**Experiment: 4**

PART A

(PART A: TO BE REFFERED BY STUDENTS)

**Aim:** **To study loops in C++ programming.**

**Learning Outcomes: Learner would be able to**

1. Interpret the scenario to decide on repetitive blocks.
2. Explain using algorithm and flowchart conditional constructs as per scenario.

**Task 1:** For given algorithm write C++ code using for loop and while loop.

1. Start
2. Print “How many numbers”
3. Input N
4. S=0
5. C=1
6. Print “Enter the number”
7. Input A
8. S=S+A
9. C=C+1
10. If C<=N then GOTO Step 6
11. Avg=S/N
12. Print “Average is “ Avg
13. Stop

**Task 2:** Is there any difference between the following for statements? Explain

* 1. for(x=1;x<100;x++);
  2. for(x=1;x<100;++x);
  3. for(x=1;x<100;x=x+1);
  4. for(x=1;x<100;x+=1);
  5. x=1; for(;x<100;x++);

**Task 3:** State the output or error without using code blocks for below blocks of code. Give your justification.

|  |  |
| --- | --- |
| 1. #include<iostream>  void main()  {  int c;  for(c=5;c<5;c++)  cout<<“\n Hello”;  } | 2. #include<iostream>  void main()  {  int x=10;  for(;x<=50;x++);  cout<<x;  } |
| 3. #include<iostream>  void main()  {  int c=5;  while( c )  {  cout<<c;  c--;  }  } | 4. #include<iostream>  void main()  {  int c=5;  while( c )  {  cout<<c;  c=c-2;  }  } |
| 5. #include<iostream>  void main()  {  int k=1000;  while(k>0)  cout<<“I am a loop”;  } | 6. #include<iostream>  void main()  {  int c=1;  while(1)  {  cout<<c;  c++;  } |
| 7. iB = 10;  while ( iB < 50)  {  iTry = iB \* 10;  cout<<"\nNow is number: "<<iTry;  iB += 5;  } | 8. int i = 0;     for (i=0; i<20; i++)     {       switch(i)       {         case 0:           i += 5;         case 1:           i += 2;         case 5:           i += 5;         default:           i += 4;           break;       }       cout<<i;     } |

**Task 4:** Write a C++ program to print Fibonacci series up to N. (for loop)

**Task 5:** Write a C++ program and draw flowchart to calculate the sum of square of digits of a number given by the user. (while loop)

**Task 6:** Write a C++ program to print the sum of the following series up to n terms where n is given by the user.

1+x+x2/2! + x3/3! +… n terms

**Task 7:** Write a C++ program to find sum of squares of numbers entered by the user till user enters a negative number. The program will not find sum of square of a number, if it is divisible by 3. (using do-while loop)

**Theory:**

You may encounter situations, when a block of code needs to be executed several number of times. In general, statements are executed sequentially: The first statement in a function is executed first, followed by the second, and so on.

Programming languages provide various control structures that allow for more complicated execution paths.

A loop statement allows us to execute a statement or group of statements multiple times. Given below is the general form of a loop statement in most of the programming languages −



C programming language provides the following types of loops to handle looping requirements.

|  |  |
| --- | --- |
| **Sr.No.** | **Loop Type & Description** |
| 1 | [**while loop**](https://www.tutorialspoint.com/cprogramming/c_while_loop.htm)  Repeats a statement or group of statements while a given condition is true. It tests the condition before executing the loop body. |
| 2 | [**for loop**](https://www.tutorialspoint.com/cprogramming/c_for_loop.htm)  Executes a sequence of statements multiple times and abbreviates the code that manages the loop variable. |
| 3 | [**do...while loop**](https://www.tutorialspoint.com/cprogramming/c_do_while_loop.htm)  It is more like a while statement, except that it tests the condition at the end of the loop body. |
| 4 | [**nested loops**](https://www.tutorialspoint.com/cprogramming/c_nested_loops.htm)  You can use one or more loops inside any other while, for, or do..while loop. |

A **for** loop is a repetition control structure that allows you to efficiently write a loop that needs to execute a specific number of times.

## Syntax

The syntax of a **for** loop in C programming language is −

**for ( init; condition; increment ) {**

**statement(s);**

**}**



#include<iostream>

int main () {

int a;

/\* for loop execution \*/

for( a = 10; a < 20; a = a + 1 ){

cout<<a<<endl;

}

return 0;

}

When the above code is compiled and executed, it produces the following result −

value of a: 10

value of a: 11

value of a: 12

value of a: 13

value of a: 14

value of a: 15

value of a: 16

value of a: 17

value of a: 18

value of a: 19

A **while** loop in C programming repeatedly executes a target statement as long as a given condition is true.

## Syntax

The syntax of a **while** loop in C programming language is −

while(condition) {

statement(s);

}



#include<iostream>

int main () {

/\* local variable definition \*/

int a = 10;

/\* while loop execution \*/

while( a < 20 ) {

cout<<a<<endl;

a++;

}

return 0;

}

When the above code is compiled and executed, it produces the following result −

value of a: 10

value of a: 11

value of a: 12

value of a: 13

value of a: 14

value of a: 15

value of a: 16

value of a: 17

value of a: 18

value of a: 19

PART B

(PART B: TO BE COMPLETED BY STUDENTS)

Students must submit the soft copy as per following segments within two hours of the practical. The soft copy must be uploaded on the portal at the end of the practical. The filename should be **PPS\_batch\_rollno\_experimentno Example: PPS\_B2\_B001\_Exp1**

|  |  |
| --- | --- |
| **Roll No.:** | **Name:** |
| **Prog/Yr/Sem:** | **Batch:** |
| **Date of Experiment:** | **Date of Submission:** |

**Task 1:**

**Task 2:**

**Task 3:**

**Task 4:**

**Task 5:**

**Task 6:**

**Task 7:**

**Conclusion (Learning Outcomes):** Reflect on the questions answered by you jot down your learnings about the Topic: Loops.

**Home Work Questions:**

1. What will be the output of the following programs?

|  |  |
| --- | --- |
| (a) void main()  {  int i=0;  for(;i;)  cout"\nHere is some mail for you";  } |  |
| (b) void main()  {  int i=0;  do  { cout<<i--;  }while(i>0);  cout<<i;  } |  |
| (c) void main()  {  int i=1,j=1;  while(1)  {  if(i>5)  break;  else  j+=i;  cout<<j;  i+=j;  }  } |  |
| (d) void main()  {  int i=0, x=0;  for ( i=1; i<5; i++)  {  if(i%2==1)  X+=I;  else  x--;  cout<<x;  continue;  }  cout<<x;  } |  |

1. Rewrite the following code segment using for.

|  |  |
| --- | --- |
| x = 100;  while ( x < 4000)  { Count ++ ;  Add = Count \* x;  x += 150;  } |  |
| cycle\_number = 1;  while ( cycle\_number < 100)  {  cout<<"\ncurrent cycle number:"<<cycle\_number;  cycle\_number \*= 2;  } |  |