Programming Basics - II

# cout → print/display

```
cout << a ;
```

# cin → takes input from user.

```
cout << "enter value ";
cin >>num;
```

// → comment, which improves readability.

# If → used to check a condition

```
if (condition)
  {        }
```

# If else statement → If condition is true, execute first block of code, else execute second block of code.

```
if (condition)
  {  ____
     ____
        }
else
  {
     ____
     ____
      }
```

# if else if → if one condition is true we ~~come out of the~~ execute that block and other blocks are not executed.

```
if (condition)
  { ____
    ____
      }
else if (condition)
  { ____
    ____ }
else { ____ }
```

if true, other blocks will not be executed else, next condition will be checked.

If true, again this block is executed & next one not executed!

Else next cond. is checked.

H/W → Explore if else if, how you can use nested statements & use?

Switch Case:- Expression ke output ke basis pe execute krte h.

```
switch (num)
{
  case 0 : ====
           break;
  case 1 : ====
           break;
  case 2 : ====
           break;
  default : ====
}
```

* If a case gets matched, execute statements under it and come out of the switch using "break".
* If no break → all cases will be executed which is unnecessary.

* If none of the cases match, default case is executed.

Example:-

```
switch (op) {
    case '+' : cout << a+b << endl;
               break;
    case '-' : cout << a-b << endl;
               break;
    case '*' : cout << a*b << endl;
               break;
    case '/' : cout << a/b << endl;
               break;
    default : cout << "default case" << endl; }
```

M T W T F S S
Page No.:
Date:
YOUVA

H/W Which is good over other → if or switch? And why?

Arguments valid for switch?

Ans: Integer, character, or mathematical expression symbol. eg switch(1) { }

switch(a+2*b) { }

* If switch is left empty without argument ie switch( ), it creates an error of "expected expression".
{ }

H/W Explore different arguments of switch such as switch(0/0) { }, etc.

Loops :- used for repetition of statements

① while Loop → while (cond^n)

```
while (cond^n)
  {  ====
  }
```

Statements will be executed until condition is true, else come out of loop.

H/W ① Print your name n times (using while loop).
② I/P → n
O/P → (1+2+3+4+ ..... +n)

For loop –

initialisation, condition, updation

```
for (int i=0 ; i<n ; i=i+1)
  {
    ===
  }
```

i++ vs ++i

post increment → (first use, then increment)

pre increment (increment first then use)

int i=5 cout << i++; output → 5 (value printed first then incremented)	int i=5 cout << ++i; output → 6 (value first incremented then printed)

Break:

```
for (int i=1; i<=n; i++)
  {
  cout << 2*i <<endl;
  //break;
  }
```

← If you include break in loop, you will come of the loop ie come out of the scope.

Continue:

```
for (——, ——, ——)
{ if (——) → true
   continue;
   ——
   ——
   ——
  }.
```

Continue is used to skip rest of the statements of the loop & jump to next iteration ie condition.

H/W. Explore: Do while. When to use this??

Important for MCQs → break, continue.
pre/post → inc/dec.

Ques- In for loop → $\text{for}(\underbrace{\text{int } i=1}; \underbrace{i<=n}; \underbrace{i++})$ { }

All conditions are optional.
Infinite loop will be created → for(; ;)
{ }

H/W. Play with for(;;){ } loop.

Variable Scope:

```
for(int i=0; i<n; i++)
 {
   cout<<i<<endl;
   }
  i=i+2;
```

Scope of i is limited to this loop and not outside it

this is undeclared and has no scope outside the loop.

error of "undeclared identifier"

eg

```
int main()
  {
    int a =5;
    int a =6;
```

produce error of "redefinition of a" since we can't declare same variable two times.

```
if true {
        int n =5;
        cout<< n <<endl;
        if(true) {
            int n =6;
        cout<< n <<endl;
          }
      }
```

here only n=5 will be accessible

here both n are accessible

This prints 6 because the rule says "print the closest one."

H/W Explore → Google → Variable scoping MCQs(10)
pre/pos inc/dec MCQs(10)
Break/continue MCQs(10)
switch/if/if else MCQs(10)