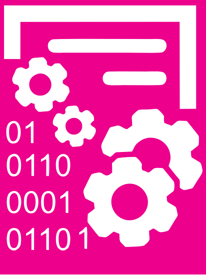
یونٹ 2 بائنری سسٹم



مختصر تعارف (Short Introduction)

کمپیوٹر صرف ''0'' اور ''1'' کی زبان سمجھتا ہے جسے مشین لینگویج کہتے ہیں۔ ایسا نمبر سسٹم جو صرف ''0'' اور ''1'' پر مشتمل ہو بائنری سسٹم کہلاتا ہے۔ کمپیوٹر کے استعمالات میں انٹرنیٹ سرفنگ، گیمیں کھیلنا، موویز دیکھنا اور دستاویزات تیار کرنا شامل ہے۔ یہ تمام چیزیں ''0'' اور ''1'' میں کیسے تبدیل ہو جاتی ہیں؟ اس یونٹ میں اس طریقہ کار پر بحث کی گئی ہے۔

حاصلاتِ تعلّم (Students Learning Outcomes)

1- نمبر سسٹم کا تعارف

- مندرجہ ذیل نمبر سسٹم پر بحث کرنا:
  - بائنری
  - ڈیسیمل
  - ہیگزاڈیسیمل

2- نمبر سسٹم کا تبادلہ

- بائنری سے ڈیسیمل اور ڈیسیمل سے بائنری سسٹم میں تبادلہ۔
- ڈیسیمل سے ہیگزاڈیسیمل اور ہیگزاڈیسیمل سے ڈیسیمل سسٹم میں تبادلہ۔
- بائنری سے ہیگزاڈیسیمل اور ہیگزاڈیسیمل سے بائنری سسٹم میں تبادلہ۔

3- میموری اور ڈیٹا سٹوریج

- میموری کیا ہے؟
- کمپیوٹر میموری میں ڈیٹا کی نمائندگی۔
- سٹوریج ڈیوائسس۔
- میموری اور سٹوریج ڈیوائسس میں فرق

4- کمپیوٹر میموری کے سائز کی پیمائش

- درجہ ذیل اصلاحات کی وضاحت کریں:
  - بٹ (Bit)
  - بائٹ (Byte)
  - کلوبائٹ (Kilo Byte)
  - میگابائٹ (Megabyte)
  - گیگابائٹ (Gegabyte)
  - ٹیرابائٹ (Terabyte)
  - پیٹابائٹ (Petaabyte)

5- بولین الجبرا

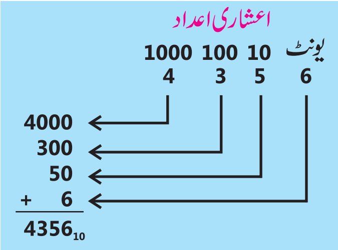
ذیل کی ضاحت کریں:

- بولین پری پوزیشن
- ٹروتھ ویلیوز (Truth Values)
- لاجیکل آپریٹرز (AND, OR, NOT)
- ٹروتھ ٹیبل (Truth Table)
- بولین الجبرا کے قوانین:
  - قانونِ تلازم
  - قانونِ تبادلہ
  - قانونِ تقسیمی
  - جمعی اور ضربی ذاتی قانون
- لاجیکل ایکسپریشن۔

2.1 نمبرسسٹم کا تعارف

عددی نظام اعدادوشمار کی نمائندگی کے لیے ایک سسٹم ہے جسے نمبر سسٹم کہتے ہیں۔ عددی مواد کا اظہار جس نظام کے تحت ہوتا ہے اُسے عددی نظام یا نمبر سسٹم کہتے ہیں۔ ہم سب اعداد کے اعشاری نظام سے واقف ہیں جہاں ہر عدد '0' سے '9' تک کے اعداد پر مشتمل ہوتا ہے۔ کمپیوٹر سسٹم میں دوسرے عددی نظام بھی استعمال ہوتے ہیں۔ ذیل میں ہم چند عددی نظاموں پر بحث کرتے ہیں:

2.1.1 اعشاری (Decimal)



شکل 2-1

ہم اپنی روزمرہ زندگی میں جس عددی نظام کو استعمال میں لاتے ہیں وہ اعداد کا اعشاری نظام ہے۔ اعشاری عددی نظام جیسا کہ نام سے بھی ظاہر ہے کی بنیاد (Base) 10 پر ہے۔ اس نظام میں ہر ہندسے کی پوزیشن کا اظہار بھی 10 کی مخصوص طاقت کے ذریعے کیا جاتا ہے۔ جیسا کہ شکل (2.1) میں دکھایا گیا ہے۔

مثال: **796**

کیا آپ جانتے ہیں؟

ریاضی میں اعداد کے اعشاری نظام کو ہندعربیک یا عربیک عددی نظام بھی کہتے ہیں۔

2.1.2 ثنائی (Binary)

ثنائی عددی نظام میں بیس دو (2) ہوتی ہے کیونکہ اس سسٹم میں تمام نمبر صرف دو ہندسوں پر مشتمل ہوتے ہیں (1 یا 0)۔ ڈیجیٹل کمپیوٹر میں ڈیٹا کو ذخیرہ کرنے کے لیے اس سسٹم کا استعمال کیا جاتا ہے۔ آپ کا نام حروف کی شکل میں ہوتا ہے لیکن کمپیوٹر کے لیے ہر حرف تہجی کی کچھ ثنائی (Binary Value) قدر ہوتی ہے۔

مثال: حرف 'A' کی ثنائی قدر 01000001 ہے اور اس کی اعشاری قدر '65' ہے۔

2.1.3 ہیگزاڈیسیمل Hexadecimal

ہیگزاڈیسیمل (Hexadecimal) سسٹم میں کل سولہ (16) نمبر ہوتے ہیں۔ مثلاً 2,1,0.............,F,E,C جہاں A=10، B=11، C=12، D=13، E=14 اور F=15

جیسا کہ شکل 2.2 میں دکھایا گیا ہے۔

مثال: 3F2B

اعشاری عددی نظام

دس مختلف علامات استعمال ہوتی ہیں

0, 1, 2, 3, 4, 5, 6, 7, 8, 9

ہیگزاڈیسیمل نظام

سولہ مختلف علامات استعمال ہوتی ہیں

0, 1, 2, 3, 4, 5, 6, 7, 8, 9
A, B, C, D, E, F

شکل 2-2

2.2 نمبر سسٹم کا تبادلہ:

ہم کسی بھی نمبر کو ایک نمبر سسٹم میں سے کسی بھی دوسرے نمبر سسٹم کے ساتھ تبدیل کر سکتے ہیں۔ ذیل میں، ہم مختلف نمبر سسٹم کے تبادلوں پر گفتگو کرتے ہیں۔

2.2.1 اعشاری سے ثنائی اور ثنائی سے اعشاری نظام تغیر و تبدل

اعشاری نمبر کو ثنائی میں تبدیل کرنے کے لیے ہم اس نمبر کو دو پر تقسیم کرتے ہیں اور حاصلِ تقسیم کو Quotient اور باقی کو remainder کہتے ہیں۔

حاصلِ تقسیم کو دو سے تقسیم کرتے رہتے ہیں۔ جب تک کہ ہم حاصل تقسیم 0 حاصل نہیں کر لیتے۔ ثنائی نمبر حاصل کرنے کے لیے ہم تمام باقی (remainder) کو الٹ ترتیب میں لکھتے ہیں۔

مثال: 156_{10} (156 اعشاری میں) کو ثنائی میں تبدیل کریں۔ ٹیبل 2.1 میں اس مسئلے کو حل کرنے کا طریقہ دکھایا گیا ہے۔ ثنائی نمبر میں لکھنے کے لیے باقی (remainder) کو نیچے سے اوپر کی طرف لے جایا جاتا ہے اسطرح $156_{10} = 1001100_2$

2	156	
2	78	– 0
2	39	-- 0
2	19	-- 1
2	9	-- 1
2	4	-- 1
2	2	-- 0
2	1	-- 0
2	0	-- 1

ٹیبل 2.1

سرگرمی

آٹھویں کلاس کے سالانہ امتحان میں آپ کو کتنے نمبر ملے تھے؟ ان نمبروں کو ثنائی میں تبدیل کریں اور اپنی کلاس کے ساتھیوں کے ساتھ نتائج (results) پر تبادلہ خیال کریں۔

ثنائی سے اعشاری:

ایک نمبر کو ثنائی نمبر سسٹم سے اعشاری نمبر سسٹم میں تبدیل کرنے کا عمل ایک مثال کی مدد سے نیچے بیان کیا گیا ہے۔

مثال:

$(1000001)_2$ کو اعشاری میں تبدیل کریں۔

$$= 1 \times 2^6 + 0 \times 2^5 + 0 \times 2^4 + 0 \times 2^3 + 0 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$
$$= 64 + 0 + 0 + 0 + 0 + 0 + 1$$
$$= (65)_{10}$$

مندرجہ بالا تبدیلی کو مندرجہ ذیل مراحل میں کیا گیا ہے۔

i۔ ثنائی نمبر لکھیں جیسا کہ اس مثال میں $(1000001)_2$

ii۔ 0 سے شروع کرتے ہوئے دو کی طاقتوں کو دائیں سے بائیں جانب لکھیں۔ اس مثال میں، دو کی طاقت 0 سے شروع ہوتی ہے اور 6 پر ختم ہوتی ہے۔

iii۔ دو کی متعلقہ طاقتوں کو ہر ثنائی قدر (Binary value) سے ضرب دیں۔مندرجہ بالا مثال میں'6' ثنائی قدر ہے۔

iv۔ ہر قیمت(value) شمار کریں۔

v۔ تمام قیمتوں (values) کو جمع کریں۔

vi۔ جواب کو اس کی بیس علامت کے ساتھ لکھیں۔

سرگرمی: 2.2

اپنے نمبروں کو ثنائی شکل میں اپنے دوستوں کے تبادل کریں اور جماعت نہم کے بورڈ امتحان میں ان کی توقعات کے بارے میں جاننے کے لیے انھیں اعشاری میں تبدیل کریں۔اپنی کلاس کے دوستوں کے ساتھ ڈبل چیک کریں کہ آپ کا شمار درست ہے یا نہیں!

سرگرمی: 2.3

ٹیبل (2.2) کے مطابق، اعشاری، ثنائی اور ہیگزاڈیسمیل میں اپنے وقت کو لکھیں:

- سکول آنے کا
- دوپہر کے کھانے کا
- کھیلنے کا

ہیگزاڈیسمیل	ثنائی	اعشاری	ہیگزاڈیسمیل	ثنائی	اعشاری
			0	0	0
B	1011	11	1	1	1
C	1100	12	2	10	2
D	1101	13	3	11	3
E	1110	14	4	100	4
F	1111	15	5	101	5
10	10000	16	6	110	6
11	10001	17	7	111	7
12	10010	18	8	1000	8
13	10011	19	9	1001	9
14	10100	20	A	1010	10

ٹیبل2-2

سرگرمی: 2.4

نمبر سسٹم کے لیے بہت سے آن لائن کنورٹر دستیاب ہیں۔ان کو تلاش کرنے اور استعمال کرنے کی کوشش کریں۔ان کو تلاش کرنے میں آپ اپنے سکول ٹیچر کی مدد لے سکتے ہیں۔

2.2.2 اعشاری سے ہیگزاڈیسمیل (Hexadecimal) اور ہیگزاڈیسمیل (Hexadecimal) سے اعشاری

اعشاری سے ہیگزاڈیسمیل

جیسا کہ ہم نے پڑھا ہے کہ ہیگزاڈیسمیل نمبر سسٹم میں بیس سولہ (16) ہوتی ہے لہٰذا کسی نمبر کو اعشاری سے ہیگزاڈیسمیل میں تبدیل کرنے کے لیے ہم

اس نمبر کو 16 سے تقسیم کرتے ہیں اور حاصل تقسیم باقی اور باقی (remainder) کو لیتے ہیں اور اس طرح ہم حاصل تقسیم کو 16 سے تقسیم کرنے کا عمل جاری رکھتے ہیں جب تک حاصلِ تقسیم (Quotient) '0' کے برابر ہو جائے۔

مثال:

$(69610)_{10}$ کو ہیگزاڈیسمیل میں تبدیل کریں۔

16	69610	
16	4350	– A
16	271	-- E
16	16	-- F
16	1	-- 0
16	0	-- 1

Table 2-3

ٹیبل (2.1) میں اس مسئلے کو حل کرنے کا طریقہ دکھایا گیا ہے۔ ہم ٹیبل سے دیکھ سکتے ہیں کہ باقی 10 کو 'A' سے ظاہر کیا گیا ہے باقی (14) کو 'E' سے ظاہر کیا گیا ہے اور باقی (remainder)(15) کو 'F' سے ظاہر کیا گیا ہے۔

ہیگزاڈیسمیل نمبر میں دکھانے کے لیے باقی کو نیچے سے اوپر کی طرف لے جایا جاتا ہے۔ لہٰذا $(10FEA)_{16}$ = $(69610)_{10}$

ہیگزاڈیسمیل سے اعشاری نظام میں تبدیلی:

ماسوائے بیس (Base) کی قیمت کے تبادلے (conversion) کا یہ طریقہ، ثنائی سے اعشاری نظام میں تبدیل کرنے کے طریقے کی طرح ہے صرف (Base) کی قیمت چونکہ ہیگزاڈیسمل کی بیس 16 ہے اس لیے "place values" 16 کی پاور (Power) سے ضرب دی جاتی ہے۔ اعشاری میں تبدیل کرنے کے لیے "Place Value" کو 16 کی طاقت پاور کے مطابق ضرب دیں، اس عمل کا آغاز ہیگزاڈیسمیل نمبر کے ہندسوں کے آگے 16 کا عدد اور اس سے متعلقہ طاقت لکھ کر کریں۔

مثال: $(C921)_{10}$ کو اعشاری میں تبدیل کریں۔

سرگرمی: 2.5

اس کا حساب کرنے کی کوشش کریں کہ $(C921)_{16}$ کی ثنائی قیمت 11001001 ہے۔

2.2.3 ہیگزاڈیسمیل سے ثنائی میں اور ثنائی سے ہیگزاڈیسمیل میں تبدیلی

ہیگزاڈیسمیل سے ثنائی:

ہیگزاڈیسمیل نمبر کو ثنائی نمبر میں تبدیل کرنے کے لیے ہیگزاڈیسمیل نمبر کو 4 ہندسوں والی ثنائی (Binary) قدروں میں تبدیل کریں۔ 4 ہندسوں والی ثنائی قدروں تلاش کرنے کے لیے دیکھیں ٹیبل 2.4

مثال:

$(A23)_{16}$ (ہیگزاڈیسمل) کوثنائی میں تبدیل کریں۔

دیے گئے نمبر میں تین ہیگزاڈیسمل ہندسے ہیں۔ ہر ہندسے کی ثنائی قیمت دی گئی ہے:

i۔ 'A' کے لیے،ثنائی قیمت 1010 ہے۔

ii۔ 2 کے لیے،ثنائی قیمت 0010 ہے۔

iii۔ 3 کے لیے،ثنائی قیمت 0011 ہے۔

ان ساری ثنائی قیمتوں کوملانے سے ہمیں 101000100011 حاصل ہوتی ہے۔لہٰذا

$(101000100011)_2$

Hexadecimal	Binary
0	0000
1	0001
2	0010
3	0011
4	0100
5	0101
6	0110
7	0111
8	1000
9	1001
A	1010
B	1011
C	1100
D	1101
E	1110
F	1111

ٹیبل 2-4

مثال:

$(70C558)_{16}$ (ہیگزاڈیسمیل) کوثنائی میں تبدیل کریں

دیے گئے نمبر میں،6ہیگزاڈیسمیل ہندسے ہیں اور ہر ہندسے کی ثنائی قیمت مختلف رنگ میں دی گئی ہے:

i۔ 7 کے لیے،ثنائی قیمت 0111 ہے۔

ii۔ 0 کے لیے،ثنائی قیمت 0000 ہے۔

iii۔ 'C' کے لیےثنائی قیمت 1100 ہے۔

iv۔ 5 کے لیے،ثنائی قیمت 0101 ہے۔

v۔ 5 کے لیے،ثنائی قیمت 0101 ہے۔

vi۔ 8 کے لیے،ثنائی قیمت 1000 ہے۔

ان ساری ثنائی قیمتوں کو ملانے سے ہمیں 011100001100010101011000 حاصل ہوتی ہے لہٰذا

$(70C558)_{10} = 011100001100010101011000$

ثنائی سے ہیگزاڈیسمیل:

ٹیبل 2.4 کی مدد سے ایسی تبدیلی کرنا بھی آسان ہے ہم دیے گئے ثنائی (Binary) نمبر کو دائیں سے جانب 4 ہندسوں کے گروپوں میں تقسیم کرتے ہیں اور ہر گروپ کو ہیگزاڈیسمل نمبر سے تبدیل کر دیتے ہیں۔

مثال: $(11000001)_2$ کو ہیگزاڈیسمیل میں تبدیل کریں۔

اوپر دیے گئے ثنائی نمبر میں 4 ہندسوں کے گروپوں میں تقسیم کیا ہے:

i۔ 0001 کے لیے، '1' ہیگزاڈیسمیل ہے۔

ii۔ 1100 کے لیے، 'C' ہیگزاڈیسمیل ہے۔

لہٰذا $(11000001)_2 = (C1)_{16}$ دائیں سے بائیں جانب گروپ بنائے ہوئے اگر انتہائی بائیں گروپ میں ثنائی ہندسے 4 سے کم ہوں تو ہم زیرو کو بائیں جانب لگائیں گے مثال کے طور پر 1010011 میں 0011 اور 101 گروپ بنتے ہیں اس طرح ہم زیرو کو بائیں طرف لگاتے ہیں اور اس کے نتیجے میں 0101,0011 ہندسے بن جائیں گے۔

مثال: $(110101111)_2$ کو ہیگزاڈیسمل میں تبدیل کریں۔

اوپر دیے گئے ثنائی نمبر سے بننے والے گروپوں کو ذیل میں دیا گیا ہے جہاں ہر گروپ زیادہ سے زیادہ 4 ثنائی ہندسے ہیں۔ 11010 1111

انتہائی بائیں جانب گروپ جس کو نیلے رنگ میں دکھایا گیا ہے میں صرف ایک ثنائی ہندسہ ہے اس میں بائیں طرف 0 لگانے سے ہمیں مندرجہ ذیل نمبر حاصل معلوم ہوتا ہے۔

0001 1010 1111

ہم ہر گروپ کو اس سے متعلقہ ہیگزاڈیسمیل نمبر سے تبدیل کرتے ہیں اور ہمیں درجہ ذیل نمبر حاصل ہوگا۔

1AF

$(110101111)_2 = (1AF)_{16}$

2.3 میموری (Memory) اور ڈیٹا سٹوریج (Data Storage)

2.3.1 میموری

کمپیوٹر میموری ایسا مادی آلہ ہے جو ڈیٹا کو محفوظ کرنے کے قابل ہو۔ بنیادی طور پر میموری کی درج ذیل دو اقسام ہیں:

1- وولاٹائل میموری (Volatile Memory)

2- نان وولاٹائل میموری (Non-volatile Memory)

شکل 2.3 میں میموری کی دونوں اقسام دیکھائی گئی ہیں۔ ذیل میں ہم ان اقسام کو تفصیل سے پڑھتے ہیں۔

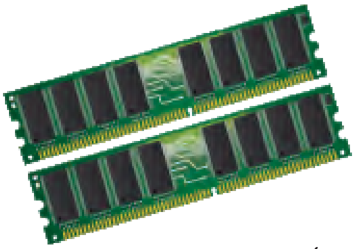
وولاٹائل/ پرائمری میموری (Volatile/Primary Memory)

یہ ایسا آلہ ہے جو اس وقت تک ڈیٹا محفوظ رکھتا ہے جب تک اسے بجلی کی فراہمی جاری رہے۔اس کی بہترین مثال ریم(RAM) ہے جو کہ اس وقت تک ڈیٹا محفوظ رکھتی ہے جب تک یہ بجلی سے منسلک رہتی ہے۔ جیسے ہی بجلی منقطع ہوتی ہے ریم میں محفوظ تمام ڈیٹا ضائع ہو جاتا ہے۔

نان وولاٹائل/سیکنڈری میموری (Non-Volatile/Secondary Memory)

ایسا آلہ جو ڈیٹا اس وقت بھی محفوظ رکھتا ہے جب یہ بجلی سے منسلک نہ بھی ہو۔''نان وولاٹائل''میموری کی عام مثالیں فلیش ڈرائیو اور میموری کارڈز ہیں۔ آپ کا کمپیوٹر اگر بند بھی ہو جائے تو اس قسم کے آلے میں ڈیٹا محفوظ ہی رہتا ہے۔

نان وولاٹائل میموری	بمقابلہ	وولاٹائل میموری



شکل 2.3 وولاٹائل بمقابلہ نان وولاٹائل

2.3.2 کمپیوٹر میموری میں ڈیٹا کی نمائندگی

ڈیجیٹل کمپیوٹر ڈیٹا کو بائنری(Binary) کی شکل میں محفوظ کرتا ہے۔اس سے مراد یہ ہوا کہ ڈیٹا چاہے یہ متن کی صورت میں ہو یا تصاویر کی شکل میں، فلم کی صورت میں ہو کسی ایپلیکیشن کی صورت میں ہو یہ کمپیوٹر کی میوری میں "0" اور "1" کی شکل میں ہی محفوظ ہوگا۔کی-بورڈ(Key Board) پر موجود تمام حروف کا بائنری کوڈ ہوتا ہے یہ کوڈ ان حروف کے''ASCII''کوڈ کہلاتے ہیں۔''ASCII'' دراصل American Standard Code for Information Interchage کا مخفف ہے۔یہ کمپیوٹر میموری میں ڈیٹا کی نمائندگی کے لیے ایک ڈی-فیکٹو سٹینڈرڈ (De-facto standard) ہے۔ٹیبل 2.5 کی-بورڈ پر موجود حروف کا ASCII کوڈ ظاہر کرتا ہے۔یہ کوڈ ڈیسیمیل کی شکل میں دکھائے گئے ہیں مگر کمپیوٹر میں یہ کوڈ بائنری میں تبدیل ہو کر استعمال ہوتے ہیں۔

Code	Character	Description	Code	Character	Description
32	**SP**	space	62	**>**	greater than
33	**!**	exclamation mark	63	**?**	question mark
34	**"**	double quote	64	**@**	"at" symbol
35	**#**	number sign	65	**A**	
36	**$**	dollar sign	66	**B**	
37	**%**	percent	67	**C**	
38	**&**	ampersand	68	**D**	
39	**'**	single quote	69	**E**	
40	**(**	left/opening parenthesis	70	**F**	
41	**)**	right/closing parenthesis	71	**G**	
42	*****	asterisk	72	**H**	
43	**+**	plus	73	**I**	
44	**,**	comma	74	**J**	
45	**-**	minus or dash	75	**K**	
46	**.**	dot	76	**L**	
47	**/**	forward slash	77	**M**	
48	**0**		78	**N**	
49	**1**		79	**O**	
50	**2**		80	**P**	
51	**3**		81	**Q**	
52	**4**		82	**R**	
53	**5**		83	**S**	
54	**6**		84	**T**	
55	**7**		85	**U**	
56	**8**		86	**V**	
57	**9**		87	**W**	
58	**:**	colon	88	**X**	
59	**;**	semi-colon	89	**Y**	
60	**<**	less than	90	**Z**	
61	**=**	equal sign	91	**[**	left/opening bracket

92	\	back slash	110	**n**	
93	]	right/closing bracket	111	**o**	
94	^	caret/circumflex	112	**p**	
95	_	underscore	113	**q**	
96	`		114	**r**	
97	**a**		115	**s**	
98	**b**		116	**t**	
99	**c**		117	**u**	
100	**d**		118	**v**	
101	**e**		119	**w**	
102	**f**		120	**x**	
103	**g**		121	**y**	
104	**h**		122	**z**	
105	**i**		123	**{**	left/opening brace
106	**j**		124	**|**	vertical bar
107	**k**		125	**}**	right/closing brace
108	**l**		126	**~**	tilde
109	**m**		127	**DEL**	delete

ٹیبل 2.5 ASCII

مثال:

اپنے پیارے وطن "Pakistan" کا نام کمپیوٹر میموری میں محفوظ کرنے کے لیے ہمیں ہر حرف کے کوڈ کے لیے ایک بائٹ کی ضرورت ہوتی ہے۔ چونکہ "Pakistan" میں 8 حروف ہیں اسی لیے اس کو محفوظ کرنے کے لیے 8 بائٹس (Bytes) درکار ہوتے ہیں۔ اِسے ٹیبل 2.6 میں دکھایا گیا ہے۔

Human's View about Memory	Code in Decimal	Code in Binary
'P'	80	1010000
'a'	97	1100001
'k'	107	1101011
'i'	105	1101001
's'	115	1110011
't'	116	1110100
'a'	97	1100001
'n'	110	1101110

ٹیبل 2.6

سرگرمی 2.6

اپنا مکمل نام لکھیں اور اس کی بائنری بنائیں۔

2.3.3 سٹوریج ڈیوائسز (Storage Devices)

کسی بھی قسم کا کمپیوٹر ہارڈویئر (Hardware) جو کہ ڈیٹا کو محفوظ کرنے یا ایک جگہ سے دوسری جگہ لے جانے کے لیے استعمال ہو، سٹوریج ڈیوائس کہلاتی ہے۔ یہ معلومات کو عارضی یا مستقل طور پر محفوظ کر سکتی ہے۔ یہ ڈیوائس کمپیوٹر کے اندر بھی لگی ہو سکتی ہے اور یہ کمپیوٹر سے باہر بھی ہو سکتی ہے۔ جو سٹوریج ڈیوائس کمپیوٹر سے باہر ہوتی ہے وہ اکثر پلگ اینڈ پلے ڈیوائسز (Plug and Play Devices) ہوتی ہے یعنی صرف اِن کو کمپیوٹر کے ساتھ منسلک کریں اور اِن کا استعمال شروع کر دیں جبکہ وہ سٹوریج ڈیوائسز جو کمپیوٹر کے اندر لگی ہوتی ہے اُن کو کمپیوٹر کے ساتھ منسلک کرنے کے لیے کمپیوٹر کو بند کر کے دوبارہ چلانا یعنی ری سٹارٹ (Restart) کرنا پڑتا ہے۔ انٹرنل سٹوریج ڈیوائسز کو مخصوص سلاٹس (Slots) میں ہی لگایا جاتا ہے۔

مثلاً ریم (Ram)، ہارڈ ڈیسک (HardDisk)، سی ڈی (CD)، یو ایس بی (USB) وغیرہ۔

میموری اور سٹوریج میں فرق

ٹیبل 2.7 میں میموری اور سٹوریج کا فرق بیان کیا گیا ہے۔

سٹوریج	میموری
یہ وہ جگہ ہوتی ہے جہاں عمومی طور پر ڈیٹا مختصر دورانیے یا طویل دورانیے کے لیے سٹور ہوتا ہے۔	یہ وہ جگہ ہوتی ہے جہاں پروسیسنگ (Processing) کے دوران ڈیٹا لوڈ ہوتا ہے۔
مستقل طور پر ڈیٹا کو محفوظ کرتی ہے۔	عارضی طور پر ڈیٹا کو محفوظ کرتی ہے۔
اِس کا سائز بڑا ہوتا ہے۔	اِس کا سائز کم ہوتا ہے۔
ڈیٹا تک رسائی کی سپیڈ کم ہوتی ہے۔	ڈیٹا تک رسائی کی سپیڈ زیادہ ہوتی ہے۔
اِس کو سیکنڈری سٹوریج بھی کہتے ہیں۔	اِس کو پرائمری سٹوریج بھی کہتے ہیں۔

ٹیبل 2.7 سٹوریج اور میموری میں فرق

2.4 کمپیوٹر میموری کے سائز کی پیمائش

کمپیوٹر میں کم سے کم جو ڈیٹا محفوظ کیا جا سکتا ہے وہ "0" یا "1" ہے اِس کو بِٹ (Bit) کہتے ہیں۔ 8 بِٹس کے مجموعے کو بائٹ (Byte) کہتے ہیں۔ کسی بھی قسم کی معلومات کو کمپیوٹر میں سٹور کرنے کے لیے کم سے کم ایک بائٹ جگہ درکار ہوتی ہے۔ پرائمری اور سیکنڈری سٹوریج ڈیوائسز میں ڈیٹا بائٹس کی صورت میں محفوظ کیا جاتا ہے۔ ٹیبل 2.8 میں مختلف ڈیٹا یونٹس دیے گئے ہیں۔

بِٹ (Bit)	سب سے چھوٹا سٹوریج یونٹ ہے صرف ایک ہی ویلیو سٹور کر سکتا ہے "0" یا "1"
بائیٹ (Byte)	8 بِٹس کے مجموعے کو بائٹ کہا جاتا ہے۔
کِلو بائیٹ (Kilobyte)	1Kb = 1024 Byte
میگا بائیٹ (Megabyte)	1Mb = 1024 Kb یا $(1024)^2$ Byte
گیگا بائیٹ (Gigabyte)	1Gb = 1024 Mb یا $(1024)^3$ Byte
ٹیرا بائیٹ (Terabyte)	1Tb = 1024 Gb یا $(1024)^4$ Byte
پیٹا بائیٹ (Petabyte)	1Pb = 1024 Tb یا $(1024)^5$ Byte

ٹیبل 2.8: ڈیٹا یونٹس

2.5 بولین الجبرا

2.5.1 بولین تجاویز/پری پوزیشن(Preposition)

پری پوزیشن ایک جملہ ہے جوکہ یا تو درست ہوسکتا ہے یا غلط،مثال کے طور پرمندرجہ ذیل بولین پری پوزیشن ہیں۔

1. ہمارے سکول میں سےکوئی پاکستان کرکٹ ٹیم میں جائے گا۔
2. میں''بورڈ'' کے امتحان میں +A گریڈ حاصل کروں گا۔
3. میں ریاضی میں مہارت حاصل کرنا چاہتا ہوں۔
4. اِس سال پاکستان سُپر لیگ کا فائنل میچ لاہور میں کھیلا جائے گا۔

مندرجہ ذیل جملے پری پوزیشن نہیں ہیں۔

1. ''آپ کیسے ہیں؟''
2. ''دروازہ بند کردو''

ہم پری پوزیشنز سے کئی لیٹر(حروف) کوبھی منسوب کرسکتے ہیں،جیسا کہ۔

''میں شطرنج کھیلتا ہوں'' =P

''میں ریاضی میں ماہر ہوں'' =Q

اِب ہم P لکھیں گے تو اِس کا مطلب پری پوزیشن ''میں شطرنج کھیلتا ہوں'' ہوگا اور جب بھی Q لکھیں گے تو اِس کا مطلب پری پوزیشن ''میں ریاضی میں ماہر ہوں'' ہوگا۔

کیا آپ جانتے ہیں؟

درست اور غلط کو بولین قدریں بھی کہا جاتا ہے۔اور یہ آئیڈیا جارج بولی اپنی ایک کتاب''The Laws of Thought'' میں پیش کیا۔

2.5.2 ٹروتھ ویلیوز(Truth values)

پری پوزیشن درست یا غلط قدر کو ظاہر کرتی ہے اور اِنہی قدروں کو ٹروتھ ویلیوز کہا جاتا ہے۔یہ قدریں کسی پری پوزیشن کے درست یا غلط ہونے پر اِس سے منسوب کی جاتی ہے۔

مثلاً:

فرض کریں P=''اسلام آباد پاکستان کا دارالخلافہ ہے'' تو آپ اِس کو ٹروتھ ویلیو 'درست'' منسوب کرسکتے ہیں۔اب ہم ایک اور پری پوزیشن فرض کرتے ہیں۔

Q=''سورج مغرب سے نکلے گا۔''اس پری پوزیشن کی ٹروتھ ویلیو غلط ہوگی۔اب ہم ایک پری پوزیشن فرض کرتے ہیں۔R=''میں نے اپنا ہوم ورک مکمل کرلیا ہے'' تو اِس کی ٹروتھ ویلیو اُس شخص پر منحصر کرے گی جس پر آپ اُس کو لاگو کر رہے ہیں۔اگر اُس شخص نے اپنا ہوم ورک مکمل کیا ہوگا تو اِس کی ویلیو'' درست''یعنی (True) ہوگی اور اگر نہیں کیا ہوگا تو اِس کی ویلیو'' غلط''یعنی (False) ہوگی۔

2.5.3 لاجیکل اوپریٹرز(Logical Operator)

بعض اوقات ہم ایک سے زیادہ پری پوزیشنز کو ایک ساتھ لکھنا چاہتے ہیں۔اِس کو ہم کمپاؤنڈ پری پوزیشن بھی کہتے ہیں۔مثال کے طور پر اگر ہمارے پاس مندرجہ ذیل دو پری پوزیشن ہیں:

1. آج سوموار ہے۔
2. میں سکول میں ہوں۔

تو'' آج سوموار ہے اور میں سکول میں ہوں'' ایک کمپاؤنڈ پری پوزیشن کہلائے گی۔کسی بھی کمپاؤنڈ پری پوزیشن کی ٹروتھ ویلیو اس کی ہر ایک پری پوزیشن کی ٹروتھ ویلیو پر اور اُس لاجیکل اوپریٹر پر منحصر کرتی ہے۔جو اِن کو آپس میں ملانے یا جوڑنے کے لیے استعمال ہوتا ہے۔اوپر دی گئی مثال میں "AND" لاجیکل اوپریٹر کا استعمال کیا گیا ہے۔ذیل میں AND(اور)، OR(یا)،NOT(نہیں) اوپریٹرز کی وضاحت دی گئی ہے۔

AND اوپریٹر(.)

اگر ہم AND اوپریٹر کو استعمال کرتے ہوئے دو یا دو سے زیادہ پری پوزیشنز کو ملاتے ہیں تو کمپاؤنڈ پری پوزیشن اُسی صورت میں True یا درست ہوگی اگر تمام منسلکہ پری پوزیشن درست یا True ہوں تو۔AND(اور) اوپریٹر کو dot(.) اوپریٹر بھی کیا جاتا ہے۔ہم P AND Q کو P.Q بھی لکھ سکتے ہیں۔

OR اوپریٹر(+)

ہم دو یا دو سے زیادہ پری پوزیشن کو ملانے کے لیے OR(یا) اوپریٹر کا استعمال بھی کر سکتے ہیں۔جیسا کہ:

'' آج سوموار ہے یا میں سکول میں ہوں'' اگر OR(یا) اوپریٹر کا استعمال کرتے ہوئے کمپاؤنڈ پری پوزیشن بنائی گئی ہے تو اِس کی ٹروتھ ویلیو، True یا درست ہو جائے گی اگر کوئی ایک پری پوزیشن بھی درست ہو تو اِس کی ٹروتھ ویلیو اُسی صورت میں False یا غلط ہوگی جب تمام پری پوزیشن غلط ہوں۔اِس اوپریٹر کو'+' کی مدد سے بھی ظاہر کیا جاتا ہے۔

NOT اوپریٹر(-)

یہ اوپریٹر دو پری پوزیشنز کو ملانے کے لیے استعمال نہیں ہوتا۔بلکہ یہ کسی پری پوزیشن کی ویلیو کا الٹ کرنے کے لیے استعمال ہوتا ہے۔جیسا کہ فرض کریں کہ P='' آج سوموار ہے'' تو NOT(P) کا مطلب یہ ہوگا کہ'' آج سوموار نہیں ہے''اِس لیے NOT اوپریٹر کے استعمال سے

True ہمیشہ False میں اور False ہمیشہ True میں بدل جاتا ہے۔ اِس کو "-" شکل کا استعمال کر کے بھی ظاہر کیا جا سکتا ہے۔ جیسا کہ:

NOT(P) = -P

2.5.4 ٹروتھ ٹیبل

کوئی پری پوزیشن درست ہے یا غلط اِس کو جانچنے کے لیے عمومی طور پر ٹروتھ ٹیبل کا استعمال کیا جاتا ہے۔ اگر کسی پری پوزیشن پر کوئی لاجیکل اوپریٹر لگایا جائے تو اِس کی ٹروتھ ویلیو جانچنے کے لیے ٹروتھ ٹیبل کا استعمال زیادہ تر ہوتا ہے۔ ہم AND اور NOT اوپریٹر کو استعمال کرتے ہوئے ٹروتھ ٹیبل بناتے ہیں۔

AND اوپریٹر کے لیے ٹروتھ ٹیبل

P AND Q کے لیے 2.9 ٹیبل میں دیا گیا ہے۔

P= "آج بارش ہو رہی ہے"

Q= "آج اتوار ہے"

P AND Q= "آج بارش ہے اور آج اتوار ہے"

P	Q	P AND Q
T	T	T
T	F	F
F	T	F
F	F	F

ٹیبل 2.9

اگر دونوں پری پوزیشنز True یا درست ہوں گی تو ہی یہ کمپاؤنڈ پری پوزیشن True یا درست ہوگی۔ اِس کا مطلب ہوگا یہ آج اتوار ہے اور بارش ہو رہی ہے۔ اِس کو ٹیبل 2.9 کی پہلی قطار (Row) میں ظاہر کیا گیا ہے۔ اِس ٹیبل کی دوسری قطار (Row) یہ ظاہر کرتی ہے کہ "آج اتوار ہے اور بارش نہیں ہو رہی"، اِسی طرح تیسری قطار یہ ظاہر کرتی ہے کہ آج اتوار نہیں ہے اور بارش ہو رہی ہے۔

اور چوتھی قطار یہ ظاہر کرتی ہے کہ آج اتوار نہیں ہے اور بارش نہیں ہو رہی ہے۔

لہٰذا P AND Q اسی صورت میں درست ہوگا جب دونوں پریپوزیشنز یعنی P اور Q درست ہوں گی۔

OR اوپریٹر کے لیے ٹروتھ ٹیبل:

آیئے! پری پوزیشن P اور Q کے لیے OR اوپریٹرز کا ٹروتھ ٹیبل دیکھتے ہیں۔

P OR Q= "یہ اتوار ہے یا آج بارش ہو رہی ہے"

یہ کمپاؤنڈ پریپوزیشن اِسی صورت میں False ہو سکتی ہے اگر آج اتوار بھی نہ ہو اور بارش بھی نہ ہو رہی ہو۔ اِس کے علاوہ یہ ہمیشہ True یا درست رزلٹ دے گی جیسا کہ ٹیبل 2.10 میں دکھایا گیا ہے۔

P	Q	P AND Q
T	T	T
T	F	T
F	T	T
F	F	F

ٹیبل 2.10

NOTاوپریٹر کے لیےٹروتھ ٹیبل:

ہم ایک ایساٹروتھ ٹیبل بھی بناسکتے ہیں جس میں NOTاوپریٹرکواستعمال کیا گیا ہو۔یہ اوپریٹر پری پوزیشن کی ویلیو کو بدل دیتا ہے۔جیسا کہ ٹیبل 2.11 میں دیکھا جاسکتا ہے:

P	NOT (P)
T	F
F	T

ٹیبل 2-11

بولین ایکسپریشن(Boolean Expression)

ہم ایک سے زیادہ اوپریٹر کے استعمال کے لیےبھی ٹروتھ ٹیبل بناسکتے ہیں۔مثال کے طور پر اگر کمپاؤنڈ پری پوزیشن ''آج توار نہیں ہےاور بارش ہورہی ہے'' کے لیےٹروتھ ٹیبل بنانا چاہتے ہیں تو اِس کا مطلب ہے کہ پری پوزیشن NOT(P)اور پری پوزیشن Q کو AND اوپریٹر کا استعمال کرتے ہوئے ساتھ ملادیں گے۔یہ کمپاؤنڈ پراپوزیشن ٹیبل 2.12 میں دکھائی گئی ہے۔

P	NOT(P)	Q	NOT(P) AND Q
T	F	T	F
T	F	F	F
F	T	T	T
F	T	F	F

ٹیبل 2-12

2.5.5 بولین الجبرا کے قوانین:

مشکل سوالات کو آسان کر کے لکھنے میں بولین الجبرا ہماری مدد کرتا ہے۔بولین الجبرا کے چند قوانین کا ذکر نیچے کیا گیا ہے۔

قانونِ مُبادلہ (Commutative Law)

یہ قانون ہمیں بتاتا ہے کہ بولین الجبرا میں دو یا دو سے زیادہ پری پوزیشن کی ترتیب اہم نہیں ہوتی۔مثلاً:

A.B = B.A ویری ایبلز کو کسی ترتیب میں بھی "AND" کیا جاسکتا ہے

اور

A+B=B+A ویری ایبلز کو کسی ترتیب سے بھی "OR" کیا جاسکتا ہے

قانونِ تلازم (Associative Law)

اِس قانون کے مطابق اگر ایک ایکسپریشن کے گروپس کی ترتیب بدل دی جائے تو اِس کے رزلٹ پر کوئی فرق نہیں پڑتا۔ اِس قانون کا AND اور OR دونوں اوپریٹرز پر ایک جیسا اثر ہوتا ہے۔ جیسا کہ:

$$(A+B)+C = A+(B+C)$$

$$(A.B).C = A.(B.C)$$

ہم OR اوپریٹر کے لیے اِس قانون کی تصدیق ٹیبل 2.14 میں دیکھ سکتے ہیں۔ اِس ٹیبل میں دونوں کالم (A+B)+C اور A+(B+C) کی ایک جیسی مقداریں (Values) ہیں۔

A	B	C	A + B	B + C	(A+B)+C	A+(B+C)
F	F	F	F	F	F	F
F	F	T	F	T	T	T
F	T	F	T	T	T	T
F	T	T	T	T	T	T
T	F	F	T	F	T	T
T	F	T	T	T	T	T
T	T	F	T	T	T	T
T	T	T	T	T	T	T

ٹیبل 2.14

اسی طرح ہم AND اوپریٹر کے لیے قانونِ تلازم (Associative Law) کی تصدیق ٹیبل 2.15 میں ملاحظہ کر سکتے ہیں۔

A	B	C	A · B	B · C	(A · B) · C	A · (B · C)
F	F	F	F	F	F	F
F	F	T	F	F	F	F
F	T	F	F	F	F	F
F	T	T	F	T	F	F
T	F	F	F	F	F	F
T	F	T	F	F	F	F
T	T	F	T	F	F	F
T	T	T	T	T	T	T

ٹیبل 2.15

قانونِ تقسیمی (Distributive Law)

اِس قانون کے مطابق:

A.(B+C) = (A.B)+(A.C)

A+(B.C) = (A+B).(A+C)

اِس قانون کی تصدیق ہم ٹیبل 2.16 میں دیکھ سکتے ہیں۔

A	B	C	$B+C$	$A\cdot B$	$A\cdot C$	$A\cdot(B+C)$	$A\cdot B+A\cdot C$
F	F	F	F	F	F	F	F
F	F	T	T	F	F	F	F
F	T	F	T	F	F	F	F
F	T	T	T	F	F	F	F
T	F	F	F	F	F	F	F
T	F	T	T	F	T	T	T
T	T	F	T	T	F	T	T
T	T	T	T	T	T	T	T

ٹیبل 2.16

سرگرمی: 2.7

$A+(B\cdot C)=(A+B)\cdot(A+C)$ کی تصدیق ٹروتھ ٹیبل کی مدد سے کریں۔

ضربی اور جمعی ذاتی قانون (Identity Law)

اگر کسی ویری ایبل کو False کے ساتھ OR کیا جائے تو رزلٹ ہمیشہ اُس ویری ایبل کی قیمت کے برابر ہی ہوگا۔ ایسی طرح اگر کسی ویری ایبل کو True کے ساتھ AND کیا جائے تو بھی رزلٹ ویری ایبل کی قیمت کے برابر ہوگا۔ جیسا کہ:

A OR FALSE = A: ویری ایبل A کو False کے ساتھ OR کیا جائے تو رزلٹ A ہی ہوگا۔ اسی طرح

A AND True = A: ویری ایبل A کو True کے ساتھ AND کیا جائے تو رزلٹ A ہی ہوگا۔

2.5.6 لاجیکل ایکسپریشن:

جب ہم لاجیکل اوپریٹرز کو بولین پری پوزیشن پر لاگو کرتے ہیں تو یہ لاجیکل ایکسپریشن بنتی ہے۔ اگر کسی بولین ایکسپریشن پر لاجیکل اوپریٹرز کا اطلاق کر دیا جائے تو ہمیں لاجیکل ایکسپریشن حاصل ہوتی ہے۔ مثلاً (P OR Q), –(P OR Q) وغیرہ۔ ٹیبل 2.14، ٹیبل 2.15 اور ٹیبل 2.16 لاجیکل ایکسپریشن کا اطلاق کیا گیا ہے۔

کیا آپ جانتے ہیں؟

کسی منفی پری پوزیشن پر NOT اوپریشن لگانے سے ہمیں مثبت پری پوزیشن حاصل ہوتی ہے۔

مثلاً:

P= ''آج مطلع صاف ہے''۔

¬P = ''آج مطلع صاف نہیں ہے''۔

¬¬P = ''آج مطلع صاف ہے''۔

اسی طرح

Q= ''آج جمعہ نہیں ہے''۔

¬Q = ''آج جمعہ ہے''۔

¬¬Q = ''آج جمعہ نہیں ہے''۔

خلاصہ (Summary)

- بائنری لینگویج '0' اور '1' پر مشتمل ہوتی ہے اور کمپیوٹر صرف بائنری لینگویج کو ہی سمجھتا ہے۔
- ڈیسیمل سسٹم کی بیس (Base) دس (10) ہوتی ہے اور اس میں 0 سے 9 تک ہندسے ہوتے ہیں۔
- ہیگساڈیسیمل نمبر سسٹم میں 16 ہندسے ہوتے ہیں۔ یعنی 0، 1، 2، 3، 4، 5، 6، 7، 8، 9، F,E,D,C,B,A
- کمپیوٹر کی میموری ایک فزیکل ڈیوائس ہے جو کہ ڈیٹا کو عارضی یا مستقل طور پر محفوظ کرنے کی صلاحیت رکھتی ہے۔
- ایسی ڈیوائسیں جو ڈیٹا کو صرف اتنی دیر تک ہی محفوظ رکھتی ہیں جب تک بجلی کی فراہمی جاری رہے وولاٹائل ڈیوائسیں کہلاتی ہیں۔ یہ عارضی سٹوریج ڈیوائسز بھی کہلاتی ہیں۔
- ایسی ڈیوائسیں جو ڈیٹا کو تب بھی محفوظ رکھتی ہیں اگر بجلی منقطع بھی ہو جائے نان وولاٹائل ڈیوائسس کہلاتی ہیں۔ یہ مستقل سٹوریج ڈیوائسیں بھی
- کہلاتی ہیں۔
- بولین ویلیو یا تو درست (True) ہو سکتی ہے یا غلط (False)۔
- ٹروتھ ٹیبل کسی سٹیٹمنٹ کو درست (True) یا غلط (False) دکھانے کے لیے استعمال ہوتا ہے۔
- قانونِ مبادلہ

  A.B = B.A
- قانونِ تلازم

  (A+B)+C = (A+B).(A+C)
- قانونِ تقسیمی۔

  A+(B.C) = (A+B).(A+C)
- ضربی اور جمعی ذاتی قانون

  A+0 = A

  A.1 = A

EXERCISE

2.1 کثیرانتخابی سوالات۔

1۔ ایکسپریشن (A+B).(A+C)______________کے برابر ہوتی ہے:

(i) A + (B.C)
(ii) A .B + A.C
(iii) A. (B.C)
(iv) A+ (B+C)

2۔ ___________قانون میں ویری ایبلز کی ترتیب ضروری نہیں ہوتی:

(i) قانونِ تلازم
(ii) قانونِ مبادلہ
(iii) قانونِ تقسیمی
(iv) ضربی اور جمعی ذاتی قانون

3۔ ’’باہر سردی ہے‘‘ ایک______________ہے:

(i) بولین پری پوزیشن
(ii) مورل پری پوزیشن
(iii) دونوں (i) اور (ii)
(iv) کوئی بھی نہیں

4۔ بائنری سسٹم میں نمبر ’’17‘‘ ________کے برابر ہوتا ہے:

(i) 10000
(ii) 10110
(iii) 10001
(iv) 10100

5۔ پیٹا بائٹ__________کے برابر ہوتا ہے:

(i) $(1024)_4$ بائیٹ
(ii) $(1024)_6$ بائیٹ
(iii) $(1024)_5$ بائیٹ
(iv) $(1024)_7$ بائیٹ

6۔ ہیکسا ڈیسمل میں __________ نمبر ہوتے ہیں:

(i) 17
(ii) 16
(iii) 18
(iv) 15

2.2 درج ذیل سوالات کے جوابات دیں۔

1۔ $(69610)_{10}$ کو ہیکسا ڈیسمل میں تبدیل کریں۔

2۔ وولا ٹائل اور نان وولا ٹائل سٹوریج ڈیوائسیس میں فرق کریں۔

3۔ اپنے کمپیوٹر میں لفظ "Phone" ایڈریس 7003 پر محفوظ کریں۔ جبکہ ہر ایک حرف ایک بائٹ جگہ لیتا ہے کمپیوٹر پر میموری پر کس ایڈریس میں کونسا لفظ آئے گا؟

4۔ عارضی اور مستقل سٹوریج ڈیوائسیس میں فرق کریں۔

5- X AND Y کے لیے ٹروتھ ٹیبل بنائیں۔ جبکہ:

X= آج بارش ہے۔

Y= آج سوموار ہے۔

2.3 خالی جگہ پُر کریں۔

1- _____ عارضی سٹوریج ڈیوائس ہے اور _________ مستقل سٹوریج ڈیوائس ہے۔

2- پروسیس کرنے کے لیے ڈیٹا _________ کے ذریعے دیا جاتا ہے۔

3- کسی بھی انفارمیشن کو محفوظ کرنے کے لیے کم سے کم _________ بائٹ/بائٹس کی ضرورت ہوتی ہے۔

4- ایک سے زیادہ پرائپوزیشنز کو ایک ساتھ لکھنے سے _________ بنتی ہے۔

5- پرائمری اور سیکنڈری سٹوریج ڈیوائسز ڈیٹا _________ کی صورت میں محفوظ کرتی ہیں۔

2.4 مندرجہ ذیل کو تبدیل کریں۔

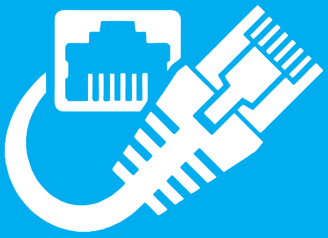
1- $(ABCD)_{16}$ کو بائنری میں۔

2- $(0010110010001101001)_2$ کو ہیگسا ڈیسمل میں۔

سرگرمی 2.8

اُستاد کلاس میں طلبہ کو ایک چارٹ دکھائے گا جس پر انگلش کے حروف اور اُن کا بائنری کوڈ لکھا گیا ہوگا۔ کلاس دو گروپوں میں تقسیم کریں اور اِن میں سے ہر ایک گروپ کوئی پانچ مشہور نام بائنری میں لکھے گا یہ مشہور نام تحریکِ آزادی کے عہد اداروں کے بھی ہو سکتے ہیں جیسا کہ ''مولانا محمد علی جوہر''۔ دونوں گروپ اپنے لکھے گئے ناموں کا ایک دوسرے کے ساتھ تبادلہ کریں گے اور اس حاصل شدہ ڈیٹا سے اصل نام (انگریزی حروف میں) لکھیں گے۔ گروپ جو پہلے یہ کام مکمل کرے گا فاتح قرار دیا جائے گا۔

یونٹ 3 نیٹ ورکس (Networks)



مختصر تعارف

اس یونٹ میں ہم کمپیوٹر نیٹ ورکس اور ڈیٹا کمیونیکیشن کے بنیادی تصورات کا مطالعہ کریں گے۔ ہم کمپیوٹر نیٹ ورک کے بنیادی اجزاء اور جغرافیائی انتظامات (geographic arrangement) کا مطالعہ کریں گے اور انٹرنیٹ پر استعمال ہونے والے کمیونیکیشن ماڈلز کا بھی تذکرہ کریں گے۔

حاصلاتِ تعلّم (Students Learning Outcomes)

1- نیٹ ورکس

- کمپیوٹر نیٹ ورکس کی وضاحت کرنا۔
- نیٹ ورک کی ساخت بیان کرنا۔
- کمپیوٹر نیٹ ورکس کی ضرورت اور اہمیت بیان کرنا۔
- کلائنٹ (Client) اور سرور (Server) کی وضاحت کرنا۔

2- کمیونیکیشن کی بنیادیں۔

- کمیونیکیشن سسٹم کے مندرجہ ذیل اجزاء کی وضاحت کرنا۔
  - پیغام بھیجنے والا/سینڈر (Sender)
  - پیغام وصول کرنے والا/رسیور (Receiver)
  - پروٹوکول (Protocol)
  - ٹرانسمیشن میڈیم/ذرائع ابلاغ

3- نیٹ ورک ماڈلز کی تقسیم

- TCP/IP کی وضاحت کرنا۔
- TCP/IP کی لیئرز (Layers) کی وضاحت کرنا۔

4- ڈیٹا کمیونیکیشن کی بنیادیں

- اس بات کی وضاحت کرنا کہ پیغام کو ایڈریسنگ (Addressing) کی ضرورت کیوں ہے؟
- ٹیلی فون یا ڈاک کے سسٹم میں ایڈریسنگ (Addressing) کی اہمیت بیان کرنا۔
- ریکوسٹ/رسپونس (Request/Response) میکانزم کی وضاحت کرنا۔
- IP ایڈریسنگ کی وضاحت کرنا۔

5- TCP/IP سوٹ (Suit) میں پروٹوکول۔

- HTTP، FTP اور SMTP پروٹوکول کی وضاحت کرنا۔

6- ایڈریسنگ کی ضرورت:

- ڈیٹا کمیونیکیشن میں ایڈریسنگ کی اہمیت جاننا۔
- ٹیلی فون اور ڈاک میں ایڈریسنگ کی وضاحت کرنا۔

7- HTTP ریکویسٹ اور ان کا ریسپونس (Response)

- IPv4 اور IPv6 میں فرق کرنا۔

8- روٹنگ (Routing)

- روٹر (Router) کے فنکشن جاننا۔
- روٹنگ کا طریقہ کار بیان کرنا۔