

TYPES, OPERATORS AND EXPRESSIONS

Data Types in C

- Integral types – `int`, `char`
- `float`, `double`
- Modifiers: `short`, `long`, `long long`
- Others: `unsigned`, `signed`
- Constants – `int` (decimal, octal, hexa), `char`, `float`, `double`, string
- NUL character `\0` (ASCII value of 0)
- String constants are different from character constants, e.g., `'A'` (ASCII value of 65) versus `"A"`

Declarations

- All variables must be declared before use.

```
int N, marks, total, count;  
float avg;
```

- A variable may also be initialized in its declaration.

```
int N, marks, total = 0, count;  
float avg = 0.0;
```

Operators

- Can be unary, binary or ternary
- Used to perform some operation (e.g., arithmetic, logical, bitwise, assignment) on operands
- In an expression with multiple operators, the order of evaluation is based on the **precedence** level
- Operators with the same precedence work by rules of **associativity**

More on Operators

- Arithmetic (* / % and + -)
- Relational (< <= > >=)
 - Value of a relational expression is 1 if the relation is true, and 0 if the relation is false
- Logical (&& and ||)
 - Value of a relational expression is 1 if the relation is true, and 0 if the relation is false
 - Short-circuit evaluation

```
char ch = getchar();  
if (ch >= 'A' && ch <= 'Z')  
    putchar(ch);  
else  
    putchar(ch);
```

Assignment and Conditional operators

- Assignment operators have low precedence, just over the comma operator

```
char ch;  
if (ch = getchar() != '\n')  
    putchar(ch);
```

has a different meaning and effect from the intended

```
if ((ch = getchar()) != '\n')  
    putchar(ch);
```

- Ternary operator - ? :

Increment and Decrement operators

- ++ and --
- Can be used either as prefix or postfix
 - Prefix – “Change value and then use”
 - Postfix – “Use and then change value”
- Can be applied only to variables
- Cause side effects

Type conversions

- Implicit
 - If either operand is long double, convert the other into long double.
 - Otherwise, if either operand is double, convert the other into double.
 - Otherwise, if either operand is float, convert the other into float.
 - Otherwise, convert char and short to int.
 - Then, if either operand is long, convert the other to long.
- Explicit

Important considerations

- C does not specify the order in which the operands of an operator are evaluated.
 - Exceptions: `&&` `||` `?:` `,`
- Side effects
 - Six operators generate side effects: prefix increment and decrement, postfix increment and decrement, assignment and function call