by jusnaid bsse 007(03457448497)comsats sahiwal.

***Syntax of all topics…***

# initialization the variable datatype space variablename=value of variable;

eg int marks=100; bool a=flase;

int marks=10,age=20; float per=55.5;

# declaring the variable.

datatype space variable name;

eg int marks; float percentage; bool A;

# increment/decrement operator.

variable name++/--; is a postfix or ++/--variable name; is a prefix

eg a++/--; ++/--a;

# compound assignment operators.

VariableNameArithmaticOperatersAssignmentOpertersValue;

e.g. a + = 20;

if int a=40;

a+=20 a=60

a-=20 a=20

a\*= 20 a=800

# comment. single line & multiple line.

e.g. a=3.14\*r\*; //this is formula for area pi r square.

/ aaaaaaaaa

agggcgdchfhvjvjjjj /

# manipulator.

eg endl and setw

# Escape sequence.

e.g. \n \a \t

### logical operators

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Condition no 1 | Condition no 2 | AND && | OR || | NOT OPERTER ! |
| True | True | Out put | Out put | (!(condition)) |
| True | False | xxxxxxxxxx | Out put | Not equal to != |
| False | True | xxxxxxxxxx | Out put |  |
| False | False | xxxxxxxxxx | xxxxxxxxxx |  |

# control staments.

1. sequence statement 2 selection statements 3 iteration statements 4 function call

### 2 Selection statements.

### If selection statement.

if(condition)

{statements;}

### If-else statement.

if(condition)

{statements;}

else

{statements;}

### Nested if-else

if(condition)

{statements;}

else if (condition)

{statements;}

else if (condition)

{statements;}

else

{statements;}

### Nested if statements

if(condition)

if(condition)

{statements;}

else

{statements;}

else

{statements;}

**Alternate of if else statements.**

(condition? true statements : false statements);

1. **Iteration statement.**

(1) for loop (2) while loop (3) do while loop

**(1) for loop**

for(initialization;condition;inc/dec operters)

eg for(int j=1;j<5;j++/--)

**(2) While loop**

while(condition)

{statements;}

**(3) Do while loop**

do

{statements;}

while(condition);

**(4 ) function call**

# (1) Defining the function

return type space funtionname(parameters)

{statements;}

# (2) Declaring the function

return type space funtionname(parameters);

# (3) Calling the function

funtionname(Areguments);

**switch statements**

switch(expression or variables)

{

case val-1: //use colon not semi colon

statement 1;

break;

case val-2: //use colon not semi colon

statement 2;

break;

-----------

-----------

case val-n: //use colon not semi colon

statement n;

break;

default:// use colon not semi colon

statements;

}