ ਬੈਂਕਿੰਗ ਦਾ ਵੀ ਇਸਤੇਮਾਲ ਕਰ ਰਹੇ ਹਾਂ।

8.1.3 ਕਿਤੇ ਵੀ ਕਿਸੇ ਵੀ ਟਾਈਮ (Anywhere anytime)

manual system ਵਿਚ ਕੰਮ ਦੇ ਸਮੇਂ ਨਿਰਧਾਰਿਤ ਹੁੰਦੇ ਹਨ ਜਿਵੇਂ ਕਿਸੇ ਬੈਂਕ ਵਿਚ ਜੇ ਪੈਸੇ ਜਮ੍ਹਾਂ ਕਰਵਾਉਣ ਜਾਂ ਨਿਕਲਵਾਉਣ ਲਈ ਪਹਿਲਾਂ ਉਹ ਸਿਰਫ ਬੈਂਕ ਦੇ ਸਮੇਂ ਦੇ ਵਿਚ ਹੀ ਕੀਤਾ ਜਾਂਦਾ ਸੀ, ਪ੍ਰੰਤੂ ਹੁਣ ATM ਮਸ਼ੀਨਾਂ ਦੇ ਆ ਜਾਣ ਨਾਲ ਜਿਨ੍ਹਾਂ ਨੂੰ ਓਪਰੇਟ ਕਰਨ ਲਈ ਕਿਸੇ ਵੀ ਆਦਮੀ ਦੀ ਜਰੂਰਤ ਨਹੀਂ ਹੁੰਦੀ ਅਤੇ 24 ਘੰਟੇ ਖੁਲ੍ਹੇ ਰਹਿੰਦੇ ਹਨ ਅਸੀਂ ਕਿਸੇ ਵੀ ਵੇਲੇ ਪੈਸੇ ਜਮ੍ਹਾਂ ਜਾਂ ਕਢਵਾ ਸਕਦੇ ਹਾਂ। ਇਸ ਤਰ੍ਹਾਂ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਨੂੰ JIT (Just-in-time) ਵੀ ਕਿਹਾ ਜਾਂਦਾ ਹੈ।

8.1.4 ਆਸਾਨ ਪ੍ਰਬੰਧ (Easy management)

ਈ-ਗਵਰਨੈਂਸ ਦੇ ਆਉਣ ਨਾਲ ਪ੍ਰਬੰਧ ਦੀ ਵਿਧੀ ਕਾਫੀ ਆਸਾਨ ਹੋ ਗਈ ਹੈ, ਪ੍ਰਬੰਧ ਦੇ ਕਾਰਜਾਂ ਦੀਆਂ ਜਿੰਨੀਆ ਵੀ ਗੁੰਝਲਾਂ ਸਨ ਉਹ ਈ-ਗਵਰਨੈਂਸ ਦੇ ਆਉਣ ਨਾਲ ਕਾਫੀ ਹੱਦ ਤੱਕ ਦੂਰ ਹੋ ਗਈਆਂ ਹਨ। ਸਰਕਾਰੀ ਕਾਰਜ ਪ੍ਰਣਾਲੀ ਵਿਚ ਬਹੁਤ ਪਾਰਦਰਸ਼ਤਾ ਆਈ ਹੈ।

8.2 ਈ-ਗਵਰਨੈਂਸ ਦੇ ਖੇਤਰ (Area of e-governance)

ਅਜ ਤਕਰੀਬਨ ਹਰ ਖੇਤਰ ਵਿਚ ਈ-ਗਵਰਨੈਂਸ ਨੂੰ ਅਪਨਾਇਆ ਜਾ ਰਿਹਾ ਹੈ। ਸਰਕਾਰ ਵਲੋਂ ਹਰ ਸਾਲ ਬੱਜਟ ਦਾ ਵੱਡਾ ਹਿੱਸਾ ਈ-ਗਵਰਨੈਂਸ ਦੇ ਲਈ ਮਨਜੂਰ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਹੇਠ ਲਿਖਿਤ ਮੁੱਖ ਖੇਤਰਾਂ ਵਿਚ ਈ-ਗਵਰਨੈਂਸ ਦੀਆਂ ਸੇਵਾਵਾਂ ਦਿਤੀਆਂ ਜਾ ਰਹੀਆਂ ਹਨ :

ਟੈਕਸਾਂ ਦੀ ਅਦਾਇਗੀ
ਪਾਣੀ ਅਤੇ ਸੀਵਰੇਜ ਦੇ ਬਿਲਾਂ ਦੀ ਅਦਾਇਗੀ।
ਬਿਜਲੀ ਦੇ ਬਿਲਾਂ ਦੀ ਅਦਾਇਗੀ।
ਟੈਲੀਫੋਨ ਦੇ ਬਿਲਾਂ ਦੀ ਅਦਾਇਗੀ।
ਸਟਿਕਰ / ਪੋਸਟਲ ਚਲਾਨਾਂ ਦੀ ਅਦਾਇਗੀ।
ਹਾਊਸਿੰਗ ਬੋਰਡ ਦੇ ਫਾਰਮਾਂ ਦੀ ਵਿਕਰੀ ਅਤੇ ਪ੍ਰਾਪਤੀ ਲਈ।
ਹਾਊਸਿੰਗ ਬੋਰਡ ਦੀ ਰਿਹਾਇਸ਼ੀ ਇਕਾਈਆਂ ਦੀ ਫੀਸ ਦੀ ਅਦਾਇਗੀ ਲਈ।
ਸਟੈਂਪ ਪੇਪਰਾਂ ਦੀ ਵਿਕਰੀ ਅਤੇ ਕੋਲੈਕਸ਼ਨ ਲਈ।
ਬਸਾਂ ਦੇ ਮਹੀਨਾਵਾਰ ਪਾਸ ਜਾਰੀ ਕਰਨ ਲਈ।
ਸੀਨੀਅਰ ਸਿਟੀਜ਼ਨ ਕਾਰਡ ਜਾਰੀ ਕਰਨ ਲਈ।
ਅੰਗਹੀਣਾਂ ਦੇ ਪ੍ਰਮਾਣ ਪੱਤਰ ਜਾਰੀ ਕਰਨ ਲਈ।
ਜਨਮ ਅਤੇ ਮੌਤ ਦੇ ਸਰਟੀਫਿਕੇਟ ਜਾਰੀ ਕਰਨ ਲਈ।
ਸਰਕਾਰੀ ਮੈਡੀਕਲ ਕਾਲਜਾਂ ਵਿਚ ਆਨ-ਲਾਈਨ, ਮਰੀਜਾਂ ਦੇ ਰਿਕਾਰਡ ਰੱਖਣ ਲਈ।
ਕਿਰਾਏਦਾਰਾਂ ਦੀ ਰਜਿਸਟਰੇਸ਼ਨ।
ਘਰੇਲੂ ਨੌਕਰਾਂ ਦੀ ਰਜਿਸਟਰੇਸ਼ਨ।
ਪਾਸਪੋਰਟ ਜਾਰੀ ਕਰਨ ਅਤੇ ਰਿਕਾਰਡ ਰੱਖਣ ਲਈ।
ਬੁਢਾਪਾ, ਵਿਧਵਾ ਅਤੇ ਅੰਗਹੀਨ ਦੀ ਪੈਂਨਸ਼ਨ ਜਾਰੀ ਕਰਨ ਲਈ।
ਕਮਿਊਨਿਟੀ ਸੈਂਟਰਾਂ ਦੀ ਬੁਕਿੰਗ ਕਰਨ ਲਈ।
ਜਾਤੀ ਅਤੇ ਅਨੁਸੂਚਿਤ ਜਾਤੀਆਂ ਦੇ ਪ੍ਰਮਾਣ ਪੱਤਰ ਜਾਰੀ ਕਰਨ ਲਈ।

8.3 ਈ-ਗਵਰਨੈਂਸ ਦੇ ਅਲੱਗ-ਅਲੱਗ ਖੇਤਰ (Various areas of e-governance)

ਆਮ ਤੌਰ ਤੇ ਈ-ਗਵਰਨੈਂਸ ਦੀ ਅਧੀਨ ਹੇਠ ਲਿਖੇ ਵਿਸ਼ਿਆਂ ਵਿਚ ਵਰਤੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ :

- **ਇੰਟਰਨੈਟ ਬੈਂਕਿੰਗ (Internet Banking)**
- **ਆਨ-ਲਾਈਨ ਰੇਲ ਅਤੇ ਹਵਾਈ ਟਿਕਟਾਂ ਦੀ ਬੁਕਿੰਗ**
- **ਪਾਸਪੋਰਟ ਆਫਿਸ ਕੰਮ-ਕਾਜ**
- **ਆਨ-ਲਾਈਨ ਵਪਾਰ**

8.3.1 ਇੰਟਰਨੈੱਟ ਬੈਂਕਿੰਗ (Internet banking)

ਇੰਟਰਨੈਟ ਬੈਂਕਿੰਗ ਨੇ ਬੈਂਕਿੰਗ ਕਾਰਜ ਪ੍ਰਣਾਲੀ ਦੀ ਰੂਪ ਰੇਖਾ ਹੀ ਬਦਲ ਦਿਤੀ ਹੈ। ਇਹ ਬੈਂਕਿੰਗ ਅਤੇ ਵਿੱਤੀ ਸੰਸਥਾਵਾਂ ਦੇ ਖੇਤਰ ਵਿਚ ਇਕ ਨਵੇਂ ਯੁਗ ਦੀ ਸ਼ੁਰੂਆਤ ਹੈ। ਹਰ ਤਰ੍ਹਾਂ ਦਾ ਕੰਮ ਜੋ ਪਹਿਲਾਂ ਬੈਂਕ ਦੀ ਸ਼ਾਖਾ ਵਿਚ ਜਾ ਕੇ ਹੀ ਹੋ ਸਕਦਾ ਸੀ ਅੱਜ ਉਹ ਮਾਊਸ ਦੇ ਕਲਿੱਕ ਤੇ ਹੀ ਸੰਭਵ ਹੈ। ਇੰਟਰਨੈੱਟ ਬੈਂਕਿੰਗ ਨੂੰ ਹੇਠ ਲਿਖੇ ਸੰਦਰਭ ਵਿਚ ਦੇਖਿਆ ਜਾ ਸਕਦਾ ਹੈ :

- ਕਸਟਮਰ ਨੂੰ 24 ਘੰਟੇ ਉਸ ਦੇ ਅਕਾਊਂਟ ਦੀ ਜਾਣਕਾਰੀ।
- ਇਕ ਤੋਂ ਦੂਜੇ ਖਾਤੇ ਵਿਚ ਅਦਾਇਗੀਆਂ ਜਾਂ ਪੈਸਾ ਟਰਾਂਸਫਰ ਕਰਨਾ।
- ਆਪਣੇ ਡੇਬਿਟ (debit) ਜਾਂ ਕਰੈਡਿਟ (credit) ਕਾਰਡਜ਼ ਨੂੰ ਆੱਨ-ਲਾਈਨ ਆੱਪਰੇਟ ਕਰਨਾ।
- ਬੈਂਕ ਦੀਆਂ ਨਵੀਂ ਸਕੀਮਾਂ ਦੀ ਜਾਣਕਾਰੀ ਇੰਟਰਨੈੱਟ ਤੇ ਦੇਣਾ।
- ਕਸਟਮਰ ਦੇ ਅਕਾਉਂਟਸ ਦੀ ਸਕਿਉਰਟੀ ਮੁਹਈਆ ਕਰਾਉਣਾ।

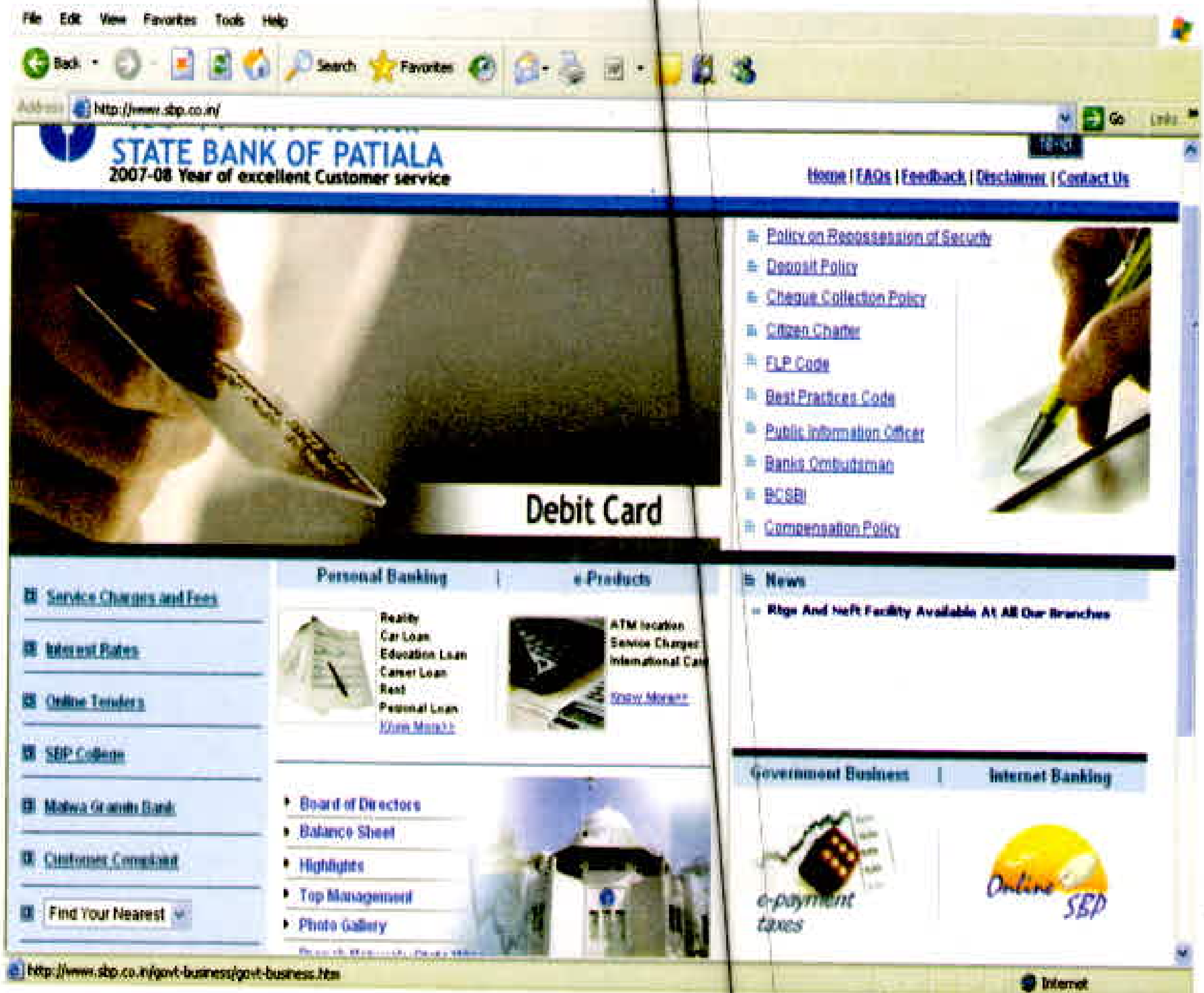
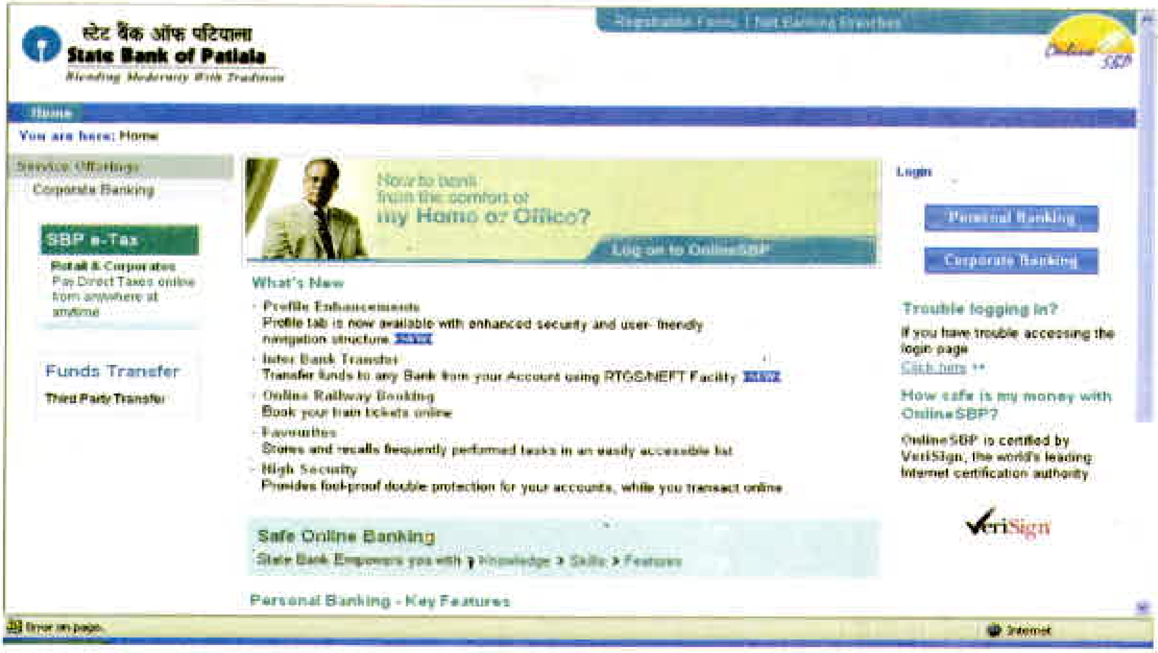


Fig1: Online (Internet Banking) provides several inbuilt features for safe and secure banking

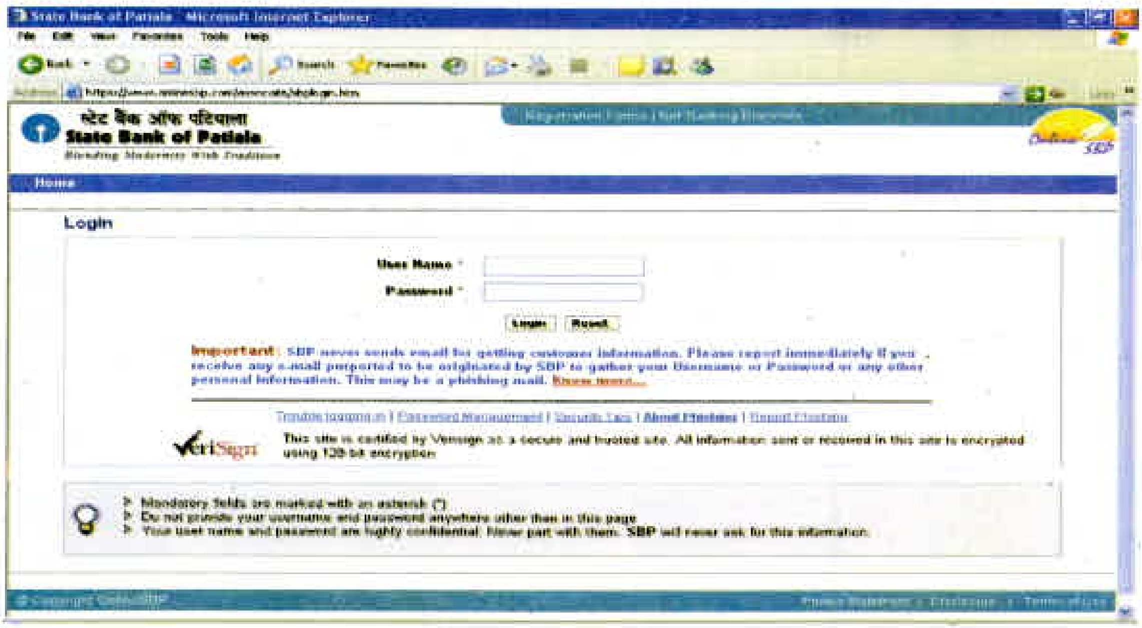
To start this banking process click on the

The following screen opens showing two types of banking as shown :

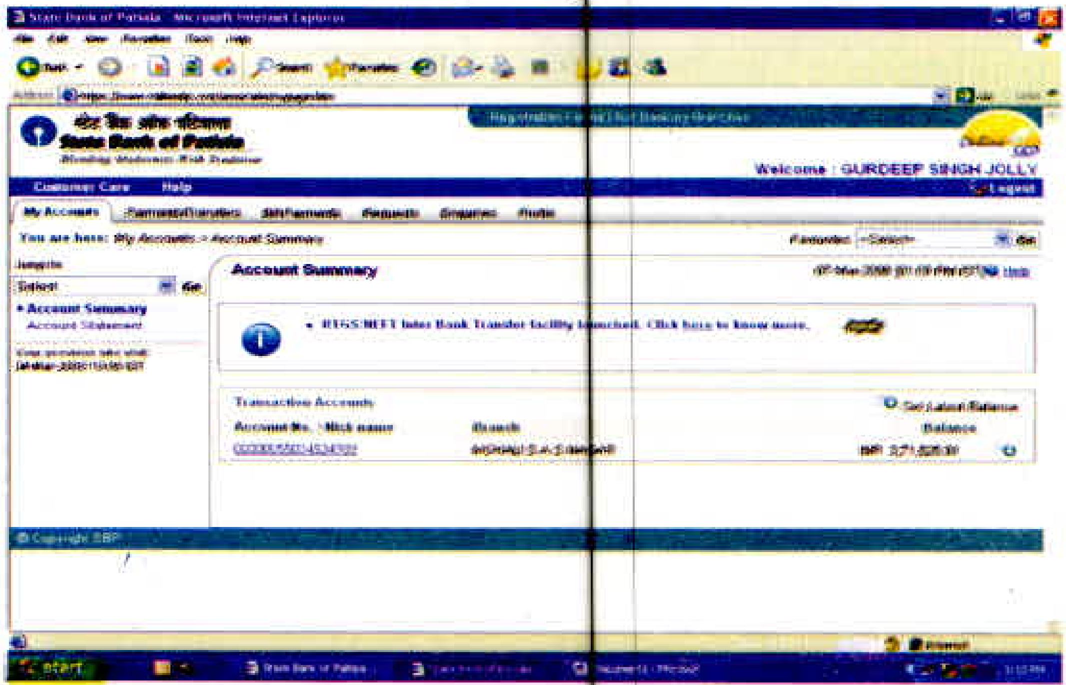


personal Account ਦੇਖਣ ਲਈ personal banking ਆਪਸ਼ਨ ਨੂੰ ਚੁਣੋ।

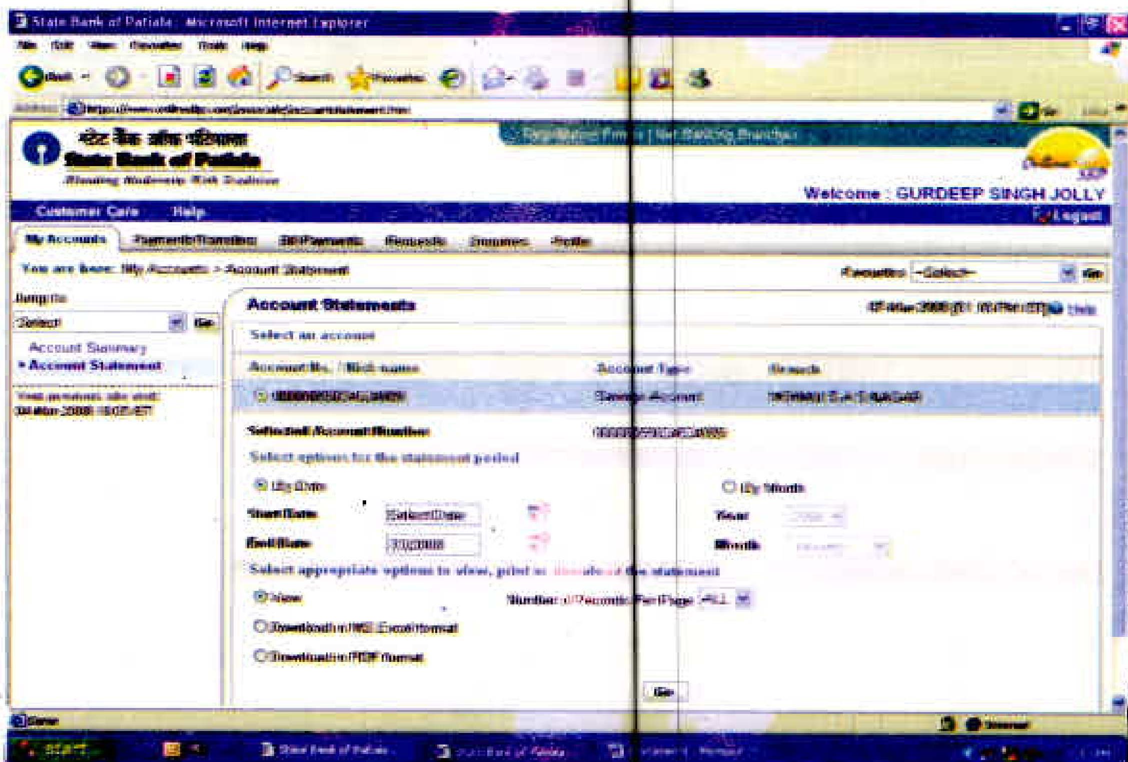
ਇਸ ਔੱਪਸ਼ਨ ਨੂੰ ਚੁਣਨ ਸਮੇਂ ਹੇਠਾਂ ਦਰਸਾਈ ਵਾਂਗ ਸਕਰੀਨ ਦਿਖਾਈ ਦੇਵੇਗੀ। ਕਿਉਂਕਿ ਅਸੀ personal banking ਦੀ ਵਰਤੋਂ ਲਈ ਚੋਣ ਕੀਤੀ ਹੈ ਇਸ ਸਕਰੀਨ ਤੇ ਤੁਸੀ ਆਪਣਾ ਯੂਜ਼ਰ ਨਾਮ ਅਤੇ ਪਾਸਵਰਡ ਭਰੋ।



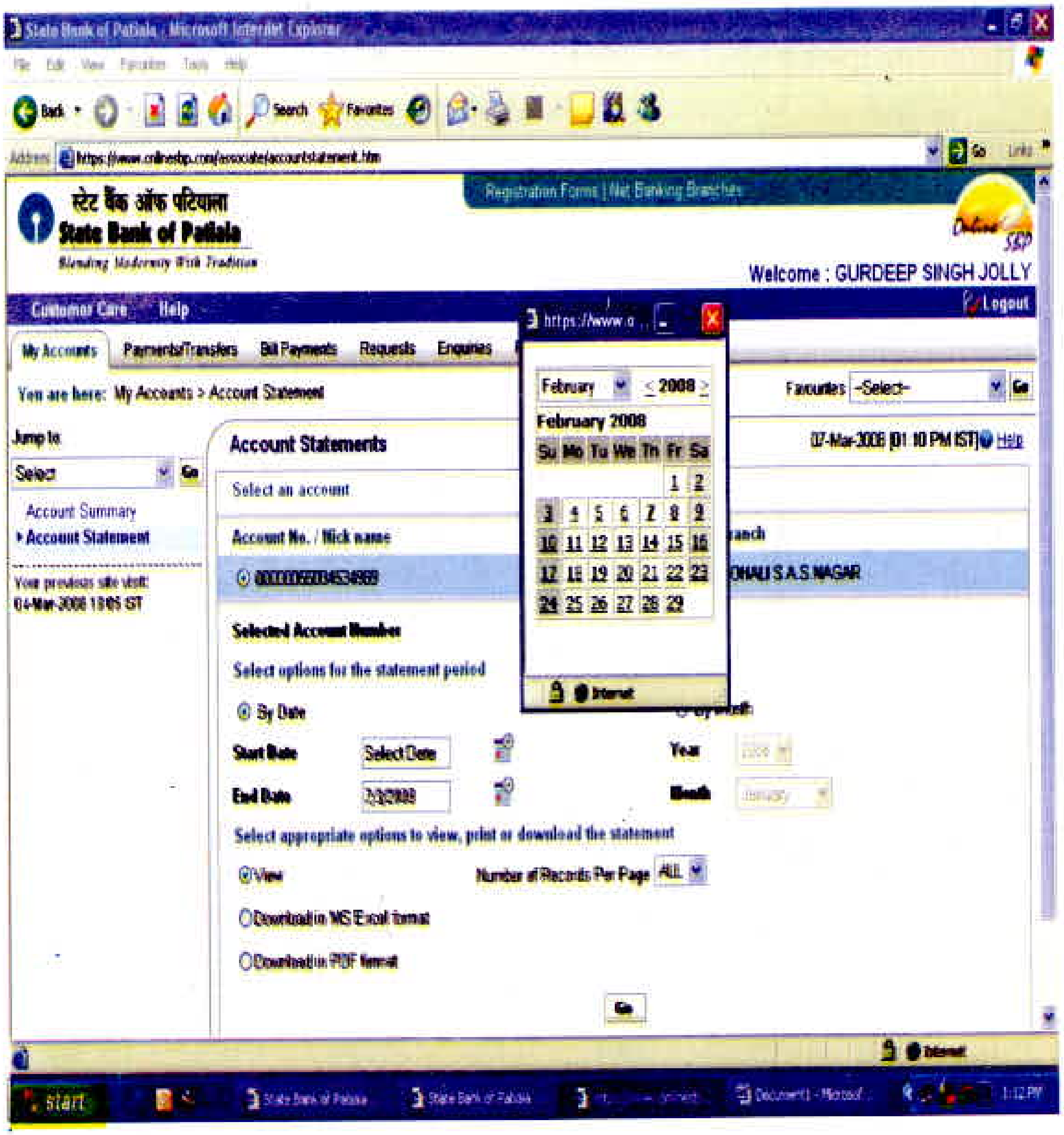
ਸਹੀ ਯੂਜ਼ਰ ਨਾਮ (user name) ਅਤੇ ਪਾਸਵਰਡ (password) ਭਰਨ ਤੇ ਹੇਠਾਂ ਦਰਸਾਈ ਵਾਂਗ ਮੇਨ ਸਕਰੀਨ ਦਿਖਾਈ ਦੇਵੇਗੀ। ਇਸ ਸਕਰੀਨ ਤੇ ਤੁਸੀ ਅਕਾਊਂਟ ਦੀ ਜਾਣਕਾਰੀ ਦੇਖ ਸਕਦੇ ਹੋ।



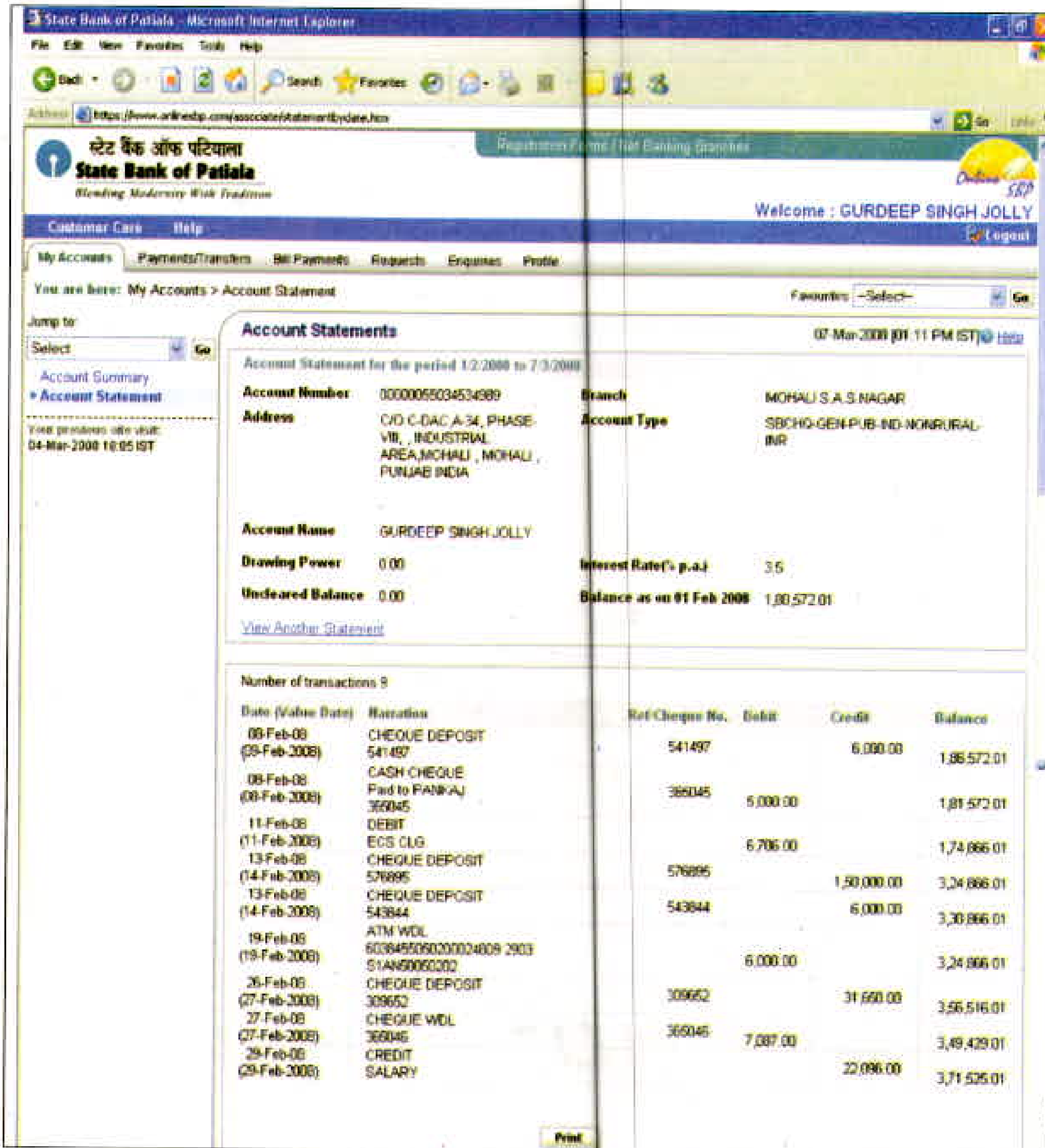
ਹੁਣ ਜੇਕਰ ਉਪਭੋਗਤਾ ਕਿਸੇ ਖਾਸ ਸਮੇਂ ਵਿਚ ਕੋਈ ਸਟੇਟਮੈਂਟ ਨੂੰ ਹੀ ਦੇਖਣਾ ਚਾਹੁੰਦਾ ਹੈ ਤਾਂ By Date Option ਨੂੰ ਚੁਣੋ ਅਤੇ select date ਉੱਪਰ ਕਲਿੱਕ ਕਰੋ। ਹੇਠਾਂ ਦਰਸਾਈ ਵਾਂਗ ਸਕਰੀਨ ਦਿਖਾਈ ਦੇਵੇਗੀ।



ਹੁਣ ਤੁਹਾਨੂੰ ਇਕ ਕੈਲੰਡਰ ਨਜ਼ਰ ਆਵੇਗਾ ਜਿਸ ਦੁਆਰਾ ਤੁਸੀ start date ਅਤੇ end date ਚੁਣ ਸਕਦੇ ਹੋ ਜਿਸ ਤਰ੍ਹਾਂ ਕਿ ਹੇਠਾਂ ਤਸਵੀਰ ਵਿਚ ਦਰਸਾਇਆ ਗਿਆ ਹੈ



ਹੁਣ ਤੁਹਾਡੀ ਚਾਹੁੰਦੀ ਅਕਾਊਂਟ ਸਟੇਟਮੇਂਟ ਸਕਰੀਨ ਤੇ ਦਿਖਾਈ ਦੇਵੇਗੀ ਜਿਸ ਤਰ੍ਹਾਂ ਹੇਠਾਂ ਤਸਵੀਰ ਵਿਚ ਦਰਸਾਇਆ ਗਿਆ ਹੈ



8.3.2 ਆਨ-ਲਾਈਨ ਰੇਲਵੇ ਅਤੇ ਹਵਾਈ ਟਿਕਟਿੰਗ (online railway & air ticketing)

ਅੱਜ ਜਿਥੇ ਹਰ ਚੀਜ਼ ਔਨ-ਲਾਈਨ ਹੋ ਰਹੀ ਹੈ ਉੱਥੇ ਰੇਲਵੇ ਅਤੇ ਹਵਾਈ ਸੰਸਥਾਵਾਂ ਨੇ ਵੀ ਇਸ ਦਿਸ਼ਾ ਵਿਚ ਪਹਿਲ ਕਰਦੇ ਹੋਏ ਆਪਣੇ ਉਪਭੋਗਤਾ (customer) ਨੂੰ ਇੰਟਰਨੈੱਟ ਦੇ ਜਰੀਏ ਹਰ ਸੁਵਿਧਾ ਮੁਹੱਈਆ ਕਰਵਾਈ ਹੈ। ਅਸੀ ਅਜ ਹਰ ਰੇਲਵੇ ਅਤੇ ਹਵਾਈ ਸੰਸਥਾ ਦੀਆਂ ਟਿਕਟਾਂ ਇੰਟਰਨੈਟ ਦੇ ਜ਼ਰੀਏ ਬੁੱਕ ਕਰਵਾ ਸਕਦੇ ਹਾਂ। ਇਥੇ ਅਦਾਇਗੀ ਦੇ ਵੀ ਬਹੁਤ ਵਿਕਲਪ ਮੌਜੂਦ ਹਨ। ਸਾਰੀ ਜਾਣਕਾਰੀ ਔਨ-ਲਾਈਨ ਉਪਲਬਧ ਹੈ। ਸਾਰੀਆਂ ਰੇਲ ਗੱਡੀਆਂ ਤੇ ਹਵਾਈ ਉਡਾਣਾਂ ਦੀ ਜਾਣਕਾਰੀ ਪਹਿਲਾਂ ਤੋ ਹੀ ਪ੍ਰਾਪਤ ਕੀਤੀ ਜਾ ਸਕਦੀ ਹੈ।

ਇਥੇ ਰਿਜ਼ਰਵੇਸ਼ਨ ਆਦਿ ਦੀ ਸੁਵਿਧਾ ਐਡਵਾਂਸ ਵਿਚ ਹੀ ਪਤਾ ਕਰ ਕੇ ਬੁਕਿੰਗ ਕਰਾਈ ਜਾ ਸਕਦੀ ਹੈ। ਇਨ੍ਹਾਂ ਏਜੰਸੀਆਂ ਦਾ ਤਾਲਮੇਲ ਬਹੁਤ ਸਾਰੇ ਬੈਂਕਾਂ ਤੇ ਵਿੱਤੀ ਸੰਸਥਾਵਾਂ ਨਾਲ ਹੁੰਦਾ ਹੈ।

ਭਾਰਤੀ ਰੇਲਵੇ ਨੇ ਦੋ ਤਰ੍ਹਾਂ ਦੀ ਟਿਕਟਿੰਗ ਪ੍ਰਣਾਲੀ ਸ਼ੁਰੂ ਕੀਤੀ ਹੋਈ ਹੈ

- **i –ਟਿਕਟਿੰਗ (i-ticketing)**
- **ਈ –ਟਿਕਟਿੰਗ (e-ticketing)**

i–ਟਿਕਟਿੰਗ (i-ticketing)

ਇਹ ਆਮ ਪ੍ਰਣਾਲੀ ਹੈ ਜਿਸ ਦੇ ਵਿਚ 2 ਜਾਂ 3 ਦਿਨ ਦੇ ਅੰਦਰ ਟਿਕਟ ਘਰ ਪਹੁੰਚਾ ਦਿਤੀ ਜਾਂਦੀ ਹੈ। i–ਟਿਕਟਿੰਗ ਵਿਚ ਕਸਟਮਰ (customer) ਨੂੰ ਘੱਟ ਤੋਂ ਘੱਟ 2 ਦਿਨ ਪਹਿਲਾਂ ਟਿਕਟ ਬੁੱਕ ਕਰਵਾਉਣੀ ਪੈਂਦੀ ਹੈ।

Create an i-Ticketing Account

Use this form to create a free i-Ticketing account for managing your organizations helpdesk issues.

Full name of your organization [Organization Name]

Organization Alias [alias]

Note: Your organization alias will be converted to all lower case.

Admin Email [admin@domain.com]

(eg: "admin@example.com")

Admin Password [ ]

Please enter a secure password.

[Submit]

Note: You must have cookies enabled in your browser to use i-Ticketing

ਈ-ਟਿਕਟਿੰਗ (e-ticketing)

ਈ - ਟਿਕਟਿੰਗ ਗੱਡੀ ਦੇ ਸਮੇਂ ਦੇ ਚਾਰਟ ਦੇ ਹਿਸਾਬ ਨਾਲ ਬੁੱਕ ਕੀਤੀ ਜਾਂਦੀ ਹੈ। ਈ - ਟਿਕਟਿੰਗ ਲਈ ਤੁਹਾਨੂੰ ਆਪਣਾ ਸ਼ਨਾਖਤੀ ਸਬੂਤ ਜਿਵੇਂ ਕਿ PAN ਕਾਰਡ, voter ID ਕਾਰਡ, ਡਰਾਈਵਿੰਗ ਲਾਇਸੈਂਸ, ਪਾਸਪੋਰਟ ਜਾਂ ਰਾਸ਼ਨ ਕਾਰਡ ਆਦਿ ਦਾ ਵੇਰਵਾ ਦੇਣਾ ਪਵੇਗਾ। ਇਸ ਤੋਂ ਬਾਅਦ ਉਸ ਦਾ ਇਕ ਪ੍ਰਿੰਟ ਲਵੋ ਅਤੇ ਜਦੋਂ ਯਾਤਰਾ ਕਰਦੇ ਹੋ ਇਹ ਪ੍ਰਿੰਟ ਅਤੇ ਆਪਣਾ ID ਸਬੂਤ ਨਾਲ ਰਖੋ। ਕਿਰਾਇਆ ਦੇਣ ਲਈ ਬਹੁਤ ਸਾਰੇ ਬੈਂਕਾਂ ਨੇ ਔਨ-ਲਾਈਨ ਸੇਵਾ ਸ਼ੁਰੂ ਕੀਤੀ ਹੋਈ ਹੈ

ਪਾਸਪੋਰਟ ਸੇਵਾਵਾਂ (Passport services)

ਅੱਜਕਲ੍ਹ ਹਰ ਪਾਸਪੋਰਟ ਆਫਿਸ ਵਿਚ ਔਨ-ਲਾਈਨ ਸੇਵਾਵਾਂ ਸ਼ੁਰੂ ਕੀਤੀਆਂ ਹੋਈਆਂ ਹਨ। ਐਪਲੀਕੈਂਟ ਆਪਣੇ ਨਾਮ ਤੇ ਦੂਜੀਆਂ ਡਿਟੇਲਜ਼ (details) ਨਾਲ ਵੈੱਬ ਉਤੇ ਰਜਿਸਟਰ ਹੋ ਸਕਦੇ ਹਨ। ਉਹਨਾ ਨੂੰ ਇਕ ਰੈਫਰੈਂਸ ਨੰਬਰ ਦੇ ਦਿੱਤਾ ਜਾਂਦਾ ਹੈ। ਉਸ ਤੋਂ ਬਾਅਦ 3 ਤੋਂ 7 ਦਿਨਾਂ ਵਿਚ ਉਹ ਆਪਣੇ ਡਾਕੂਮੈਂਟ ਅਤੇ ਫੀਸ ਨਾਲ ਪਾਸਪੋਰਟ ਆਫਿਸ ਵਿਚ ਜਮਾਂ ਕਰਵਾ ਸਕਦੇ ਹਨ।

8.4 ਸੁਵਿਧਾ ਸੈਂਟਰ (Suvidha Centre)

ਅੱਜ ਕਲ ਹਰ ਸ਼ਹਿਰ ਵਿਚ ਇਕ ਖਿੜਕੀ ਉੱਪਰ ਹੀ ਸ਼ਹਿਰ ਦੇ ਨਾਗਰਿਕਾਂ ਨੂੰ ਮੌਲਿਕ ਸੁਵਿਧਾ ਉੱਪਲਬਧ ਕਰਾਈ ਜਾ ਰਹੀ ਹੈ ਜਿਸ ਲਈ ਹਰ ਸ਼ਹਿਰ ਵਿਚ ਸੁਵਿਧਾ ਕੇਂਦਰਾਂ ਦੀ ਸਥਾਪਨਾ ਕੀਤੀ ਜਾ ਰਹੀ ਹੈ। ਹਰ ਜ਼ਿਲ੍ਹੇ ਵਿਚ ਸੁਵਿਧਾ ਕੇਂਦਰ ਬਹੁਤ ਤੇਜ਼ੀ ਨਾਲ ਬਣਾਏ ਜਾ ਰਹੇ ਹਨ। ਹਰ ਸੁਵਿਧਾ ਕੇਂਦਰ ਵਾਤਾ-ਨਕੂਲ ਹੈ ਅਤੇ ਪੂਰਣ ਰੂਪ ਵਿਚ ਕੰਪਿਊਟਰੀਕ੍ਰਿਤ ਹੈ।

ਇਹਨਾਂ ਵਿਚ ਹਰ ਤਰ੍ਹਾਂ ਦੀਆਂ ਮੌਲਿਕ ਸੁਵਿਧਾਵਾਂ ਜਿਵੇਂ ਕਿ ਸਟਾਂਮ-ਪੇਪਰ ਦੀ ਵਿਕਰੀ, ਐਫੀਡੇਵਿਟ ਦੀ ਤਸਦੀਕ, ਜਨਮ ਪ੍ਰਮਾਣ ਪੱਤਰ, ਮੌਤ ਪ੍ਰਮਾਣ ਪੱਤਰ, ਪਾਸਪੋਰਟ ਸੰਬਧੀ ਅਰਜ਼ੀਆਂ ਦਾ ਭੁਗਤਾਨ, ਵਿਆਹ ਦਾ ਰਜਿਸਟਰੀਕਰਣ, ਹਸਤਾਖਰ ਦੀ ਤਸਦੀਕ, ਹਥਿਆਰਾਂ ਦੇ ਪ੍ਰਮਾਣ ਪੱਤਰ, ਡਰਾਇਵਿੰਗ ਲਾਈਸੈਂਸ ਦਾ ਜਾਰੀ ਕਰਨਾ, ਵਾਹਨਾਂ ਦਾ ਪੰਜੀਕਰਨ, ਬਿਜਲੀ ਅਤੇ ਪਾਣੀ ਦੇ ਬਿਲਾਂ ਦਾ ਭੁਗਤਾਨ, ਰੈਵਿਨਿਊ ਰਿਕਾਰਡ ਦੀ ਕਾਪੀ ਲੈਣਾ ਆਦਿ ਉੱਪਲਧ ਹਨ। ਆਮ ਤੌਰ ਤੇ ਹਰ ਕੰਮਕਾਜ ਵਾਲੇ ਦਿਨ ਹਰ ਸੁਵਿਧਾ ਕੇਂਦਰ ਤੇ 1000 ਦੇ ਕਰੀਬ ਵਿਅਕਤੀ ਆਂਉਦੇ ਹਨ। ਹਰ ਸੁਵਿਧਾ ਨੂੰ ਚੰਗੀ ਤਰ੍ਹਾਂ ਨਿਰੀਖਣ ਕਰਨ ਲਈ ਉਪਯੁਕਤ ਸਾਫਟਵੇਅਰ ਉਪੱਲਬਧ ਹਨ। ਪਹਿਲਾਂ ਹਰ ਸੁਵਿਧਾ ਨੂੰ ਪ੍ਰਾਪਤ ਕਰਨ ਲਈ ਆਮ ਜਨਤਾ ਨੂੰ ਕਈ ਵਾਰ ਰਿਸ਼ਵਤ ਦੇਣੀ ਪੈਂਦੀ ਸੀ ਪ੍ਰੰਤੂ ਹੁਣ ਆਨ-ਲਾਈਨ ਹੋ ਜਾਣ ਕਰਕੇ ਇਸ ਵਿਚ ਵੀ ਕਮੀ ਆਈ ਹੈ, ਕਹਿਣ ਦਾ ਮੱਤਲਬ ਹੈ ਪ੍ਰਸ਼ਾਸ਼ਨ ਦੀ ਕਾਰਵਾਈ ਵਿਚ ਵਧੇਰੇ ਪਾਰਦਰਸ਼ਿਤਾ ਆਈ ਹੈ। ਹਾਲਾਂਕਿ ਸਾਡੇ ਕੋਲ 40% ਤੱਕ ਸਟਾਫ ਦੀ ਕਮੀ ਹੈ ਪਰ ਫਿਰ ਵੀ ਸੁਵਿਧਾ ਕੇਂਦਰਾਂ ਕਰਕੇ ਇਕ ਤਾਂ ਸਟਾਫ ਦੀ ਕਮੀ ਪੂਰੀ ਹੋਈ ਹੈ ਦੂਜਾ ਕੰਮ ਵਿਚ ਜਿਹੜੀ ਦੇਰੀ ਹੁੰਦੀ ਸੀ ਉਹ ਵੀ ਕਿਸੇ ਹੱਦ ਤਕ ਦੂਰ ਹੋਏ ਹਨ। ਸੁਵਿਧਾ ਕੇਂਦਰ ਨੇ ਕੋਰਿਅਰ (courier) ਸਰਵਿਸ ਵੀ ਬਹੁਤ ਤੇਜ ਕਰ ਦਿਤੀ ਹੈ ਜਿਸ ਰਾਹੀਂ ਜਰੂਰੀ ਕਾਗਜਾਤ ਹੁਣ 48 ਘੰਟਿਆਂ ਵਿਚ ਵਿਅਕਤੀਆਂ ਦੇ ਘਰ ਪਹੁੰਚਾ ਦਿਤੇ ਜਾਂਦੇ ਹਨ। ਹਰ ਸੁਵਿਧਾ ਕੇਂਦਰ ਵਿਚ ਹੈਲਪ-ਲਾਈਨ ਨੰਬਰ ਮੌਜੂਦ ਹਨ। ਕੋਈ ਵੀ ਵਿਅਕਤੀ ਟੈਲੀਫੋਨ ਕਰਕੇ ਜ਼ਰੂਰੀ ਜਾਣਕਾਰੀ ਪ੍ਰਾਪਤ ਕਰ ਸਕਦਾ ਹੈ। ਕੋਈ ਵੀ ਸ਼ਿਕਾਇਤ ਕਿਸੇ ਵੀ ਤਰ੍ਹਾਂ ਦੀ ਸੁਵਿਧਾ ਕੇਂਦਰ ਵਿਚ ਦਰਜ ਕਰਵਾਈ ਜਾ ਸਕਦੀ ਹੈ।

ਹੁਣ ਤੱਕ ਸੰਬੰਧਤ ਦਸਤਾਵੇਜ਼ ਦੀ ਰਜਿਸਟਰੇਸ਼ਨ ਲਈ ਧਾਂਦਲੀਆਂ ਕੀਤੀਆਂ ਜਾਂਦੀਆਂ ਸਨ, ਪ੍ਰੰਤੂ ਸਭ ਕੁਝ ਕੰਪਿਊਟਰੀਕ੍ਰਿਤ ਹੋਣ ਕਰਕੇ ਇਹਨਾਂ ਦੇ ਮਾਮਲਿਆਂ ਵਿਚ ਵੀ ਕਮੀ ਆਈ ਹੈ। ਵਿਕਰੇਤਾ ਜਾਂ ਵੇਚਣ ਵਾਲੇ ਦੇ ਸਾਰੇ ਫੋਨ ਨੰਬਰ ਉਸੇ ਦਿਨ ਇਕਠੇ ਕਰ ਕੇ ਸਬ-ਰਜ਼ਿਸਟਰਾਰ ਦੁਆਰਾ ਰਜਿਸਟਰਾਰ ਨੂੰ ਭੇਜ ਦਿੱਤੇ ਜਾਂਦੇ ਹਨ ਅਤੇ ਇਹ ਵੀ ਧਿਆਨ ਰੱਖਿਆ ਜਾਂਦਾ ਹੈ ਕਿ ਕਿਤੇ ਕੋਈ ਝਗੜੇ ਵਾਲੀ ਪ੍ਰਾਪਰਟੀ ਤਾਂ ਨਹੀਂ ਹੈ। ਇਸ ਨਾਲ ਰਿਸ਼ਵਤ ਦੇ ਕੇਸਾਂ ਵਿਚ ਬਹੁਤ ਹੱਦ ਤਕ ਕਮੀ ਆਈ ਹੈ ਅਤੇ ਪ੍ਰਸ਼ਾਸ਼ਨ ਵਧੇਰੇ ਪਾਰਦਰਸ਼ੀ ਹੋਇਆ ਹੈ ਅਤੇ ਜੇਕਰ ਕੋਈ ਇਸ ਤਰ੍ਹਾਂ ਦਾ ਕੇਸ ਸਾਹਮਣੇ ਆਉਂਦਾ ਹੈ ਤਾਂ ਸੰਬੰਧਤ ਵਿਅਕਤੀਆਂ ਦੇ ਵਿਰੁੱਧ ਜ਼ਰੂਰੀ ਤੇ ਕਾਨੂੰਨੀ ਕਾਰਵਾਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।ਸੰਬੰਧਤ ਆਫਿਸਾਂ ਵਿਚ ਕਲੋਜ਼-ਸਰਕਟ ਟੈਲੀਵਿਜ਼ਨ ਕੈਮਰੇ ਵੀ ਲਗਾਏ ਗਏ ਹਨ ਤਾਂ ਜੋ ਹਰ ਅਫਸਰ ਦੇ ਕੰਮ-ਕਾਜ ਤੇ ਲਗਾਤਾਰ ਨਜ਼ਰ ਰੱਖੀ ਜਾ ਸਕੇ।

ਇਸ ਤੋਂ ਇਲਾਵਾ ਇਕ ਹੋਰ ਖੇਤਰ ਜਿਸ ਵਿਚ ਵਧੇਰੇ ਮੁਸ਼ਕਿਲਾਂ ਪੇਸ਼ ਆਉਂਦੀਆਂ ਸਨ, ਉਹ ਹੈ ਸਥਾਈ ਪ੍ਰਾਪਰਟੀ ਦੇ ਡਾਕੂਮੈਂਟ ਦੀ ਅਟੈਸਟੇਸ਼ਨ ਦਾ। ਇਸ ਨੂੰ ਵੀ ਸੁਚੱਜੇ ਢੰਗ ਨਾਲ ਚਲਾਉਣ ਲਈ ਸਾਫਟਵੇਅਰ ਵਿਕਸਿਤ ਕੀਤੇ ਗ਼ਏ ਹਨ ਤੇ ਸੁਵਿਧਾ ਕੇਂਦਰਾਂ ਵਿਚ ਲਗਾਏ ਗਏ ਹਨ। ਇਸ ਤਰ੍ਹਾਂ ਦੇ ਸਭ ਕੇਸ 14 ਦਿਨ ਦੇ ਅੰਦਰ ਪਟਵਾਰੀ ਦੁਆਰਾ update ਕੀਤੇ ਜਾਂਦੇ ਹਨ। ਜੇਕਰ ਪਟਵਾਰੀ ਕੇਸ update ਨਹੀਂ ਕਰਦੇ ਤਾਂ ਉਨ੍ਹਾਂ ਤੋਂ ਕਾਰਨ ਪੁਛਿਆ ਜਾਂਦਾ ਹੈ। ਇਸ ਤਰ੍ਹਾਂ ਦੀ ਪ੍ਰੋਸੈਸ ਤਕਰੀਬਨ ਹਰ ਦਫ਼ਤਰ ਵਿਚ ਉਪਲਬਧ ਹੈ।

ਸੁਵਿਧਾ : (Single User – Friendly Window Disposal and Helpline for Applicants)

ਇਹ ਸਿਸਟਮ ਨੈਸ਼ਨਲ ਇਨਫਰਮੇਟਿਕ ਸੈਂਟਰ (National Infomatic Cente -NIC) ਦੁਆਰਾ ਨਾਗਰਿਕਾਂ ਦੀ ਮੁਸ਼ਕਿਲਾਂ ਨੂੰ ਦੂਰ ਕਰਨ ਲਈ ਸ਼ੁਰੂ ਕੀਤਾ ਗਿਆ ਹੈ। ਹਰ ਰਾਜ ਤੇ ਕੇਂਦਰ ਸ਼ਾਸ਼ਿਤ ਪ੍ਰਦੇਸ਼ ਇਸ ਨੂੰ ਅਪਣਾ ਰਿਹਾ ਹੈ। ਸਭ ਤੋਂ ਪਹਿਲਾਂ ਸੁਵਿਧਾ ਕੇਂਦਰ ਡੀ ਸੀ ਆਫਿਸ ਵਿਚ ਲਗਾਏ ਗਏ ਸਨ ਪ੍ਰੰਤੂ ਇਹਨਾਂ ਦੀ ਵਧਦੀ ਲੋਕਪ੍ਰਿਯਤਾ ਨੂੰ ਦੇਖਦੇ ਹੋਏ ਇਨ੍ਹਾਂ ਨੂੰ ਹਰ ਆਫਿਸ ਵਿਚ ਅਪਣਾਇਆ ਜਾ ਰਿਹਾ ਹੈ।

ਸੁਵਿਧਾ ਕੇਂਦਰਾਂ ਤੇ ਹੇਠ ਲਿਖਿਤ ਸੇਵਾਵਾਂ ਉਪਲਬਧ ਹਨ (Feature of the SUVIDHA Centre):

- ਸੇਵਾਵਾਂ ਲਈ ਇਕੋ ਥਾਂ ਉੱਤੇ ਸਾਰੀ ਐਪਲੀਕੇਸ਼ਨਜ਼ ਨੂੰ ਜਮ੍ਹਾਂ ਕਰਵਾਉਣਾ।
- ਸੇਵਾ ਪ੍ਰਦਾਨ ਕਰਨ ਲਈ ਇਕ ਨਿਸ਼ਚਿਤ ਅਵਧੀ ਜਾਂ ਤਾਰੀਖ।
- ਡੀ ਸੀ ਦੁਆਰਾ ਦੇਰੀ ਹੋਏ ਕੇਸਾਂ ਦਾ ਨਿਰੀਖਣ ਕਰਨਾ।
- ਇਕੋ ਜਗ੍ਹਾ ਉਤੇ ਸੇਵਾਵਾਂ ਪ੍ਰਦਾਨ ਕਰਨਾ। ਨਾਗਰਿਕ ਨੂੰ ਬਰਾਂਚਿਜ਼ (branches) ਵਿਚ ਜਾਣ ਦੀ ਜ਼ਰੂਰਤ ਨਹੀ ਹੁੰਦੀ।
- ਐਪਲੀਕੇਸ਼ਨ ਦਾ ਮੌਕੇ ਤੇ ਹੀ ਨਿਰੀਖਣ ਕਰਨਾ ਤਾਂ ਜੋ ਨਾਗਰਿਕਾਂ ਨੂੰ ਬਾਰ ਬਾਰ ਨਾ ਆਉਣਾ ਪਵੇ। ਜੋ ਵੀ ਜ਼ਰੂਰੀ ਦਸਤਾਵੇਜ਼ ਨਾਲ ਲਗਣੇ ਹਨ ਉਹਨਾਂ ਦਾ ਤੁਰੰਤ ਨਿਰੀਖਣ ਕਰਨਾ। ਹੁਣ ਐਪਲੀਕੇਸ਼ਨ ਪੂਰੀ ਨਾ ਹੋਣ ਦੇ ਕਾਰਨ ਰੱਦ ਨਹੀ ਹੋ ਸਕਦੀ।
- ਫੋਟੋ ਨੂੰ ਮੌਕੇ ਤੇ ਹੀ ਖਿੱਚ ਲਿਆ ਜਾਂਦਾ ਹੈ ਇਸ ਤਰ੍ਹਾਂ ਨਾਗਰਿਕ ਦੇ ਸਮੇਂ ਅਤੇ ਪੈਸੇ ਦੀ ਬਚਤ ਹੁੰਦੀ ਹੈ।
- ਜਿੰਨੇ ਵੀ ਪੈਸੇ ਜਮਾਂ ਕਰਵਾਉਣੇ ਹਨ ਉਸੇ ਸੈਂਟਰ ਤੇ ਜਮਾਂ ਹੋ ਜਾਂਦੇ ਹਨ। ਨਾਗਰਿਕ ਨੂੰ ਅਲੱਗ ਤੋਂ ਬੈਂਕ ਜਾਣ ਦੀ ਜ਼ਰੂਰਤ ਨਹੀ ਪੈਂਦੀ।
- ਜੋ ਵੀ ਨਵੀਂ ਸਕੀਮ ਜਾਂ ਤਰੀਕਾ ਹੈ ਉਸ ਦੀ ਜਾਣਕਾਰੀ ਮੌਕੇ ਤੇ ਨਾਲ ਦੇ ਨਾਲ ਹੀ ਉਪਲਬਧ ਹੋ ਜਾਂਦੀ ਹੈ।
- ਸਾਰੇ ਐਪਲੀਕੇਸ਼ਨ ਫਾਰਮ ਉਸੇ ਕਾਉਂਟਰ ਤੇ ਉਪਲਬਧ।
- ਸਾਰੇ ਕੇਂਦਰਾਂ ਨੂੰ ਆਪਿਸ ਵਿਚ ਲਿੰਕ ਕਰਕੇ ਇਨਫਰਮੇਸ਼ਨ ਐਕਸਚੇਂਜ ਕੀਤੀ ਜਾਣੀ ਸੰਭਵ ਹੈ।
- ਤੁਰੰਤ ਹੀ ਸਾਰੀ ਸੇਵਾਵਾਂ ਕਰਨ ਦੀ ਸੁਵਿਧਾ।

Note:
For complete detail of services available at SUVIDHA Centre refer "Appendix –A"

ਸਮੱਸਿਆਵਾਂ

ਪ੍ਰਸ਼ਨ 1. ਹੇਠਾਂ ਦਿਤੇ ਪ੍ਰਸ਼ਨਾਂ ਦੇ ਜਵਾਬ ਦਿਓ ?

1.1 JIT ਤੋਂ ਤੁਸੀਂ ਕੀ ਸਮਝਦੇ ਹੋ ?
1.2 i-ticketing ਕੀ ਹੁੰਦੀ ਹੈ ?
1.3 E-ticketing ਕੀ ਹੁੰਦੀ ਹੈ ?
1.4 e-governance ਦਾ ਕੀ ਮਹੱਤਵ ਹੈ ?
1.5 ਸੁਵਿਧਾ ਸੈਂਟਰ ਦੇ ਕੋਈ 5 ਮੁੱਖ ਉਪਯੋਗਾਂ ਦਾ ਵਰਨਣ ਕਰੋ ?

ਪ੍ਰਸ਼ਨ 2. ਖਾਲੀ ਸਥਾਨ ਭਰੋ ?

2.1 ਸਮਾਜਿਕ ਕਾਰਜਾਂ ਨੂੰ ਕਰਨ ਲਈ ਪ੍ਰਕਿਰਿਆਵਾਂ ਦੇ ਸਮੂਹ ਨੂੰ ____________ ਕਿਹਾ ਜਾਂਦਾ ਹੈ ?
2.2 ਸਰਕਾਰੀ ਕਾਰਜਾਂ ਨੂੰ ਕਰਨ ਲਈ ਸਰਕਾਰ ________ ਦੀ ਪ੍ਰਕਿਰਿਆ ਦਾ ਇਸਤੇਮਾਲ ਕਰਦੀ ਹੈ ?
2.3 ਸੁਵਿਧਾ ਦਾ ਅਰਥ ਹੈ ________________________________ ?
2.4 ਐਫੀਡੇਵਿਟ ਦੀ ਤੁਰੰਤ ਤਸਦੀਕ ____________ ਸੈਂਟਰ ਵਿਚ ਹੋ ਸਕਦੀ ਹੈ ?
2.5 ਵਾਹਨਾਂ ਦੀ ________________ ਸੁਵਿਧਾ ਸੈਂਟਰ ਵਿਚ ਹੋ ਸਕਦੀ ਹੈ ?

ਪ੍ਰਸ਼ਨ 3. ਦੱਸੋ ਸਹੀ ਹੈ ਜਾਂ ਗਲਤ ?

3.1 i-Tickets ਦੀ ਵਰਤੋਂ ਨਾਲ ਕੇਵਲ ਆੱਨ-ਲਾਈਨ ਰੇਲਵੇ ਟਿਕਟ ਬੁਕਿੰਗ ਹੁੰਦੀ ਹੈ ?
3.2 i-Ticketing ਵਿਚ ਟਿਕਟ ਬੁਕਿੰਗ 2 ਦਿਨ ਪਹਿਲਾਂ ਹੋਣੀ ਜਰੂਰੀ ਹੈ ?
3.3 e-Ticketing ਦੀ ਵਰਤੋਂ ਤਤਕਾਲ ਬੁਕਿੰਗ ਲਈ ਹੁੰਦੀ ਹੈ ?
3.4 e-governance ਸੁਵਿਧਾਵਾਂ ਹਾਸਿਲ ਕਰਨ ਲਈ ਕੰਪਿਊਟਰ ਦੀ ਵਰਤੋਂ ਨਹੀਂ ਕੀਤੀ ਜਾਂਦੀ ?
3.5 ਆਨ ਲਾਈਨ ਬੈਂਕਿੰਗ ਦੀ ਮਦਦ ਨਾਲ ਇਕ ਖਾਤੇ ਤੋਂ ਦੂਜੇ ਖਾਤੇ ਵਿਚ ਪੈਸੇ ਭੇਜਣਾ ਸੰਭਵ ਹੈ ?

Answer to objective questions

Q	1	2	3	4	5
Q2	ਈ-ਗਵਰਨੈਂਸ	ਸੁਵਿਧਾ ਸੈਂਟਰ	Single user friendly window disposal & help – line for applicants	ਸੁਵਿਧਾ ਸੈਂਟਰ	ਰਜਿਸਟਰੀ
Q3	True	True	True	False	True

Appendix - A

Suwidha Centre Services

Sr.No	Service Name	Sub Case	Requirements	Time
1.	Registration of Marriage		Application Form, DOB, Ration Card / Fard / Voter Id Card, Sarpanch / Namberdar (Bridge & Groom), Marriage Card, Marriage Photo, Attested Copy of the above Document	1
2.	Identity Card		Application Form, One photograph, Application Verified from the Department where he works.	1
3.	Ration Card	New Ration Card	Two D1 Form, Affidavit, 3 Family Photograph, Neighbors Ration Card Copy Attested	
		Change of Name	Application Form, News Paper Cutting, Affidavit, Ration Card Original, D1 form from Depot.	
		Change of Address	Application Form, Ration Card original, D1 Form Depot, Affidavit, Neighbor Ration Card Copy	
		Adding Name	Application Form, Deletion Certificate, Affidavit	
4.	NOC for Building Plan		Form A, Registry Copy, Ministry Form, Blue Prints Copy	
5.	Marriage-ability Certificate		Application Form, Affidavit from Parents (Joint) Affidavit from Sarpanch/ Lamberdar /MC., Ration Card Copy from applicant., Birth Certificate/ Passport Copy Attested	
6.	Counter Sign		Simple Application with the Original Document which has to be Counter Signed	
7.	Passport		Simple Application form, Demand Draft, Date of Birth Proof, Ration Card,	
8.	Caste Certificate	SC/BC/Rural/OBC	Application, Ration Card Copy, Affidavit, Book page issue to Sarpanch	
9.	Demarcation of Land / Evaluation of Land		Original Fard, Simple Application	
10.	Arms Licence	New Licence	Application, Ration Card/Voter Id Card Attested, Challan Form	
		Entry of Weapon	Original Bill of Weapon, Application Form, Original Licence of Weapon, Weapon	
		Renewal of Licence	Application, Weapon, Challan form	

		Duplicate Licence	Licence Lost (FIR, Application, Challan Form) Licence Damaged (Licence Original, Application, Challan Form)	
		NOC for Import of Weapon	Licence Photo Copy Attested, Application	
		NOC for Sale of Weapon	Application, Licence Photo Copy Attested	
		NOC for Sale/Transfer Death Case	Original Licence of Dead Person, Affidavits from all Varris of dead person, Attested Photo Copy receipt of that place where weapon is deposited, Death Certificate, Attested Photocopy of Applicant's Licence (If he don't have licence then Sale Order can be pass)	
		Extension of Purchase Period	Application, License Photocopy attested	
		Addition of Weapon	Affidavit, Application, Original Licence	
		Extension of Jurisdiction	Application, Reason, Including the proper proof of Bills, Property papers or Business bases bills.	
		Retainer Entry	Date of Birth Proof Attested Copy Required, Ration Card Photo Copy Attested, Licence of Bank	
11.	Permission for Loud Speaker		Simple Application Form	
12.	Non Encumbrance Certificate		Application, Fard, Affidavit	
13.	Ashirwad Scheme		Application, Affidavit (Father/Mother), Attested Ration Card Photo Copy, DOB Proof, SC Certificate Photo Copy Attested, Marriage Card	
14.	Pension	Old Age Pension	Form, Affidavit, Ration Card Attested Copy, Pension form attested from Doctor	
		Widow Pension	Death Certificate Photo Copy, Ration Card Photo Copy, Affidavit, Form	
		Disable Pension	Form, Affidavit, Ration Card Photo Copy, Handicapped Certificate Photo copy Attested	
15.	Attestation of Affidavit		Identity Proof (Ration Card/Voter ID Card/ Driving Licence/ Evidence from Sarpanch/ Namberdar/MC)	
16.	Attestation of Document		Original Documents	
17.	Form Filling		Nil	
18.	Copying		Simple Question Form with Court Fee	
19.	Dependent Certificate		Application, death Certificate, Affidavit, Ration card Copy	

20.	Buss Pass		Attested Copy of disability certificate, three snaps	
21.	Permission for fairs		Simple Application (4 Copy)	
22.	NOC for Petrol Pump		Map, Company letter, Fard, Akshijra, copy of registry, Copy of LOI	
23.	New Supply of Water Connection		Application form, Sarpanch certified form, affidavit, Map	
24.	Concession of Electricity Bill		Old bill, copy of ration card, copy of SC certificate	
25.	Agriculture Accident	In case of death	Application, attested copy of death certificate, attested copy of ration card, affidavit	
		Other case	Ration card Copy, Affidavit	
26.	Driving Licence	Learning Licence	Application Form, Dob Proof, Residence Proof	
		New Driving Licence	Application Form, Learning Licence	
		Renewal Licence	Application Form with Old Driving Licence	
27.		New RC	Form20, Form 21 (Sale Certificate), Form 22, Temp. No, Invoice (Bill), Residence Proof (Attested Copy), Insurance (Attested Copy), Affidavit (Magistrate Attested), Challan Paid (Road Tax), Form 60 or PAN Card (Attested Copy) for Vehicles costing more than Rs. 50,000/-, Form 61 for Tractor, Fard	
		Duplicate RC	Form 26, FIR (For Lost RC), Insurance (Attested Copy), Affidavit (Magistrate Attested), Fee Rs. 100	
		For Transfer (PB32 only)	Form 29&30, Insurance (Attested Copy), Residence Proof of Purchaser (attested Copy), Affidavit by purchaser, affidavit by Seller, Fee Rs. 30 for 2 wheeler, 100 for autos	
28.		Electricity Bill	Electricity Bill	
29.		Railway reservation	Identity Proof of one of the traveler	
30.	Birth & Death Certificate	Birth & Death Certificate	Simple Application Form	
		Correction in Certificate	Application Form, affidavit (sarpanch / Namberdar & applicant), ration card, voter card, bank copy, birth certificate of other child if any, birth certificate of mother /father, passport	
		Adding name	Application, affidavit	

TABLE : ASCII Character Set

To print one, press the ALT key (hold it down) and type the decimal number

Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	00	Null	32	20	Space	64	40	@	96	60	`
1	01	Start of heading	33	21	!	65	41	A	97	61	a
2	02	Start of text	34	22	"	66	42	B	98	62	b
3	03	End of text	35	23	#	67	43	C	99	63	c
4	04	End of transmit	36	24	$	68	44	D	100	64	d
5	05	Enquiry	37	25	%	69	45	E	101	65	e
6	06	Acknowledge	38	26	&	70	46	F	102	66	f
7	07	Audible bell	39	27	'	71	47	G	103	67	g
8	08	Backspace	40	28	(	72	48	H	104	68	h
9	09	Horizontal tab	41	29	)	73	49	I	105	69	i
10	0A	Line feed	42	2A	*	74	4A	J	106	6A	j
11	0B	Vertical tab	43	2B	+	75	4B	K	107	6B	k
12	0C	Form feed	44	2C	,	76	4C	L	108	6C	l
13	0D	Carriage return	45	2D	-	77	4D	M	109	6D	m
14	0E	Shift out	46	2E	.	78	4E	N	110	6E	n
15	0F	Shift in	47	2F	/	79	4F	O	111	6F	o
16	10	Data link escape	48	30	0	80	50	P	112	70	p
17	11	Device control 1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	50	32	2	82	52	R	114	72	r
19	13	Device control 3	51	33	3	83	53	S	115	73	s
20	14	Device control 4	52	34	4	84	54	T	116	74	t
21	15	Neg. acknowledge	53	35	5	85	55	U	117	75	u
22	16	Synchronous idle	54	36	6	86	56	V	118	76	v
23	17	End trans. block	55	37	7	87	57	W	119	77	w
24	18	Cancel	56	38	8	88	58	X	120	78	x
25	19	End of medium	57	39	9	89	59	Y	121	79	y
26	1A	Substitution	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	59	3B	;	91	5B	[	123	7B	{
28	1C	File separator	60	3C	<	92	5C	\	124	7C	|
29	1D	Group separator	61	3D	=	93	5D	]	125	7D	}
30	1E	Record separator	62	3E	>	94	5E	^	126	7E	~
31	1F	Unit separator	63	3F	?	95	5F	_	127	7F	□

ਸ਼ਬਦਾਵਲੀ (Glossary)

ਏ (A)	
Argument (ਆਰਗੁਮੈਂਟ)	ਜਦੋਂ ਇਕ ਫੰਕਸ਼ਨ ਨੂੰ ਬੁਲਾਉਣਾ ਹੋਵੇ ਤਾਂ ਫੰਕਸ਼ਨ ਨੂੰ ਭੇਜੇ ਗਏ ਅਸਲ ਮੁੱਲਾਂ ਨਾਲ ਸੰਬੰਧਿਤ ਹੈ।
Argument Matching (ਆਰਗੁਮੈਂਟ ਮੈਚਿੰਗ)	ਇਸ ਗੱਲ ਦਾ ਫੈਸਲਾ ਕਰਨ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਕਿ ਫੰਕਸ਼ਨਜ਼ ਦੇ ਕਿਹੜੇ ਸੈੱਟ ਨੂੰ ਵਿਸ਼ੇਸ਼ ਨਾਮ ਦਾ ਫੰਕਸ਼ਨ ਤੈਅ ਕਰਨਾ ਹੈ, ਫੰਕਸ਼ਨ ਮੰਗ ਵਿਚ **ਆਰਗੁਮੈਂਟ** ਦਿੰਦੇ ਹੋਏ। ਮੇਲ ਇਕ ਫੰਕਸ਼ਨ ਮੰਗ ਵਿਚ **ਆਰਗੁਮੈਂਟ** ਦਿੰਦੇ ਹਨ।
ASCII	ਅਰਥਮੈਟਿਕ ਕੋਡ ਚਿੰਨ੍ਹਾਂ ਦਾ ਮਿਆਰ ਹੁੰਦੇ ਹਨ। ਸ਼ਾਬਦਿਕ ਤੌਰ ਤੇ ASCII (American standard code) ਸੂਚਨਾ ਅੰਦਰ ਤਬਾਦਲੇ ਲਈ ਮਿਆਰੀ ਕੋਡ ਲਈ ਹੁੰਦਾ ਹੈ। ਇਹ ਇਨਟੀਜ਼ਰਲ ਅਨੁਵਾਦ ਕੋਡ ਲਈ ਬਹੁਤ ਆਮ ਚਿੰਨ੍ਹ ਹੁੰਦਾ ਹੈ।
Assignment (ਅਸਾਈਨਮੈਂਟ)	ਪੂਰਵ-ਮੌਜੂਦਾ ਆਬਜੈਕਟ ਲਈ ਇਕ ਮੁੱਲ ਪ੍ਰਦਾਨ ਕਰਨ ਦੀ ਪ੍ਰਕ੍ਰਿਆ।
Assignment Operator (ਅਸਾਈਨਮੈਂਟ ਓਪਰੇਟਰ)	ਅਸਾਈਨਮੈਂਟ ਕਰਨ ਦਾ ਓਪਰੇਟਰ।
ਬੀ (B)	
Backbone Network (ਬੈਕਬੋਨ ਨੈੱਟਵਰਕ)	ਇਕ ਨੈਟਵਰਕ ਜਿਹੜਾ ਅਨੇਕਾਂ LANs ਨੂੰ ਜੋੜਣ ਲਈ ਬੈਕਬੋਨ ਦਾ ਕੰਮ ਕਰਦਾ ਹੈ।
Bandwidth (ਬੈਂਡਵਿਡਥ)	ਇਕ ਨੈਟਵਰਕ ਵਿਚ ਦਿੱਤੇ ਗਏ ਸਮੇਂ ਵਿਚ ਡਾਟਾ ਦੇ ਟਰਾਂਸਫਰ ਕਰਨ ਦੀ ਮਾਤਰਾ (quantity) ਦੀ ਸ਼ਮਤਾ ਨੂੰ ਕਿਹਾ ਜਾਂਦਾ ਹੈ। ਜਾਂ ਇਕ ਸਿਗਨਲ ਦਾ ਪ੍ਰਸਾਰ ਕਰਨ ਵਾਲੇ ਮਾਧਿਅਮ ਦੀ ਸਮਰਥਾ।
Boolean (ਬੂਲੀਅਨ)	C ਕੀ-ਵਰਡ ਬੂਲੀਅਨ ਡਾਟਾ ਟਾਈਪ ਨੂੰ ਡਿਕਲੇਅਰ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
Bridge (ਬ੍ਰਿਜ)	ਡਿਵਾਈਸ ਜਿਹੜਾ ਦੋ ਨੈੱਟਵਰਕਾਂ ਨੂੰ ਇਕਠਿਆਂ ਜੋੜਦਾ ਹੈ।
ਸੀ (C)	
Cable (ਕੇਬਲ)	ਤਾਂਬੇ ਦੀ ਤਾਰ ਜਾਂ ਸੁਰੱਖਿਅਤ ਕਵਰ ਵਿਚ ਲਪੇਟੀ ਆਪਟੀਕਲ ਫਾਈਬਰ ਦਾ ਸੰਚਾਰ ਮਾਧਿਅਮ।
Call (ਕਾਲ)	ਪ੍ਰੋਗਰਾਮ, ਰੂਟੀਨ ਜਾਂ ਓਪ-ਰੂਟੀਨ ਦੇ ਕੰਟਰੋਲ ਨੂੰ ਬਦਲਣਾ।
Call By Reference (ਹਵਾਲੇ ਨਾਲ ਬੁਲਾਉਣਾ)	ਇਕ ਫੰਕਸ਼ਨ ਵੱਲ ਆਰਗੁਮੈਂਟ, ਅਤੇ ਪੁਆਂਇੰਟਰ ਗੁਜ਼ਾਰਨਾ। ਫਿਰ ਆਰਗੁਮੈਂਟ ਮੁੱਲ ਬਦਲ ਸਕਦਾ ਹੈ।
Call by value (ਮੁੱਲ ਦੁਆਰਾ ਬੁਲਾਉਣਾ)	ਇਕ ਫੰਕਸ਼ਨ ਨੂੰ ਆਰਗੁਮੈਂਟ ਦੀ ਕਾਪੀ ਭੇਜਣੀ। ਫੰਕਸ਼ਨ for ਆਰਗੁਮੈਂਟ ਮੁੱਲ ਨਹੀਂ ਬਦਲ ਸਕਦਾ। C ਦੀ ਵਰਤੋਂ ਮੁੱਲ ਆਰਗੁਮੈਂਟ ਰਾਹੀਂ ਹੁੰਦੀ ਹੈ। ਪਰੰਤੂ ਹਵਾਲੇ ਦੇ ਰਾਹੀਂ ਵੀ ਹੁੰਦੀ ਹੈ
Capture (ਕੈਪਚਰ)	ਇਕ ਫਾਈਲ ਵਿਚ ਆਡਿਓ, ਵੀਡਿਓ ਅਤੇ ਸਥਿਰ ਤਸਵੀਰਾਂ ਨੂੰ ਡਿਜ਼ੀਟਲ ਡਾਟਾ ਵਜੋਂ ਰਿਕਾਰਡ ਕਰਨ ਲਈ।
Capture Device (ਕੈਪਚਰ ਡਿਵਾਈਸ)	ਇਕ ਹਾਰਡਵੇਅਰ ਡਿਵਾਈਸ ਜਿਸਦੀ ਮਦਦ ਨਾਲ ਆਡਿਓ ਅਤੇ ਵੀਡਿਓ ਨੂੰ ਬਾਹਰੀ ਸਰੋਤ (source) ਤੇ ਕੰਪਿਊਟਰ ਵਿਚ ਟਰਾਂਸਫਰ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
Carriage Return Character (ਕੈਰਿਜ ਰਿਟਰਨ ਕਰੈਕਟਰ)	ਕੇਰਿਜ ਰਿਟਰਨ ਕਰੈਕਟਰ, ਇਕ ਚਿੰਨ੍ਹ ਜਿਹੜਾ ਪਰਿਣਾਮ ਧਾਰਾ ਵਿਚ ਸੰਕੇਤ ਦਿੰਦਾ ਹੈ ਕਿ ਪ੍ਰਿੰਟਿੰਗ ਉਸ ਭੌਤਿਕ ਰੇਖਾ ਤੋਂ ਸ਼ੁਰੂ ਹੋਣੀ ਚਾਹੀਦੀ ਹੈ ਜਿਸ ਵਿਚ ਕੈਰਿਜ ਰਿਟਰਨ ਦੇ ਕਰੈਕਟਰ ਵਾਪਰਦੇ ਹਨ। ਕੈਰਿਜ ਵਾਪਸੀ ਇਕ ਚਿੰਨ੍ਹ ਹੈ ਜੋ '\r' ਦੁਆਰਾ C ਭਾਸ਼ਾ ਵਿਚ ਦਿਤਾ ਗਿਆ ਹੈ।

Case Clause (ਕੇਸ ਖੰਡ)	ਇਕ ਸਵਿਚ ਸਟੇਟਮੇਂਟ ਵਿਚ, ਕੇਸ ਪੱਧਰ ਦਾ ਅਨੁਸਰਣ ਕਿਸੇ ਵੀ ਸੰਖਿਆ ਸਟੇਟਮੇਂਟਸ ਦੁਆਰਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
Case Label (ਕੇਸ ਲੇਬਲ)	ਸ਼ਬਦ **ਕੇਸ** ਦਾ ਅਨੁਸਰਣ ਇਕ ਕਾਨਸਟੈਂਟ ਐਕਸਪ੍ਰੇਸ਼ਨ ਅਤੇ ਵਿਸ਼ਰਾਮ ਚਿੰਨ੍ਹ (:) ਦੁਆਰਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਜਦੋਂ ਸਿਲੈਕਟਰ ਵਲੋਂ ਕਾਨਸਟੈਂਟ ਐਕਸਪ੍ਰੇਸ਼ਨ ਦੇ ਮੁੱਲ ਦਾ ਮੁਲਾਂਕਣ ਕੀਤਾ ਜਾਂਦਾ ਹੈ। ਲੇਬਲ ਦੀ ਅਨੁਸਰਣ ਕਰਨ ਵਾਲੇ ਸਟੇਟਮੇਂਟਸ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।
Character (ਕਰੈਕਟਰ)	(1) ਅੱਖਰ, ਡਿਜ਼ਿਟਸ ਜਾਂ ਹੋਰ ਚਿਨ੍ਹ ਜਿਹੜੇ ਡਾਟਾ ਦੇ ਐਕਸਪ੍ਰੇਸ਼ਨ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ। (2) ਇਕ ਜਾਂ ਵਧੇਰੇ ਬਾਈਟਾਂ ਦਾ ਸਾਰ ਜੋ ਇਕ ਹਰੇ ਗ੍ਰਾਫਿਕ ਚਿਨ੍ਹ ਜਾਂ ਕੰਟਰੋਲ ਕੋਡ ਦਾ ਪ੍ਰਗਟਾਵਾ ਕਰਦਾ ਹੈ।
Character Constant (ਕਰੈਕਟਰ ਕਾਂਸਟੈਂਟ)	ਕਾਨਸਟੈਂਟ ਲੇਪ ਚਿੰਨ੍ਹ ਵਿਚ ਨਥੀ ਇਕ ਚਿੰਨ੍ਹ ਜਾਂ ਬਚਾਊ ਸਾਰ।
Character Set (ਕਰੈਕਟਰ ਸੈੱਟ)	1)ਚਿੰਨ੍ਹ ਸੈੱਟ ਭਿੰਨ-ਭਿੰਨ ਚਿੰਨ੍ਹਾਂ ਦਾ ਅਨੰਤ ਸੈੱਟ (finite set) ਜਿਹੜਾ ਦਿਤੇ ਗਏ ਮੰਤਵ ਲਈ ਮੁਕੰਮਲ ਹੁੰਦਾ ਹੈ। ਮਿਸਾਲ ਲਈ, ਚਿੰਨ੍ਹ ਸੈੱਟ ISO ਸਟੈਂਡਰਡ 646 ਵਿਚ, 7 ਬਿਟ ਕੋਡ ਵਾਲੇ ਚਿੰਨ੍ਹ ਸੈੱਟ ਚਿੰਨ੍ਹ ਸੂਚਨਾ ਪ੍ਰਕ੍ਰਿਆ ਅੰਤਰ ਤਬਦੀਲੀ (2) ਇਕ ਪ੍ਰੋਗਰਾਮਿੰਗ ਭਾਸ਼ਾ ਲਈ ਸਮੂਹ ਵੈਧ ਚਿੰਨ੍ਹ ਜਾਂ ਇਕ ਕੰਪਿਊਟਰ ਸਿਸਟਮ ਲਈ (3) ਵਿਸ਼ੇਸ਼ ਕਾਰਣ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਚਿੰਨ੍ਹਾਂ ਦਾ ਇਕ ਸਮੂਹ, ਮਿਸਾਲ ਲਈ, ਚਿੰਨ੍ਹਾਂ ਦਾ ਇਕ ਸੈੱਟ ਜੋ ਪ੍ਰਿੰਟਰ ਛਾਪ ਸਕਦਾ ਹੈ।
Character String (ਕਰੈਕਟਰ ਸਟਰਿੰਗ)	ਢਤਮ ਕੀਤੇ ਗਏ ਚਿੰਨ੍ਹਾਂ ਦਾ ਨਿਰੰਤਰ ਸਾਰ ਜਿਸ ਵਿਚ ਪਹਿਲਾਂ ਵਿਅਰਥ ਬਾਈਟ ਸ਼ਾਮਲ ਹੈ
Character Variable (ਕਰੈਕਟਰ ਵੇਰਿਏਬਲ)	ਚਿੰਨ੍ਹ ਡਾਟਾ ਮੱਦ ਦਾ ਨਾਮ ਜਿਸ ਦਾ ਮੁੱਲ ਅਸਾਈਨਡ ਹੋਵੇ ਜਾਂ ਪ੍ਰੋਗਰਾਮ ਲਾਗੂ ਕਰਨ ਸਮੇਂ ਬਦਲਿਆ ਗਿਆ ਹੋਵੇ।
Client/Server (ਗ੍ਰਾਹਕ/ਸਰਵਰ)	ਇਕ ਨੈੱਟਵਰਕ ਸਿਸਟਮ ਜਿਸ ਵਿਚ ਇਕ ਜਾਂ ਵਧੇਰੇ ਫਾਈਲ ਸਰਵਰ ਉਪਲਬਧ ਹੋਣ ਜੋ ਸੇਵਾਵਾਂ ਪ੍ਰਦਾਨ ਕਰਦੇ ਹੋਣ। ਜਿਵੇਂ ਕਿ ਨੈੱਟਵਰਕ ਪ੍ਰਬੰਧ, ਪ੍ਰਯੋਗ ਅਤੇ ਕੇਂਦਰੀ ਡਾਟਾ ਸਟੋਰੇਜ਼ ਵਰਕਸਟੇਸ਼ਨ (ਗ੍ਰਾਹਕਾਂ ਲਈ)
Clip (ਕਲਿੱਪ)	ਵੱਡੀ ਵੀਡਿਓ ਫਾਈਲ ਦਾ ਛੋਟਾ ਅੰਸ਼।
Collection (ਕੋਲੈਕਸ਼ਨ)	ਕਲਿਪਸ ਦੀ ਸਾਂਭ ਸੰਭਾਲ ਕਰਨ ਲਈ ਮੈਮਰੀ ਸਪੇਸ ਜਿਵੇਂ ਕਿ ਇਕ ਫੋਲਡਰ।
Compression (ਕੰਪ੍ਰੈਸ਼ਨ)	ਇਕ ਪ੍ਰਕ੍ਰਿਆ ਜਿਸ ਦੁਆਰਾ ਡਿਜ਼ੀਟਲ ਮੀਡੀਆ ਫਾਈਲ ਵਿਚੋਂ ਉਸ ਦਾ ਆਕਾਰ ਘਟਾਉਣ ਲਈ ਫਾਲਤੂ ਡਾਟਾ ਨੂੰ ਕਢਿਆ ਜਾਂਦਾ ਹੈ।
Contents (ਕਾਂਟੈਨਟਸ)	ਡਿਜ਼ੀਟਲ ਮੀਡਿਆ ਫਾਈਲ ਵਿਚ ਰੱਖੀ information ਜਿਵੇਂ ਕੀ ਆਡੀਓ, ਤਸਵੀਰ ਜਾਂ ਟੈਕਸਟ (text)।
Comma Expression (ਵਿਸ਼ਰਾਮ ਚਿੰਨ੍ਹ ਐਕਸਪ੍ਰੇਸ਼ਨ)	ਇਕ ਐਕਸਪ੍ਰੇਸ਼ਨ ਜਿਸ ਵਿਚ ਇਕ ਕੌਮੇ ਜਾਂ ਵਿਸ਼ਰਾਮ ਚਿੰਨ੍ਹ ਦੁਆਰਾ ਵੱਖ ਕੀਤੇ ਗਏ ਦੋ ਉਪਰੈਂਡਸ ਹੁੰਦੇ ਹੋਣ। ਭਾਵੇਂ ਕੰਪਾਈਲਰ ਐਕਸਪ੍ਰੇਸ਼ਨ ਦੁਆਰਾ ਦੋਹਾਂ ਓਪਰੈਂਡਾਂ ਦਾ ਮੁਲਾਂਕਣ ਕੀਤਾ ਜਾਂਦਾ ਹੈ, ਐਕਸਪ੍ਰੇਸ਼ਨ ਦਾ ਮੁੱਲ ਸਹੀ ਓਪਰੈਂਡ ਦਾ ਮੁੱਲ ਹੁੰਦਾ ਹੈ। ਜੇਕਰ ਖੱਬਾ ਉਪਰੈਂਡ ਮੁੱਲ ਉਤਪਾਦਨ ਕਰਦਾ ਹੈ ਤਾਂ ਕੰਪਾਈਲਰ ਇਸ ਮੁੱਲ ਨੂੰ ਨਕਾਰਦਾ ਹੈ। ਵਿਸ਼ੇਸ਼ ਕਰਕੇ ਖੱਬਾ ਉਪਰੈਂਡ ਇਕ ਕੌਮੇ ਦਾ ਐਕਸਪ੍ਰੇਸ਼ਨ ਬਿਆਨ ਕਰਨ ਲਈ ਹੋਰ ਗੈਰ-ਪ੍ਰਭਾਵ ਪੈਦਾ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
Compile (ਕੰਪਾਈਲ)	ਸੋਰਸ ਕੋਡ ਤੋਂ ਆਬਜੈਕਟ ਰੂਪ ਵਿਚ ਰੂਪਾਂਤਰ ਕਰਨਾ।
Compiler Options (ਕੰਪਾਈਲਰ ਆਪਸ਼ਨ)	ਕੀ-ਵਰਡ ਜਿਹੜੇ ਕੰਪਾਈਲੇਸ਼ਨ ਦੇ ਕੁਝ ਵਿਸ਼ੇਸ਼ ਪਹਿਲੂਆਂ ਤੇ ਕੰਟਰੋਲ ਕਰਨ ਲਈ ਨਿਸ਼ਚਿਤ ਕੀਤੇ ਜਾ ਸਕਦੇ ਹਨ।
Composite (ਕੰਪੋਜ਼ਿਟ)	ਇਕਹਿਰੇ ਫਰੇਮ ਜਾਂ ਪ੍ਰਦਰਸ਼ਨ ਵਿਚ, ਦੋ ਜਾਂ ਵਧੇਰੇ ਫਿਲਮ, ਵੀਡੀਓ ਜਾਂ ਇਲੈਕਟ੍ਰਾਨਿਕ ਪ੍ਰਤਿਬਿੰਬਾਂ ਦਾ ਸੁਮੇਲ।
Condition (ਕੰਡੀਸ਼ਨ)	(1) ਇਕ ਸੰਪਰਕੀ ਐਕਸਪ੍ਰੇਸ਼ਨ ਜਿਹੜਾ ਸਹੀ ਜਾਂ ਗਲਤ ਦੇ ਮੁੱਲ ਦਾ ਮੁਲਾਂਕਣ ਕਰਦਾ ਹੈ। (2) ਇਕ ਅਪਵਾਦ ਜਿਹੜਾ ਭਾਸ਼ਾ ਵਾਤਾਵਰਣ ਦੁਆਰਾ ਪਛਾਣਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਇਸ ਪ੍ਰਕਾਰ ਯੂਜ਼ਰ ਨੂੰ ਸਰਗਰਮ ਕਰਦਾ ਹੈ ਅਤੇ ਭਾਸ਼ਾ ਕੰਡੀਸ਼ਨਜ਼ ਦੇ ਹੈਂਡਲਰ ਨੂੰ ਯੋਗ ਬਣਾਉਂਦਾ ਹੈ। ਆਮ ਪ੍ਰੋਗਰਾਮ ਲਈ ਕੋਈ ਵੀ ਬਦਲਾਵ ਇਕ ਪ੍ਰਯੋਗ ਹੁੰਦਾ ਹੈ। ਸ਼ਰਤਾਂ ਹਾਰਡਵੇਅਰ ਜਾਂ ਸੰਚਾਲਿਤ ਸਿਸਟਮ ਦੁਆਰਾ ਫੜ੍ਹੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ਅਤੇ ਪਰਿਣਾਮ ਹੁੰਦਾ ਹੈ ਰੁਕਾਵਟ। ਉਹ ਭਾਸ਼ਾ-ਵਿਸ਼ੇਸ਼ ਪੈਦਾ ਹੋਏ ਕੋਡ ਜਾਂ ਭਾਸ਼ਾ ਲਾਇਬ੍ਰੇਰੀ ਕੋਡ ਦੁਆਰਾ ਵੀ ਫੜ੍ਹੇ ਜਾ ਸਕਦੇ ਹਨ।

Conditional Compilation Directive (ਕੰਡੀਸ਼ਨਲ ਕੰਪਾਈਲੇਸ਼ਨ ਡਾਇਰੈਕਟਿਵ)	ਪ੍ਰੀ-ਪ੍ਰੋਸੈਸਰ ਨਿਰਦੇਸ ਦਿੰਦਾ ਹੈ ਕਿ ਇਕ ਨਿਸ਼ਚਤ ਸਥਿਤੀ ਕੰਪਾਈਲੇਸ਼ਨ ਡਾਇਰੈਕਟਿਵ ਦੇ ਮੁਲਾਂਕਣ ਤੇ ਨਿਰਭਰ ਕਰਦੇ ਹੋਏ ਫਾਈਲ ਵਿਚ (ਕੰਡੀਸ਼ਨਲ ਕੰਪਾਈਲੇਸ਼ਨ ਡਾਇਰੈਕਟਿਵ) ਨਿਸ਼ਚਿਤ ਸੋਰਸ ਕੋਡ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਨੂੰ ਹੀ ਪ੍ਰੀ-ਪ੍ਰੋਸੈਸਰ ਕੀਤਾ ਜਾਵੇ।
Conditional Expression (ਕੰਡੀਸ਼ਨਲ ਐਕਸਪ੍ਰੇਸ਼ਨ)	ਇਕ ਸੰਯੁਕਤ ਐਕਸਪ੍ਰੇਸ਼ਨ ਜਿਸ ਵਿਚ ਇਕ ਕੰਡੀਸ਼ਨ ਸ਼ਾਮਲ ਹੁੰਦਾ ਹੈ (ਪਹਿਲਾ ਐਕਸਪ੍ਰੇਸ਼ਨ) ਐਕਸਪ੍ਰੇਸ਼ਨ ਦਾ ਮੁਲਾਂਕਣ ਤਦ ਹੁੰਦਾ ਹੈ ਜੇਕਰ ਕੰਡੀਸ਼ਨ ਦਾ ਗੈਰ ਜ਼ੀਰੋ ਮੁੱਲ ਹੋਵੇ (ਦੂਜਾ ਐਕਸਪ੍ਰੇਸ਼ਨ) ਅਤੇ ਐਕਸਪ੍ਰੇਸ਼ਨ ਦਾ ਮੁਲਾਂਕਣ ਤਦ ਹੁੰਦਾ ਹੈ ਜੇਕਰ ਕੰਡੀਸ਼ਨ ਦਾ ਮੁੱਲ ਜ਼ੀਰੋ (ਤੀਜਾ ਐਕਸਪ੍ਰੇਸ਼ਨ) ਹੋਵੇ।
Const (ਕਾਂਸਟ)	(1) ਇਕ ਡਾਟਾ ਆਬਜੈਕਟ ਦਾ ਗੁਣ ਆਰੋਪਣ ਜਿਹੜਾ ਡਿਕਲੇਅਰ ਕਰਦਾ ਹੈ ਕਿ ਆਬਜੈਕਟ ਕਦੇ ਨਹੀਂ ਬਦਲਿਆ ਜਾ ਸਕਦਾ। (2) ਇਕ ਫੰਕਸ਼ਨ ਦਾ ਗੁਣ ਆਰੋਪਣ ਜਿਹੜਾ ਡਿਕਲੇਅਰ ਕਰਦਾ ਹੈ ਕਿ ਫੰਕਸ਼ਨ ਵਲੋਂ ਇਸਦੀ ਸ਼੍ਰੇਣੀ ਦੇ ਡਾਟਾ ਮੈਂਬਰਾਂ ਵਿਚ ਸੋਧ ਨਹੀਂ ਲਿਆਂਦੀ ਜਾਵੇਗੀ।
Constant (ਕਾਂਸਟੈਂਟ)	(1) ਪ੍ਰੋਗਰਾਮਿੰਗ ਭਾਸ਼ਾ ਵਿਚ, ਇਕ ਭਾਸ਼ਾ ਆਬਜੈਕਟ ਜਿਹੜਾ ਕੇਵਲ ਇਕ ਵਿਸ਼ੇਸ਼ ਮੁੱਲ ਹੀ ਲੈਂਦਾ ਹੈ। (2) ਇਕ ਮੁੱਲ ਦੇ ਨਾਲ ਡਾਟਾ ਮੱਦ ਜੋ ਕਿ ਪ੍ਰੋਗਰਾਮ ਚਲਣ ਦੌਰਾਨ ਬਦਲਿਆ ਨਹੀਂ ਜਾ ਸਕਦਾ।
Constant Expression (ਕਾਂਸਟੈਂਟ ਐਕਸਪ੍ਰੇਸ਼ਨ)	ਇਕ ਐਕਸਪ੍ਰੇਸ਼ਨ ਜਿਸਦਾ ਉਹ ਮੁੱਲ ਹੁੰਦਾ ਹੈ ਜਿਸਦਾ ਫੈਸਲਾ ਕੰਪਾਈਲੇਸ਼ਨ ਦੇ ਦੌਰਾਨ ਕੀਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਜਿਹੜਾ ਪ੍ਰੋਗਰਾਮ ਦੇ ਚਲਣ ਦੌਰਾਨ ਬਦਲਦਾ ਨਹੀਂ।
Control Character (ਕੰਟਰੋਲ ਕਰੈਕਟਰ)	(1) ਇਕ ਚਿੰਨ੍ਹ ਜਿਸਦਾ ਵਾਪਰਣਾ ਇਕ ਵਿਸ਼ੇਸ਼ ਸੰਦਰਭ ਦੇ ਦੌਰਾਨ ਚਿੰਨ੍ਹ ਤੋਂ ਇਲਾਵਾ ਹੋਰ ਚਿੰਨ੍ਹ ਜਿਹੜਾ ਟੈਕਸਟ ਦੀ ਰਿਕਾਰਡਿੰਗ, ਪ੍ਰੋਸੈਸਿੰਗ, ਸੰਚਾਰ ਜਾਂ ਦੁਭਾਸ਼ਾ ਨੂੰ ਪ੍ਰਭਾਵਿਤ ਕਰਦਾ ਹੈ।
Control Statement (ਕੰਟਰੋਲ ਸਟੇਟਮੈਂਟ)	ਇਕ ਭਾਸ਼ਾ ਸਟੇਟਮੈਂਟ ਜਿਹੜਾ ਲਾਗੂ ਕਰਨ ਦੇ ਆਮ ਮਾਰਗ ਨੂੰ ਬਦਲਦਾ ਹੈ।
Cursor Iteration (ਕਰਸਰ ਇਟਰੇਸ਼ਨ)	ਇਕ ਸੰਗ੍ਰਹਿ ਵਿਚ ਕਰਸਰ ਨੂੰ ਅਗਲੇ ਅੰਸ਼ ਤਕ ਉਦੋਂ ਤਕ ਦੁਹਰਾਏ ਜਾਣ ਦੀ ਪ੍ਰਕ੍ਰਿਆ ਹੁੰਦੀ ਹੈ ਜਦੋਂ ਤਕ ਕਿਸੇ ਕੰਡੀਸ਼ਨ ਦੀ ਸੰਤੁਸ਼ਟੀ ਨਹੀਂ ਹੋ ਜਾਂਦੀ।
ਡੀ (D)	
Declaration (ਡਿਕਲੇਅਰੇਸ਼ਨ)	C ਵਜੂਦ ਜਿਹੜੀ ਇਕ ਪ੍ਰੋਗਰਾਮ ਵਿਚ ਇਕ ਜਾਂ ਵਧੇਰੇ ਨਾਮਾਂ ਨੂੰ ਪ੍ਰਵੇਸ਼ ਕਰਦਾ ਹੈ।
Declaration Statement (ਡਿਕਲੇਅਰੇਸ਼ਨ ਸਟੇਟਮੈਂਟ)	ਇਕ ਸਟੇਟਮੈਂਟ ਦੇ ਰੂਪ ਵਿਚ ਡਿਕਲੇਅਰ ਜਿਹੜੀ C ਵਿਚ ਵਰਤੀ ਜਾ ਸਕਦੀ ਜਿਥੇ ਆਮ ਤੌਰ ਤੇ ਸਟੇਟਮੈਂਟ ਵਰਤੇ ਜਾਣਗੇ
Default argument (ਡਿਫਾਲਟ ਆਰਗੂਮੈਂਟ)	ਇਕ ਫੰਕਸ਼ਨ ਲਈ ਆਪਸ਼ਨਲ ਆਰਗੂਮੈਂਟ। ਇਸ ਫੰਕਸ਼ਨ ਡਿਕਲੇਅਰੇਸ਼ਨ ਵਿਚ ਨਿਸ਼ਚਿਤ ਮੁੱਲ ਦੀ ਵਰਤੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਬਸ਼ਰਤੇ ਆਰਗੂਮੈਂਟ ਨਾ ਦਿੱਤੀ ਗਈ ਹੋਵੇ।
Digital Video (ਡਿਜ਼ਿਟਲ ਵੀਡਿਓ)	ਡਿਜ਼ੀਟਲ ਫਾਰਮੇਟ ਵਿਚ ਸਟੋਰ ਕੀਤੀ ਗਈ ਵੀਡਿਓ ਤਸਵੀਰ ਜਾਂ ਸਾਊਂਡ।
ਈ (E)	
e-governance (ਈ-ਗਵਰਨੈਂਸ)	e-governance ਦਾ ਮਤਲੱਬ information technology ਦੀ ਵਰਤੋਂ ਨਾਲ ਕਾਰਜਾਂ ਨੂੰ ਵਧੇਰੇ ਚੰਗੇ ਤਰੀਕੇ ਨਾਲ ਆਮ ਨਾਗਰਿਕਾਂ ਅਤੇ ਵਪਾਰਿਆਂ ਨੂੰ ਮੁੱਹਈਆ ਕਰਵਾਉਣਾ ਹੁੰਦਾ ਹੈ।
Else (ਏਲਸ)	C ਕੀ-ਵਰਡ, if ਸਟੇਟਮੈਂਟ ਦਾ ਭਾਗ
Enum (ਏਨਮ)	C ਕੀ-ਵਰਡ ਦੀ ਵਰਤੋਂ ਗਿਣਤੀ ਦੀ ਡਿਕਲੇਅਰ ਲਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।
e-ticketing (ਈ-ਟਿਕਟਿੰਗ)	ਇਹ ਇਕ ਆਨ-ਲਾਈਨ (online) ਟਿਕਟ ਹੈ ਇਸ ਟਿਕਟ ਨੂੰ e-mail ਰਾਹੀਂ ਭੇਜਿਆ ਜਾ ਸਕਦਾ ਹੈ ਜਾਂ printable online ਤੇ ਪ੍ਰਿੰਟ ਕੀਤਾ ਜਾ ਸਕਦਾ ਹੈ।
Expression (ਐਕਸਪ੍ਰੇਸ਼ਨ)	ਐਕਸਪ੍ਰੇਸ਼ਨ ਕਿਸੇ ਪ੍ਰਕਾਰ ਦੇ ਮੁੱਲ ਪੈਦਾ ਕਰਨ ਲਈ ਕਾਂਸਟੈਂਟ ਵੇਰੀਏਬਲ ਅਤੇ ਸੰਚਾਲਕਾਂ ਦੇ ਸੁਮੇਲ।
Expression Statement (ਐਕਸਪ੍ਰੇਸ਼ਨ ਸਟੇਟਮੈਂਟ)	ਇਕ ਸਟੇਟਮੈਂਟ ਜੋ ਐਕਸਪ੍ਰੇਸ਼ਨ ਹੁੰਦਾ ਹੈ ਜਿਵੇਂ ਕਿ ਫੰਕਸ਼ਨ ਮੰਗ ਜਾਂ ਐਸਾਈਨ

Extern (ਐਕਸਟਰਨ)	ਇਕ C ਕੀ-ਵਰਡ ਜੋ ਬਾਹਰੀ ਨਾਮ ਨੂੰ ਡਿਕਲੇਅਰ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
ਐਫ (F)	
False (ਗਲਤ)	ਬੂਲੀਅਨ ਟਾਈਪ ਲਈ ਮੁੱਲ ਨਿਸ਼ਚਿਤ ਕਰਨ ਵਾਲਾ C ਕੀ-ਵਰਡ ਜੋ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
Frame (ਫਰੇਮ)	ਵੀਡਿਓ ਨੂੰ ਬਨਾਉਣ ਲਈ ਤਰਤੀਬਵਾਰ (sequence) ਤਸਵੀਰਾਂ ।
Frame Rate (ਫਰੇਮ ਰੇਟ)	ਪ੍ਰਤੀ ਸੈਕਿੰਡ ਦਿਖਾਏ ਜਾਣ ਵਾਲੇ ਵੀਡਿਓ ਫਰੇਮ ਦੀ ਗਿਣਤੀ। ਪ੍ਰਤੀ ਸੈਕਿੰਡ ਜਿਆਦਾ ਫਰੇਮ ਦੀ ਸੰਖਿਆ ਇਕ smooth ਵੀਡਿਓ ਨੂੰ ਦਰਸਾਉਂਦੀ ਹੈ।
Function (ਫੰਕਸ਼ਨ)	ਇਕ C ਵਜੂਦ ਜੋ ਕਿ ਸਟੇਟਮੈਂਟਸ ਦਾ ਸਾਰ ਹੁੰਦਾ ਹੈ। ਇਸਦੀ ਆਪਣੀ ਹੀ ਗੁੰਜਾਇਸ਼ ਹੁੰਦੀ ਹੈ। ਆਰਗੂਮੈਂਟ ਮੁੱਲਾਂ ਦੇ ਸੈੱਟ ਨੂੰ ਸਵੀਕਾਰ ਕਰਦਾ ਹੈ ਅਤੇ ਮੁਕੰਮਲ ਕਰਨ ਤੇ ਮੁੱਲ ਮੋੜਦਾ ਹੈ।
ਜੀ (G)	
Gigabyte (ਗੀਗਾਬਾਈਟਸ)	ਸੂਚਨਾ ਦਾ ਇਕ ਬਿਲੀਅਨ ਬਾਈਟਸ/ਇਕ ਹਜ਼ਾਰ ਮੈਗਾਬਾਈਟਸ।
Global Variable (ਗਲੋਬਲ ਵੇਰਿਏਬਲਸ)	ਇਕ ਵੇਰਿਏਬਲਸ ਜਿਹੜਾ ਸਮੁੱਚੇ ਪ੍ਰੋਗਰਾਮ ਦੇ ਵਿਚ ਪਹੁੰਚਯੋਗ ਹੁੰਦਾ ਹੈ ਜਿਸਦਾ ਜੀਵਨ ਕਾਲ ਪ੍ਰੋਗਰਾਮ ਜਿਨ੍ਹਾ ਹੀ ਹੁੰਦਾ ਹੈ।
Goto	C ਕੀ-ਵਰਡ ਜੋ C ਫੰਕਸ਼ਨ ਦੇ ਵਿਚ ਕੰਟਰੋਲ ਬਦਲਣ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
ਐਚ (H)	
Header (ਹੈਡਰ)	ਇਹ ਫਾਈਲ ਸਟਰਕਚਰ ਦਾ ਉਹ ਹਿੱਸਾ ਹੁੰਦਾ ਹੈ ਜਿਸ ਵਿਚ ਕਾਂਟੈਨਟਸ (contents) ਨੂੰ render ਕਰਨ ਲਈ information ਸਟੋਰ ਹੁੰਦੀ ਹੈ।
Header File (ਹੈਡਰ ਫਾਈਲ)	ਇਕ ਫਾਈਲ ਜਿਸ ਵਿਚ ਸ਼੍ਰੇਣੀ ਡਿਕਲੇਅਰਸ, ਪ੍ਰੀਪ੍ਰੋਸੈਸਰ, ਡਾਇਰੈਕਟਿਵ ਆਦਿ ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ ਅਤੇ ਰੁਪਾਂਤਰ ਇਕਾਈ ਵਿਚ ਸ਼ਾਮਲ ਹੁੰਦੇ ਹਨ। ਇਨ੍ਹਾਂ ਦਾ ਵਿਸਥਾਰ ਪ੍ਰੀਪ੍ਰੋਸੈਸਰ ਦੁਆਰਾ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
ਆਈ (I)	
Initialization (ਇਨੀਸ਼ਿਯਲਾਈਜ਼ੇਸ਼ਨ)	ਇਕ ਵੇਰਿਏਬਲਸ ਜਾਂ ਕੌਨਸਟੈਂਟ ਨੂੰ ਮੁਢਲਾ ਮੁੱਲ ਦੇਣ ਲਈ।
Initialize (ਇਨੀਸ਼ਿਯਲਾਈਜ਼)	ਆਰੰਭ-ਕਰਨ ਦੀ ਪ੍ਰਕ੍ਰਿਆ।
Initializer (ਇਨੀਸ਼ਿਯਲਾਈਜ਼ਰ)	ਆਰੰਭ-ਕਰਨ ਸਮੇਂ ਇਕ ਆਬਜੈਕਟ ਨੂੰ ਇਨੀਸ਼ਿਯਲਾਈਜ਼ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਐਕਸਪ੍ਰੈਸ਼ਨ ਜਾਂ ਮੁੱਲ
Int (ਇੰਟ)	C ਕੀ-ਵਰਡ ਅਤੇ ਮੌਲਿਕ ਟਾਈਪ ਜੋ ਇਨਟੀਜ਼ਰਲ ਟਾਈਪ ਨੂੰ ਡਿਕਲੇਅਰ ਕਰਨ ਲਈ ਵਰਤੇ ਜਾਂਦੇ ਹਨ।
Integral Variable (ਇਨਟੀਜ਼ਰਲ ਵੇਰਿਏਬਲਸ)	ਸਿਸਟਮ ਜਿਸ ਰਾਹੀ ਇਨਟੀਜ਼ਰਲ ਨੂੰ ਹਸਤਾਖਰ ਜਾਂ ਗੈਰ ਹਸਤਾਖਰ ਵਿਚ ਵੈਰੀਏਬਲਸ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
Internet Banking (ਇੰਟਰਨੈੱਟ ਬੈਂਕਿੰਗ)	ਇਸ ਪ੍ਰਕਾਰ ਬੈਂਕਿੰਗ ਦੁਆਰਾ ਆਮ ਬੈਂਕਿੰਗ ਸੁਵਿਧਾਵਾਂ ਜਿਵੇਂ ਕਿ ਭੁਗਤਾਨ (payment), ਮਨੀ ਟਰਾਂਸਫਰ (money transfer) ਅਤੇ ਅਕਾਂਉਟ ਬਕਾਇਆ (account balance) ਆਦਿ ਵਰਗੇ ਕੰਮ internet ਦੁਆਰਾ ਮੁਹਈਆ ਕਰਵਾਏ ਜਾਂਦੇ ਹਨ।
i-ticketing (ਆਈ-ਟਿਕਟਿੰਗ)	ਇਹ ਇਕ ਮੁਫਤ web based ਟਿਕਟਿੰਗ ਸਿਸਟਮ ਹੈ
ਕੇ (K)	
Keyword (ਕੀ-ਵਰਡ)	C ਵਿਚ ਰਾਖਵਾਂ ਸ਼ਨਾਖਤ ਕਰਤਾ, ਜੋ ਡਾਟਾ-ਟਾਈਪ ਭਾਸ਼ਾ ਦੇ ਸਟੇਟਮੈਂਟ ਆਦਿ ਨੂੰ ਸੂਚਿਤ ਕਰਦਾ ਹੈ

ਐਲ (L)	
Library (ਲਾਇਬ੍ਰੇਰੀ)	ਫਾਈਲਾਂ ਦਾ ਸੈਟ ਜੋ ਇਕੱਠਾ ਹੁੰਦਾ ਹੈ। ਇਕ ਲਿੰਕਰ ਵਲੋਂ ਉਨ੍ਹਾਂ ਨੂੰ ਮੁੜ-ਮੁੜ ਖੋਜਿਆ ਜਾਂਦਾ ਹੈ ਅਤੇ ਜਿਹੜੀਆਂ ਲਕਸ਼ ਫਾਈਲਾਂ ਦੀ ਲੋੜ ਹੁੰਦੀ ਹੈ ਵਰਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ।
literal (ਸ਼ਾਬਦਿਕ)	1234 ਦੀ ਤਰ੍ਹਾਂ ਸਥਾਈ।
Local variable (ਲੋਕਲ ਵੇਰਿਏਬਲਸ)	ਇਕ ਵੇਰੀਏਬਲਸ ਜੋ ਇਕ ਫੰਕਸ਼ਨ ਦੁਆਰਾ ਸਥਾਨਕ ਡਿਕਲੇਅਰ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
Long (ਲਾਂਗ)	C ਵਿਚ ਇਕ ਲੰਬੇ ਪੂਰਣ ਅੰਕ ਡਾਟਾ ਕਿਸਮ ਨੂੰ ਡਿਕਲੇਅਰ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ
Long double (ਲਾਂਗ ਡਬਲ)	C ਵਿਚ ਅਸਥਾਈ ਨੁਕਤਾ ਕਿਸਮ।
ਐਮ (M)	
Microphone Noise (ਮਾਈਕਰੋਫੋਨ ਨੋਆਈਸ)	ਇਹ ਅਣਚਾਹੀ ਆਵਾਜ਼ ਹੁੰਦੀ ਹੈ ਅਤੇ microphone ਨਾਲ ਜਦੋਂ ਕੋਈ ਆਬਜੈਕਟ ਸੰਪਰਕ ਕਰਦਾ ਹੈ ਉਸ ਵੇਲੇ ਪੈਦਾ ਹੁੰਦੀ ਹੈ
ਐਨ (N)	
Network Modem (ਨੈੱਟਵਰਕ ਮਾਡਮ)	ਇਕ ਮਾਡਮ ਜੋ ਕਿ ਇਕ ਸਥਾਨਕ ਖੇਤਰੀ ਲੋਕਲ ਏਰੀਆ ਨੈੱਟਵਰਕ (ਲੈਨ) ਜੋ ਕਿ ਨੈੱਟਵਰਕ ਤੇ ਕਿਸੇ ਵੀ ਨੈੱਟਵਰਕ ਸਟੇਸ਼ਨ ਤੋਂ ਪਹੁੰਚਣ ਯੋਗ ਹੁੰਦਾ ਹੈ।
Network Operating System (ਨੈੱਟਵਰਕ ਉਪਰੇਟਿੰਗ ਸਿਸਟਮ)	ਨੈੱਟਵਰਕ ਓਪਰੇਟਿੰਗ ਸਿਸਟਮ ਇਕ ਤੋਂ ਵਧੇਰੇ ਕੰਪਿਊਟਰਾਂ ਦੇ ਦਰਮਿਆਨ ਸੰਚਾਰ ਅਤੇ ਸੂਚਨਾ ਭੇਜਣ ਲਈ ਤਿਆਰ ਕੀਤਾ ਗਿਆ ਹੈ। ਮਿਸਾਲਾਂ ਲਈ Windows NT ਸਰਵਰ ਸ਼ਾਮਲ ਹਨ।
Node (ਨੋਡ)	ਇਕ ਨੈੱਟਵਰਕ ਕੁਨੈਕਸ਼ਨ ਦਾ ਅੰਤਿਮ ਬਿੰਦੂ। ਨੋਡਜ ਵਿਚ ਇਕ ਨੈੱਟਵਰਕ ਦੇ ਨਾਲ ਜੁੜਿਆ ਹੋਇਆ ਕੋਈ ਵੀ ਡਿਵਾਈਸ ਸ਼ਾਮਲ ਹੁੰਦਾ ਹੈ ਜਿਵੇਂ ਕਿ ਫਾਈਲ ਸਰਵਰ, ਪ੍ਰਿੰਟਰ ਜਾਂ ਵਰਕ-ਸਟੇਸ਼ਨ।
Null (ਨਲ)	ਇਕ ਵਿਸ਼ੇਸ਼ ਕਾਨਸਟੈਂਟ ਮੁੱਲ।
ਓ (O)	
Object (ਆਬਜੈਕਟ)	ਇਸ ਦੇ ਅਨੇਕਾਂ ਅਰਥ ਹਨ। C++, ਵਿਚ ਇਕ ਸ਼੍ਰੇਣੀ ਦੀ ਮਿਸਾਲ ਦਾ ਅਕਸਰ ਹਵਾਲਾ ਦਿਤਾ ਜਾਂਦਾ ਹੈ ਅਤੇ ਇਹ ਇਕ ਵੇਰੀਏਬਲਸ ਜਾਂ ਹੋਰ ਵਜੂਦ ਜਿਸ ਵਿਚ ਭੰਡਾਰਣ ਸ਼ਾਮਲ ਹੁੰਦਾ ਹੈ।
ਪੀ (P)	
Parameter (ਪੈਰਾਮੀਟਰ)	ਇਹ ਕਾਲਿੰਗ ਕੋਡ ਨੂੰ ਬੁਲਾਏ ਜਾਣ ਤੇ ਵਿਧੀ ਦੇ ਮੁੱਲ ਦੀ ਆਗਿਆ ਦਿੰਦਾ ਹੈ।
Ports (ਪੋਰਟਸ)	ਇਕ ਕੇਬਲ ਲਈ ਕੁਨੈਕਸ਼ਨ ਪੁਆਇੰਟ।
Programming Environment (ਪ੍ਰੋਗਰਾਮਿੰਗ ਵਾਤਾਵਰਣ)	ਵਿਕਸਿਤ ਹੋ ਰਹੇ ਸਾਫ਼ਟ-ਵੇਅਰ ਵਿਚ ਜਿਸ ਵਿਚ ਇਕ ਕੰਪਾਇਲਰ, ਲਿੰਕਰ, ਡੀ-ਬਗਰ ਅਤੇ ਬ੍ਰਾਉਜ਼ਰ ਸ਼ਾਮਲ ਲਈ, ਵਰਤੇ ਜਾਣ ਵਾਲੇ ਸੰਗਠਿਤ ਉਪਕਰਣਾਂ ਦਾ ਸੈਟ।
Project File (ਪ੍ਰੋਜੈਕਟ ਫਾਈਲ)	ਇਹ ਚਲ ਰਹੇ (current) ਪ੍ਰਜੈਕਟ ਦੀ ਉਹ ਫਾਈਲ ਹੁੰਦੀ ਹੈ ਜਿਸ ਵਿਚ import ਕੀਤੀਆਂ ਗਈਆਂ ਫਾਈਲਾਂ ਬਾਰੇ ਜਾਣਕਾਰੀ ਉਪਲਬਧ ਹੁੰਦੀ ਹੈ।
Protocol (ਪ੍ਰੋਟੋਕੋਲ)	ਨਿਯਮਾ ਅਤੇ ਪਰੰਪਰਾਵਾਂ ਦੇ ਇਕ ਸੈਟ ਦਾ ਰਸਮੀ ਸਟੇਟਮੈਂਟ ਜਿਹੜਾ ਇਸ ਗੱਲ ਦਾ ਪ੍ਰਬੰਧ ਕਰਦਾ ਹੋਵੇ ਕਿ ਇਕ ਨੈੱਟਵਰਕ ਉਤੇ ਡਿਵਾਈਸ ਕਿਵੇਂ ਸੂਚਨਾਵਾਂ ਅਦਾਨ-ਪ੍ਰਦਾਨ ਕਰਦੇ ਹਨ।

ਆਰ (R)	
Reference (ਹਵਾਲਾ)	ਇਕ ਆਬਜੈਕਟ ਲਈ ਹੋਰ ਨਾਮ/ਇਕ ਆਬਜੈਕਟ ਵਲ ਪਹੁੰਚ ਜੋ ਕਿ ਹਵਾਲੇ ਦੀ ਵਰਤੋਂ ਕਰਦੀ ਹੈ। ਹਵਾਲਿਆਂ ਨੂੰ ਵਿਸ਼ੇਸ਼ ਕਰਕੇ ਪੁਆਂਇੰਟਰਾਂ ਵਜੋਂ ਲਾਗੂ ਕੀਤਾ ਜਾਂਦਾ ਹੈ।
Register (ਰਜਿਸਟਰ)	C ਕੀ-ਵਰਡ ਦੀ ਵਰਤੋਂ ਕੰਪਾਇਲਰ ਨੂੰ ਇਸ਼ਾਰੇ ਵਜੋਂ ਕੀਤੀ ਜਾਂਦੀ ਹੈ ਕਿ ਇਕ ਵਿਸ਼ੇਸ਼ ਸਥਾਨਕ ਵੇਰੀਏਬਲ ਇਕ ਮਸ਼ੀਨ ਰਜਿਸਟਰ ਵਿਚ ਰਖਿਆ ਜਾਣਾ ਚਾਹੀਦਾ ਹੈ।
Return (ਰਿਟਰਨ)	C ਕੀ-ਵਰਡ ਦੀ ਵਰਤੋਂ ਇਕ ਫੰਕਸ਼ਨ ਤੋਂ ਵਾਪਸ ਮੋੜਣ ਵਾਲੇ ਮੁੱਲ ਲਈ ਕੀਤੀ ਜਾਂਦੀ ਹੈ।
Return value (ਰਿਟਰਨ ਮੁੱਲ)	ਇਕ ਫੰਕਸ਼ਨ ਤੋਂ ਵਾਪਸ ਆਇਆ ਮੁੱਲ।
ਐਸ (S)	
Short (ਛੋਟਾ)	ਇਕ C ਬੁਨਿਆਦੀ ਕਿਸਮ ਜਿਹੜੀ ਛੋਟੇ ਪੂਰਨ ਅੰਕਾਂ ਦਾ ਐਲਾਨ ਕਰਨ ਲਈ ਵਰਤੀ ਜਾਂਦੀ ਹੈ।
Signed (ਸਾਈਂਡ)	C ਕੀ-ਵਰਡ ਜੋ ਕਿ ਹਸਤਾਖਰ ਡਾਟਾ-ਟਾਈਮ ਦੇ ਪ੍ਰਤੀਕ ਵਜੋਂ ਵਰਤੀ ਜਾਂਦੀ ਹੈ।
Sizeof (ਆਕਾਰ ਦਾ)	C ਕੀ-ਵਰਡ ਜੋ ਕਿ ਇਕ ਆਬਜੈਕਟ ਜਾਂ ਟਾਈਪ ਦਾ ਆਕਾਰ ਲੈਣ ਲਈ ਹੁੰਦਾ ਹੈ।
Speed of Data Transfer (ਡਾਟਾ ਬਦਲੀ ਦੀ ਗਤੀ)	ਉਹ ਦਰ ਜਿਸ ਤੇ ਇਕ ਨੈੱਟਵਰਕ ਰਾਹੀ ਸੂਚਨਾ ਯਾਤਰਾ ਕਰਦੀ ਹੈ ਜੋ ਕਿ ਆਮ ਤੌਰ ਤੇ ਪ੍ਰਤੀ ਸੈਕਿੰਡ ਮੈਗਾ-ਬਿੱਟ ਵਿਚ ਮਾਪੀ ਜਾਂਦੀ ਹੈ।
Split (ਸਪਲਿਟ)	ਆਡਿਓ ਜਾਂ ਵੀਡਿਓ ਕਲਿਪ ਨੂੰ ਦੋ ਭਾਗਾਂ ਵਿਚ ਵੰਡਣਾ।
Star-Wired Ring (ਸਟਾਰ-ਵਾਈਰਡ ਰਿੰਗ)	ਨੈੱਟਵਰਕ ਟੋਪੋਲੋਜੀ ਜਿਹੜੀ ਨੈੱਟਵਰਕ ਯੰਤਰਾਂ ਨੂੰ ਜੋੜਦੀ ਹੈ। (ਜਿਵੇਂ ਕਿ ਸੰਪੂਰਨ ਸਰਕਲ ਵਿਚ ਕੰਪਿਊਟਰ ਅਤੇ ਪ੍ਰਿੰਟਰ।)
Statement (ਸਟੇਟਮੈਂਟ)	ਇਕ ਪ੍ਰੋਗਰਾਮ ਦੇ ਭਾਗ ਜਿਹੜੇ ਅਸਲ ਵਿਚ ਕੰਮ ਕਰਦੇ ਹਨ।
Static Member (ਸਟੈਟਿਕ ਮੈਂਬਰ)	ਇਕ ਸ਼੍ਰੇਣੀ ਜਿਹੜੇ ਕਿ ਪਹੁੰਚ ਕੰਟਰੋਲ ਦੇ ਮਕਸਦ ਲਈ ਇਕ ਸ਼੍ਰੇਣੀ ਦਾ ਭਾਗ ਹੁੰਦੇ ਹਨ ਪ੍ਰੰਤੂ ਸ਼੍ਰੇਣੀ ਦੇ ਕਿਸੇ ਵਿਸ਼ੇਸ਼ ਆਬਜੈਕਟ ਮਿਸਾਲਾਂ ਤੇ ਕੰਮ ਨਹੀਂ ਕਰਦੇ।
Static Object (ਸਟੈਟਿਕ ਆਬਜੈਕਟ)	ਇਕ ਆਬਜੈਕਟ ਜਿਹੜਾ ਕਿ ਫੰਕਸ਼ਨ ਲਈ ਸਥਾਨਕ ਹੁੰਦਾ ਹੈ ਜਾਂ ਰੁਪਾਂਤਰ ਇਕਾਈ ਲਈ ਅਤੇ ਜਿਸ ਦਾ ਜੀਵਨਕਾਲ ਇਕ ਪ੍ਰੋਗਰਾਮ ਦੇ ਜੀਵਨ ਬਰਾਬਰ ਹੁੰਦਾ ਹੈ।
Storyboard (ਸਟੋਰੀ-ਬੋਰਡ)	ਇਹ ਵਰਕ-ਸਪੇਸ (workspace) ਦਾ ਉਹ ਭਾਗ ਹੁੰਦਾ ਹੈ ਜਿਸ ਵਿਚ ਕਲਿੱਪਸ ਦੀ ਤਰਤੀਬ ਨੂੰ ਦਿਖਾਇਆ ਜਾਂਦਾ ਹੈ।
Suvidha Centre (ਸੁਵਿਧਾ ਸੈਂਟਰ)	ਇਹਨ੍ਹਾਂ ਸੈਂਟਰਾਂ ਵਿਚ ਜਨਤਾ ਦੀਆਂ ਆਮ ਜਰੂਰਤ ਆਉਣ ਵਾਲੀਆਂ ਸੇਵਾਵਾਂ /ਸਹੂਲਤਾਂ ਪ੍ਰਧਾਨ ਕੀਤੀਆਂ ਜਾਂਦੀਆਂ ਹਨ ਜਿਵੇਂ ਕਿ ਪਾਸਪੋਰਟ ਬਨਵਾਉਣਾ, ਵਾਹਨਾਂ ਦੀ ਰਜ਼ਿਸਟਰੇਸ਼ਨ ਆਦਿ।
Switch (ਸਵਿਚ)	ਕੀ-ਵੱਰਡ ਇਕ ਸਟੇਟਮੈਂਟ ਕਿਸਮ ਦੀ ਟਿੱਪਣੀ ਕਰਦਾ ਹੈ ਜੋ ਕਿ ਮੁੱਲ ਉਤੇ ਆਧਾਰਤ ਸਟੇਟਮੈਂਟਸ ਦੇ ਅਨੇਕਾਂ ਭਾਗਾਂ ਚੋਂ ਇਕ ਨੂੰ ਡਿਸਪੈਚ ਕਰਨ ਲਈ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ।
Syntax (ਸਟਰਿੰਗ ਬਣਤਰ)	ਸਟਰਿੰਗ-ਬਣਤਰ ਨਿਯਮ ਜਿਹੜੇ ਕਿ C ਵਿਚ ਪ੍ਰਬੰਧ ਕਰਦੇ ਹਨ ਕਿ ਐਕਸਪ੍ਰੇਸ਼ਨ, ਸਟੇਟਮੈਂਟ, ਡਿਕਲੇਅਰੇਸ਼ਨ ਅਤੇ ਪ੍ਰੋਗਰਾਮ ਕਿਵੇਂ ਬਣਾਉਣੇ ਹਨ।
ਟੀ (T)	
Tag (ਟੈਗ)	ਇਕ ਸ਼੍ਰੇਣੀ, ਨਿਰਮਾਣ ਜਾਂ ਏਕਤਾ ਨੂੰ ਦਿੱਤਾ ਗਿਆ ਨਾਮ।
Timeline (ਟਾਈਮ-ਲਾਈਨ)	ਇਹ ਯੂਜ਼ਰ interface ਦਾ ਉਹ ਭਾਗ ਹੈ ਹੁੰਦਾ ਹੈ ਜਿਸ ਵਿਚ ਪ੍ਰੋਜੈਕਟ ਨੂੰ ਬਨਾਉਣ ਵਾਲੀਆਂ ਜਾਂ ਕਲਿਪਸ ਨੂੰ ਦਿਖਾਇਆ ਜਾਂਦਾ ਹੈ।
Trim (ਟਰਿਮ)	ਇਸ ਨਾਲ ਅਸੀ ਫਾਈਲ ਜਾਂ ਕਲਿਪ ਦੇ ਕੁਝ ਭਾਗਾਂ ਨੂੰ ਡਿਲੀਟ (delete) ਕੀਤੇ ਬਗੈਰ ਉਹਨ੍ਹਾਂ ਦੇ ਸਥਾਨ ਤੋਂ ਛੁਪਾ ਸਕਦੇ ਹਾਂ।
Trim Point (ਟਰਿਮ ਪੁਆਂਇਟ)	ਇਹ ਪੁਆਇਟ ਫਾਈਲ ਜਾਂ ਕਲਿੱਪ ਵਿਚ ਪਲੇਅ ਬੈਕ (playback) ਲਈ ਸ਼ੁਰੂਆਤ ਅਤੇ ਅੰਤ ਨੂੰ ਦਰਸਾਉਂਦੇ ਹਨ। ਆਮ ਤੌਰ ਤੇ ***start trim*** ਪੁਆਂਇੰਟ ਅਤੇ ***end trim*** ਪੁਆਂਇੰਟ ਹੁੰਦੇ ਹਨ।

ਯੂ (U)	
Unsigned (ਅਨਸਾਈਂਡ)	C ਕੀ-ਵਰਡ ਵਲੋਂ ਪੂਰਨ ਅੰਕ ਗੈਰ ਅਧਿਕਾਰਤ ਬੁਨਿਆਦੀ ਟਾਈਪ ਨੂੰ ਡਿਕਲੇਅਰ ਕਰਦਾ ਹੈ।
USB (ਯੂ ਐਸ ਬੀ)	ਪੋਰਟ ਇਕ ਹਾਰਡਵੇਅਰ ਜਿਹੜਾ ਘਟ ਗਤੀ ਦੀਆਂ ਸੀਮਾਵਾਂ ਲਈ ਇੰਟਰਫੇਸ ਕਰਦਾ ਹੈ। ਜਿਵੇਂ ਕਿ ਕੀ-ਵਰਡ, ਮਾਊਸ, ਸਕੈਨਰ, ਪ੍ਰਿੰਟਰ ਅਤੇ ਟੈਲੀਫੋਨੀ ਡਿਵਾਈਸ ਆਦਿ।
ਡਬਲਯੂ (W)	
While (ਵਾਈਲ)	C ਭਾਸ਼ਾ ਵਿਚ ਇਕ ਲੂਪ ਜੋ ਸਟੇਟਮੈਂਟਸ ਦੀ ਪਰਿਸਥਿਤੀ ਨੂੰ ਲਾਗੂ ਕਰਨ ਲਈ ਬਾਰ-ਬਾਰ ਵਰਤਿਆ ਜਾਂਦਾ ਹੈ ਜਦ ਤਕ ਨਿਸ਼ਚਤ ਹਾਲਾਤ *(condition)* ਸਹੀ ਹੈ।
Window Media File (ਵਿੰਡੋਜ਼ ਮੀਡਿਆ ਫਾਈਲ)	ਇਹ ਫਾਈਲ ਆਡੀਓ, ਵੀਡੀਓ ਅਤੇ ਸਕਰਿਪਟ (script) ਡਾਟਾ ਨੂੰ ਸਟੋਰ ਕਰਦੀ ਹੈ। ਵਿੰਡੋਜ਼ ਮੀਡਿਆ ਫਾਈਲ ਦੀਆਂ ਐਕਸਟੈਂਸ਼ਨ (extension) ਹਨ .wma; .wme; .wms; .wmv ਆਦਿ।
Window Media Technologies (ਵਿੰਡੋਜ਼ ਮੀਡਿਆ ਤਕਨੀਕਾਂ)	ਇਹ ਉਹ ਡਿਜੀਟਲ ਮਿਡੀਆ ਸਾਫਟਵੇਅਰ ਹਨ ਜੋ ਮਾਈਕਰੋਸਾਫਟ (Microsoft) ਨੇ ਬਣਾਏ ਹਨ। ਜਿਵੇਂ ਕਿ ਵਿੰਡੋਜ਼ ਮਿਡੀਆ ਸਰਵਿਸ, ਵਿੰਡੋਜ਼ ਮਿਡੀਆ encodes ਆਦਿ।
Windows Movie Makers (ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ)	ਇਸ ਸਾਫਟਵੇਅਰ ਦੀ ਮੱਦਦ ਨਾਲ ਅਸੀ ਆਡੀਓ ਅਤੇ ਵੀਡੀਓ ਨੂੰ capture, ਐਡਿਟ ਅਤੇ arrange ਕਰ ਸਕਦੇ ਹਾਂ।
Workgroup (ਵਰਕਗਰੁੱਪ)	LAN ਤੇ ਵਰਕ-ਸਟੇਸ਼ਨ ਅਤੇ ਸਰਵਰ ਜੋ ਇਕ ਦੂਜੇ ਨਾਲ ਡਾਟਾ ਦਾ ਵਿਚਾਰ ਵਟਾਂਦਰਾ ਅਤੇ ਸੰਚਾਰ ਕਰਨ ਲਈ ਬਣਾਇਆ ਗਿਆ ਹੈ।
Workspace (ਵਰਕ-ਸਪੇਸ)	ਇਹ ਵਿੰਡੋਜ਼ ਮੂਵੀ ਮੇਕਰ ਦਾ ਉਹ ਭਾਗ ਹੈ ਜਿਸ ਵਿਚ ਮੂਵੀ ਬਣਾਈ ਜਾ ਸਕਦੀ ਹੈ। ਇਸ ਦੇ ਦੋ ਵਿਊ (view) ਹੁੰਦੇ ਹਨ : (1) ਸਟੋਰੀ-ਬੋਰਡ (story board) ਅਤੇ (2) ਟਾਈਮਲਾਈਨ (timeline).
Workstation (ਵਰਕਸਟੇਸ਼ਨ)	ਵਰਕਸਟੇਸ਼ਨ ਨੈੱਟਵਰਕ ਦੇ ਨਾਲ ਜੁੜਿਆ ਹੋਇਆ ਇਕ ਕੰਪਿਊਟਰ ਜਿਸ ਤੇ ਯੂਜ਼ਰ ਨੈੱਟਵਰਕ ਤੇ ਸਟੋਰ ਕੀਤੇ ਗਏ ਸਾਫਟਵੇਅਰ ਦੇ ਨਾਲ ਪਰਸਪਰ ਪ੍ਰਭਾਵ ਲੈਂਦਾ ਹੈ।