Operator Precedence and Associativity

- Operator precedence is a predefined rule of priority of operators. If more than one operators are involved in an expression the operator of higher precedence is evaluated first.
- Associativity indicates the order in which multiple operators of same precedence executes.

Precedence	Operator	Associativity
1	(), {}	Left to right
2	++, --, !	Right to left
3	*, /, %	Left to right
4	+, -	Left to right
5	<, <=, >, >=	Left to right
6	== ,!=	Left to right
7	&&	Left to right
8	||	Left to right
9	?:	Right to left
10	=, *=, /=, %=, -=	Right to left

Program

1) **What are the final value of a, b, z and x after execution of the following statements.**

```
int x=10,a=10,b=11,z=12;
 x+=++a-b++;
 x-=++z%--b;
 x*=++a+b++*z--;
```

2) **What will be the output of the program.**

```
 int m=100,n=50;
printf("%d\n",10+m++);
 printf("%d\n",10+n--);
 printf("%d\n",10+--n);
```

3) **What are the final values of x, y and z**

```
int main()
{
 int x=16,y=18,z;
 y+=x++;
z=++x+y++;
x=x+y+z--;
printf("\nx=%d\ny=%d\nz=%d ",x,y,z);
```

}

4) What will be the output of x, y, z and i in the following program

```
int main()
{
 int x=10,y=20,z=5,i;
 i=x*(++y+z++)%3+y/x*2-5;
 x+=z;
 y-=z;
 x--;
 z+=10;
 z--;
 printf("\nx=%d\ny=%d\nz=%d",x,y,z);
 printf("\ni=%d",i);
}
```

5) What will be the output of x, y, z and k in the following program

```
int main( )
{
 int x=10,y=20,z,k;
 z=++y/x--;
 y+=10;
 x+=20;
 y--;
 x++;
 z*=15;
 k=(x==y?(x+y):z);
 printf("\nx=%d\ny=%d\nz=%d\nk=%d",x,y,z,k);
}
```

6) Find the value of k

```
 int i=4,j=6,k=7;
 k+=(5+i*j%7)*6/(i+j)-3;
```

7) Find the value of x, y, z ,w, p and q

```
int main()
{
 int x=10,y=5,z,w=9,p,q;
 x+=10;
 y=x++;
 z=--x;
 w/=x;
 p=x+y;
 q=p+w;
 z=p++;
 }
```

8) Given the code below write the output of the following program

```
#include<stdio.h>
#include<conio.h>
int main()
{
 int x,y,z;
 x=3,y=4,z=5;
 printf("Given x=%d,y=%d and z=%d\n",x,y,z);
 x=++x+y;
 printf("x=x+y assigns %d to x\n",x);
 x=6;
 x+=y;
 printf("x+=y assigns %d to x,\n",x);
 x=2;
 z=z*x+y++;
 printf("z=z*x+y assigns %d to z\n",z);
 z=10;
 z=z*(x+y);
 printf("z=z*(x+y)assigns %d to z\n",z);
 z=10;
 z*=x+y;
 printf("z*=x+y assigns %d to z\n",z);
 z=10;
 z*=x+y;
 printf("z*=x+y assigns %d to z",z);

}
```

9) Find the output of the following program

```
int main()
{
 int x=100;
 printf("%d\n",10+x++);
 printf("%d",10+ ++x);
}
```

10) Evaluate

i) k=3*4/5+5/5+4-1+6/8

ii) m=x/2+x*4/x –x +y/3 where y=1.5,x=3 and assume m to be float

iii) i=j/2*3+3/8+j*j*j%4 if j=4

++x ⇒ x = x+1 (pre increment)

x++ ⇒ x = x+1 (post increment)

--x ⇒ x = x-1 (Pre decrement)

x-- ⇒ x = x-1 (post decrement)

1) y = x++;

y = x

x = x+1

eg. x=5; y;

y = x++

y = 5

x = 5+1 = 6

2) y = ++x

x = x+1

y = x

eg. x=5; y;

y = ++x

x = x+1 = 6

y = 6

3) y = x--

y = x;

x = x-1;

eg. x=5; y;

y = x--

y = 5

x = x-1 = 5-1 = 4

4) y = --x

x = x-1

y = x

eg. x=5; y;

y = --x

x = x-1 = 4

y = 4

```
int main()
{
int a=10;
printf("a=%d", a);
printf("a=%d", ++a);
printf("a=%d", a++);
printf("a=%d", a);
getch();
return 0;
}
```

O/P

a=10
a=11
a=11
a=12

a
10
11
12

$x=5, y$

$y = x++ + x++$

$y = x+x \Rightarrow 5+5 = 10$

$x = x+1 = 5+1 = 6$

$x = x+1 = 6+1 = 7$

$\therefore x=7, y=10$

$x=5, y$

$y = ++x + x++$

$y = x+x = 6+6 = 12$

$x = x+1 = 6+1 = 7$

$\therefore x=7, y=12$

$x=5, y$

$y = x++ + ++x$

$x = x+1 = 5+1 = 6$

$y = x+x = 6+6 = 12$

$x = x+1 = 6+1 = 7$

$\therefore x=7, y=12$

$x=5, y$

$y = ++x + ++x$

$x = x+1 = 5+1 = 6$

$x = x+1 = 6+1 = 7$

$y = 7+7 = 14$

$\therefore x=7, y=14$

a) int x = 2;
x++;
printf("value of x is %d", x);
o/p 3

b) int x = 2, y;
y = x++;
Printf("value of x = %d & y = %d", x, y);
o/p
x = 3, y = 2

c) int x, y;
x = 2;
y = ++x;
Printf("x = %d & y = %d", x, y); o/p
x = 3, y = 3

d) int x = 2;
printf("value of x = %d", x++); 2
printf("value of x = %d", x); 3

e) int x = 2;
printf("value of x = %d", ++x); value of x = 3
printf("value of x = %d", x); value of x = 3

Program:

1) What are the final values of a, b, z and x after execution of the following statements.

```
int a=10, a=10, b=11, z=12;
x += ++a -b++;
x -= ++z % --b;
x *= ++a +b++ * z--;
```

a	b	x	z
~~10~~	~~11~~	~~10~~	~~12~~
~~11~~	~~12~~	~~10~~	~~13~~
12	~~11~~	8	12
	12	1240	

x += ++a - b++;

The equivalent expression is,

a = a+1 = 10+1 = 11

x = x+(a-b)
= 10 + (11-11)
= 10

b = b+1 = 11+1 = 12

x -= ++z % --b;

The equivalent expression is,

b = b-1 = 12-1 = 11
z = z+1 = 12+1 = 13

x = x - (z % b)
= 10 - (13 % 11)
= 10 - 2
= 8

x *= ++a + b++ * z--;

The equivalent expression is,

a = a+1 = 11+1 = 12

x = x * (a + b * z) = 8 * (12 + 11*13) = 8 * 155 = 1240
z = z-1 = 13-1 = 12
b = b+1 = 11+1 = 12

∴ a = 12
b = 12
~~x~~ z = 12
x = 1240

2) What will be the output of the program

```
int m=100, n=50;
printf("%d\n", 10+m++);
printf("%d\n", 10+n--);
printf("%d\n", 10+--n);
```

Soln:

m: 100 → 101

n: 50 → 49 → 48

Evaluating first statement,

10+m = 10+100 = 110

m = m+1 = 100+1 = 101

Evaluating second statement,

10+n = 10+50 = 60

n = n-1 = 50-1 = 49

Evaluating third statement

n = n-1 = 49-1 = 48

10+n = 10+48 = 58

Output

110
60
58

What are the final values of x, y and z.

```
int main()
{
int x=16, y=18, z;
y+= x++;
z = ++x +y++;
x = x+y+z--;
printf("\n x=%d\n y=%d \n z=%d", x, y, z);
}
```

x	y	z
~~16~~	~~18~~	~~52~~
17	34	51
18	35	

1) y+= x++;

The equivalent expression is

y = y+x ⇒ 18+16 = 34

x = x+1 ⇒ 16+1 = 17

2) z = ++x + y++;

The equivalent expression is

x = x+1 ⇒ 17+1 = 18

z = x+y ⇒ 18+34 = 52

y = y+1 ⇒ 34+1 = 35

3) x = x+y+z--;

x = x+y+z = 18+35+52 = 105

z = z-1 = 52-1 = 51

output

x = 105

y = 35

z = 51

4) What will be the output of x, y, z and i in the following program:

```
int main()
{
int x=10, y=20, z=5, i;
i = x * (++y + z++) %3 + y/x * 2 - 5;
x += z;
y -= z;
z--;
z += 10;
z--;
Printf ("\n x = %d \n y = %d \n z = %d", x, y, z);
Printf ("\n i = %d", i);
}
```

x	y	z	i
~~10~~	~~20~~	~~5~~	1
16	24	6	
15	15	16	
		15	

Soln:

x=10, y=20, z=5, i;

1) i = x * (++y + z++) %3 + y/x * 2 - 5

The equivalent expression is,

y = y+1 = 20+1 = 21

i = x * (y+z) %3 + y/x * 2 - 5

= 10 * (21+5) %3 + 21/10 * 2 - 5

= 10 * 26 %3 + 21/10 * 2 - 5

= 260 %3 + 21/10 * 2 - 5

= 2 + 21/10 * 2 - 5

= 2 + 2 * 2 - 5

= 2 + 4 - 5

= 6 - 5

= 1

z = z+1 = 5+1 = 6

2) x += z ⇒ x = x + z = 10 + 5 = 15

3) y -= z ⇒ y = y - z = 21 - 6 = 15

4) x-- ⇒ x = x - 1 = 16 - 1 = 15

5) z += 10 ⇒ z = z + 10 ⇒ 6 + 10 = 16

6) --z ⇒ z = z - 1 ⇒ 16 - 1 = 15

Output

x = 15
y = 15
z = 15
i = 1

5) What will be the output of x, y, z and k in the following problems.

```
int main()
{
 int x=10, y=20, z, k;
 z = ++y / x--;
 y+=10;
 x+=20;
 y--;
 x++;
z*=15;
k=(x==y ? (x+y) : z);
Printf("x=%d \n y=%d \n z=%d \n k=%d", x, y, z, k);
}
```

x	y	z	k
~~10~~	~~20~~	~~2~~	60
~~9~~	~~21~~	30	
~~29~~	~~31~~		
30	30		

Soln:

1) `z = ++y / x--;`

The equivalent expression is,

$y = y+1 \Rightarrow 20+1 = 21$

$z = y/x \Rightarrow 21/10 = 2$

$x = x-1 \Rightarrow 10-1 = 9$

2) `y+=10`

$y = y+10 \Rightarrow 21+10 \Rightarrow 31$

3) `x+=20`

$x = x+20 \Rightarrow 9+20 \Rightarrow 29$

4) $y-- \Rightarrow y = y-1 = 31-1$

$= 30$

x++ ⇒ x = x+1 ⇒ 29+1 = 30

6) z*=15 ⇒ z = z*15
= 2*15
= 30

7) k = (x==y ? (x+y) : z)

Here, x==y is true so,
k = x+y = 30+30 = 60

Output
x=30
y=30
z=30
k=60

6) find the value of k:

int i=4, j=6, k=7

k += (5+i*j%7)*6/(i+j)-3

Now, Equivalent expression is,

k += (5+4*6%7)*6/(4+6)-3

or, k += (5+24%7)*6/(4+6)-3

or, k += (5+3)*6/(4+6)-3

or, k += 8*6/10-3

or, k += 48/10-3

or, k += 4-3

k += 1

k = k+1 = 7+1 = 8

7) Find the value of x, y, z, w, p and q.

```
int main()
{
int x=10, y=5, z, w=9, p, q;
x++ = 10;
y=x++;
z=--x;
w=x;
p=x+y;
q=p+w;
z=p++;
}
```

x	y	z	w	p	q
10 → 20 → 21	5 → 20	20	9	40 → 41	40

Soln: Evaluating the given expression

1) x++ = 10;

x = x+10 ⇒ 10+10 = 20;

2) y = x++;

y = x;

⇒ y = 20

x = x+1;

= 20+1

= 21

3) z = --x;

x = x - 1

= 21 - 1

= 20

z = x

∴ z = 20

4) $w = x$

$w = w/x = 9/20 = 0$

5) $p = x + y$

$= 20 + 20 = 40$

6) $q = p + w$

$= 40 + 0$

$= 40$

7) $z = p + 1$

$z = p$

or, $z = 40$

$p = p + 1 = 40 + 1$

$= 41$

$x = 20$

$y = 20$

$z = 40$

$w = 0$

$p = 41$

$q = 40$

(8) Soln:

Given,

i) $x = 3, y = 4, z = 5$

ii) $x = ++x + y$

$x = x + 1 \Rightarrow 3 + 1 = 4$

$x = x + y \Rightarrow 4 + 4 = 8$

3) $x = 6$:

$x += y$

$\Downarrow$

$x = x + y = 6 + 4$

$= 10$

4) $x = 2$;

$z = z * x + y++$;

$\Downarrow$

or, $z = z * x + y$

$z = 5 * 2 + 4$

$= 10 + 4$

$= 14$

$y = y + 1 = 4 + 1 = 5$

5) $z = 10$

$z = z * (x + y)$

$= 10 * (2 + 5)$

$= 10 * 7$

$= 70$

6) $z = 10$

$z *= x + y$

$z = z * (x + y)$

$= 10 * (2 + 5)$

$= 10 * 7 = 70$

7) $z = 10$

$z *= x + y \Rightarrow z = z * (x + y) = 10 * (2 + 5) = 10 * 7 = 70$

x	y	z
3	4	5
4		14
8		1y
6		70
10		10
2		70
		70

Output

Given $x = 3, y = 4$ and $z = 5$

$x = x + y$ assigns 8 to x

$x += y$ assigns 10 to x

$z = z * x + y$ assigns 14 to z

$z = z * (x + y)$ assigns 70 to z

$z *= x + y$ assigns 70 to z

$z = x + y$ assigns 70 to z

9) Find the output of the following program.

```
int main()
{
  int x=100;
  Printf("%d\n", 10 + x++);
  Printf("%d\n", 10 + ++x);
}
```

x
100
101
102

Soln:

Evaluating first statement,

~~x = x + 1 = 100 + 1~~
~~x = 101~~

$10 + x \Rightarrow 10 + 100 = 110$

$x = x + 1 \Rightarrow 100 + 1 = 101$

Evaluating second statement,

$x = x + 1$
$= 101 + 1$
$= 102$

$10 + x \Rightarrow 10 + 102$
$= 112$

Output

110
112

10) Evaluate:

1) $k = 3*4/5 + 5/5 + 4 - 1 + 6/8$

$= 12/5 + 5/5 + 4 - 1 + 6/8$ [∵ operation $3*4$]

$= 2 + 5/5 + 4 - 1 + 6/8$ [" $12/5$]

$= 2 + 1 + 4 - 1 + 6/8$ [" $5/5$]

$= 2 + 1 + 4 - 1 + 0$ [" $6/8$]

$= 3 + 4 - 1 + 0$ [" $2+1$]

$= 7 - 1 + 0$ [" $3+4$]

$= 6 + 0$ [" $7-1$]

$= 6$ [" $6+0$]

2) $m = a/2 + a*4/a - a + y/3$

where, $y = 1.5$, $a = 3$ and assume m to be float

Evaluating the given expression,

$m = 3/2 + 3*4/3 - 3 + 1.5/3$

$= 1 + 3*4/3 - 3 + 1.5/3$ [operation $3/2$]

$= 1 + 12/3 - 3 + 1.5/3$ [" $3*4$]

$= 1 + 4 - 3 + 1.5/3$ [" $12/3$]

$= 1 + 4 - 3 + 0.5$ [" $1.5/3$]

$= 5 - 3 + 0.5$ [" $1+4$]

$= 2 + 0.5$ [" $5-3$]

$= 2.5$ [" $2+0.5$]

(iii) $i = j/2*3 + 3/8 + j*j*j\%4$ if $j=4$

Soln:

$i = 4/2*3 + 3/8 + 4*4*4\%4$ [initialization of variable]

$= 2*3 + 3/8 + 4*4*4\%4$ [operation 4/2]

$= 6 + 3/8 + 4*4*4\%4$ [" 2*3]

$= 6 + 0 + 4*4*4\%4$ [" 3/8]

$= 6 + 0 + 16*4\%4$ [" 4*4]

$= 6 + 0 + 64\%4$ [" ~~16*4~~]

$= 6 + 0 + 0$ [" 64%4]

$= 6 + 0$ [" 6+0]

$= 6$ ~~i=~~ [" 6+0]

Note

int & int => int

int & float => float

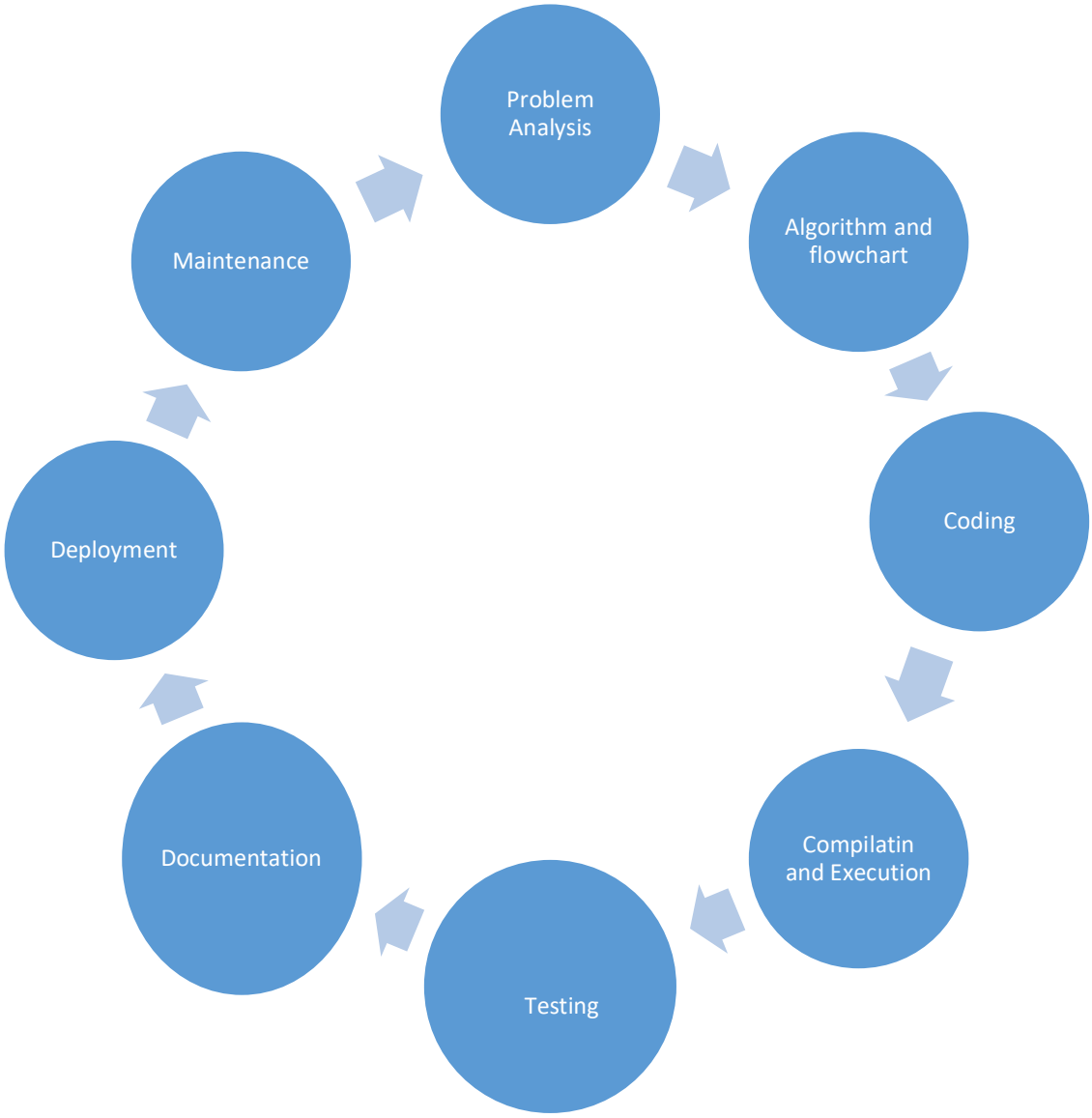
float & int => float

float & float => float

Software development Lifecycle

Software Development Life Cycle is a basically a detailed plan which describes how to create, maintain, alter and improve a specific software product (program).

SDLC consists of following activities:



1. **Problem Analysis:** It is important to give a clear and precise problem. In this step, we should specify the following things.
    - Objectives
    - Input requirements
    - Output requirements
    - Processing requirements
    - Evaluating feasibility
2. **Algorithm and flowchart:** Algorithm and flowchart helps to convert users' needs, logic or ideas into a suitable form, which helps the programmer in coding and implementation.
3. **Coding:** The coding is a process of transforming program logic or ideas into computer language format.
4. **Compilation and execution:**
   Compilation is the process of converting high level language into low level language because computer understand only low level language. Once, compilation is completed then the program is linked with another library files needed for execution.
5. **Debugging and Testing:** Debugging is the process of isolating and correcting any type of errors and testing ensures the program performs correctly the required task.
6. **Documentation:** Every step in the project is documented for future reference. Documentation of program helps to those who use, maintain and extend the program in future.
7. **Deployment:** The software is deployed after it has been approved for release.
8. **Maintenance:** Software Maintenance is the process of modifying a software for the following reason.
    - To remove the a bug from the program.
    - To improve the efficiency of code.
    - Add new function to fulfill new user requirements.