



① int → 4 bytes  $a = 10$  → 28 bits  
② float → 4 Bytes  
③ char → 1 Byte  
④ double → 8 Bytes  
⑤ boolean → 1B || 2B

① Derived  
① Arrays  
② Functions  
③ Pointers

① Derived  
① Arrays:  
→ Coln of similar data type.

① Declaration static size  
① a [5]  $\rightarrow \{1, 2, 3, 4, 5\}$   
② a [ ] = {1, 2, 3, 4, 5, 6}

② Dimension  
① array size of A  
② a [2] [c] →



Union

float b = 10.2  
 int a = 10

$\Rightarrow a = 10$

$\Rightarrow b = 10.2$

$\Rightarrow a =$

0 Operator:  
+ - \* / %

① Arithmetic      ② Relation      ③ Comparison  
 $\rightarrow \ll \gg < > = <= >=$        $\rightarrow 11 \% 3 \quad 11 / 3$

④ Logical      ⑤ Assignment      ⑥ Ternary  
 $\rightarrow \& \& \neg$        $\rightarrow == !=$        $\rightarrow a \oplus b$       operand  
 $\rightarrow \&= \&= \neg=$        $\rightarrow == \neq$       operand |  $(+)$  open )

⑦ Ternary      ⑧ Shift  
 $\rightarrow ?: :$        $\rightarrow \ll \gg \lll \ggg$

$\equiv \equiv$  only check values       $\equiv \equiv \equiv$  determine types & values as wp

etc. -

Binary Addition:

$$\begin{array}{r}
 00010101 \\
 + 00111010 \\
 \hline
 00101111
 \end{array}$$

Bitwise Operators:

- AND ( $\&$ )
- OR ( $^$ )
- NOT ( $\sim$ )
- Left Shift ( $ll$ )
- Right Shift ( $rr$ )
- XOR ( $\wedge$ )

Ternary (?)

$a = 10, b = 20;$

$$x = \begin{cases} A & : B \\ C & : D \end{cases}$$

Cond

$2 \gg 1$

$001 = 1$

⑧ Shift operator

⑨ leftshift ( $<<$ )

⑩ rightshift ( $>>$ )

2  $<< 1$

010  $\rightarrow 2$   
100  $\rightarrow 4$

Unary operators

① increment

pre post

$++a$   $a++$

② decrement

pre post  
 $--a$   $a--$

Variables

2-types

① local      ② global

```

int add(a,b) {
    a = 10;           → local variable
    b = 20;
    c = a+b;
    printf("%d",c);
}

int main() {
    int x;
    x = add(10,20);
    printf("%d",x);
}

```

int add(a,b) →  
 return a+b;  
 ↴  
 int main() {  
 int a = 10; b = 20;  
 d = a+b;  
 printf("%d",d);  
 }

a = 10 , b = 20

$a = a + b \Rightarrow 10 + 20 = 30$

$b = a - b \Rightarrow 30 - 20 = 10 \quad 2$