


$$(3)_2$$

11

$$(8)_9 = 10$$

A simple hand-drawn smiley face with two dots for eyes and a curved line for a mouth.

8) $\sqrt{12}$

$(10)_7 = \overline{1010} \mid \overline{100} \mid \overline{1000}$

BNS $\rightarrow 0, 1, 2$

ONS \rightarrow 7/0

DNS $\rightarrow 0,9$ (15)

HDN₅ → 16 characters
(0-9-a-z-f)

~~→ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, a, b, c, d, e, f~~

computer gives address by hexadecimal no. system.

8 GB RAM \rightarrow $(8 \times 1024 \text{ MB})$
↓

1024 kb.

1024 b.

$8 \times 2^{10} \times 2^6 \times 2^{10}$
 8×2^{30} bytes

12233 = 34 characters

2

1 < 33 = 57

octal 15-25 characters

decimal 12-29 characters

Hexadecimal 9-12 characters

o

a, b, c, d, e, f
10 15

16 ⇒ 10 11

20 21

.

af

ff

ft

ff

ff

10 10

10

10

hexadecimal

0x17

Hexadecimal: (0x12)

Now, to store these addresses we have pointers.

pointers store the addresses of different variables.

Syntax

char = 'A'

bool → true/false

int → 4 bytes

float → 4 bytes

double → 8 bytes

~~DNS~~

~~Decimal numbers~~

(X)

↓
Hexadecimal numbers.

233

(int) a = 10;

float b;

bool c;

double d;

Syntax for pointers.

...) * (boinkuska ucan)

Syntax for +
(datatype of variable) * (pointer ka naam)
= & variable

syntax

int * aptr = &a;
float * bptr = &b;

logical AND
'&&'

Bitwise AND
'&'

referencing
int &y = n;

address
&
same concept