**Rajarshi janak University**

**University Campus**

Ramanand chowk , Janakpurdham

**Lab Report**

On

**Object-Oriented programming with C++**

**Submitted By :-**  **Submitted To :-**

**Name:- Khalid Hussain** **Department of Science & Technology**

**Roll No** :- 08

**Date** :- 2080/07/23 Checked By

**LIST OF EXPEREMENTS**

**Name of the experiment Page No.**

1. **To Study the Prime number program in C++ 1**
2. **To Study the Fibonacci Series in C++ 2**
3. **To Study the Palindrome program in C++ 3**
4. **To Study the Factorial Program in C++ 4**
5. **To Study the Armstrong Number in C++ 5**
6. **To Study the Sum of digits Program in C++ 6**
7. **To Study the Program to reverse number in C++ 7**
8. **To study the Program to swap two numbers without third 7 variable in C++**
9. **To study the Program of Matrix Multiplication in C++ 8-9**
10. **To Study the Program to convert Decimal to Binary. 10**
11. **To study the Program to find greatest of four numbers. 11**

**Date** :- 2080/07/23 Signature of Teacher

**Experiment No. :-1**

**Name of The Experiment:-** Prime Number Program in C++

Prime number is a number that is greater than 1 and divided by 1 or itself. In other words, prime numbers can't be divided by other numbers than itself or 1. For example 2, 3, 5, 7, 11, 13, 17, 19, 23.... are the prime numbers.

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n, i, m=0, flag=0;
6. cout << "Enter the Number to check Prime: ";
7. cin >> n;
8. m=n/2;
9. **for**(i = 2; i <= m; i++)
10. {
11. **if**(n % i == 0)
12. {
13. cout<<"Number is not Prime."<<endl;
14. flag=1;
15. **break**;
16. }
17. }
18. **if** (flag==0)
19. cout << "Number is Prime."<<endl;
20. **return** 0;
21. }

**Output:-**

Enter the Number to check Prime: 17

Number is Prime.

Enter the Number to check Prime: 57

Number is not Prime.

1

**Experiment No. :-2**

**Name of The Experiment:-** Fibonacci Series in C++

In case of fibonacci series, next number is the sum of previous two numbers for example 0, 1, 1, 2, 3, 5, 8, 13, 21 etc. The first two numbers of fibonacci series are 0 and 1.

Fibonaccci Series in C++ without Recursion

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main() {
4. **int** n1=0,n2=1,n3,i,number;
5. cout<<"Enter the number of elements: ";
6. cin>>number;
7. cout<<n1<<" "<<n2<<" "; //printing 0 and 1
8. **for**(i=2;i<number;++i){ //loop starts from 2 because 0 and 1 are already printed
9. n3=n1+n2;
10. cout<<n3<<" ";
11. n1=n2;
12. n2=n3;}
13. **return** 0;  }

**Output:-**C++ vs Java

Enter the number of elements: 10

0 1 1 2 3 5 8 13 21 34

**Fibonnaci series using recursion in C++**

1. #include<iostream>
2. **using** **namespace** std;
3. **void** printFibonacci(**int** n){
4. **static** **int** n1=0, n2=1, n3;
5. **if**(n>0){
6. n3 = n1 + n2;
7. n1 = n2;
8. n2 = n3;
9. cout<<n3<<" ";
10. printFibonacci(n-1); } }
11. **int** main(){
12. **int** n;
13. cout<<"Enter the number of elements: ";
14. cin>>n;
15. cout<<"Fibonacci Series: ";
16. cout<<"0 "<<"1 ";
17. printFibonacci(n-2);  //n-2 because 2 numbers are already printed
18. **return** 0;  }

**Output:-**

Enter the number of elements: 15

Fibonacci Series: 0 1 1 2 3 5 8 13 21 34 55 89 144 233 377

**2**

**Experiment No. :-3**

**Name of The Experiment:-** Palindrome program in C++

A **palindrome number** is a number that is same after reverse. For example 121, 34543, 343, 131, 48984 are the palindrome numbers.

**Palindrome number algorithm**

* Get the number from user
* Hold the number in temporary variable
* Reverse the number
* Compare the temporary number with reversed number
* If both numbers are same, print palindrome number
* Else print not palindrome number

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n,r,sum=0,temp;
6. cout<<"Enter the Number=";
7. cin>>n;
8. temp=n;
9. **while**(n>0)
10. {
11. r=n%10;
12. sum=(sum\*10)+r;
13. n=n/10;
14. }
15. **if**(temp==sum)
16. cout<<"Number is Palindrome.";
17. **else**
18. cout<<"Number is not Palindrome.";
19. **return** 0;
20. }

**Output:**

Enter the Number=121

Number is Palindrome.

Enter the number=113

Number is not Palindrome.

3

**Experiment No. :-4**

**Name of The Experiment:-** Factorial program in C++

Factorial of n is the product of all positive descending integers. Factorial of n is denoted by n!. For example:

1. 4! = 4\*3\*2\*1 = 24
2. 6! = 6\*5\*4\*3\*2\*1 = 720

Factorial Program using Recursion

**Syntax:-**

1. #include<iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** factorial(**int**);
6. **int** fact,value;
7. cout<<"Enter any number: ";
8. cin>>value;
9. fact=factorial(value);
10. cout<<"Factorial of a number is: "<<fact<<endl;
11. **return** 0;
12. }
13. **int** factorial(**int** n)
14. {
15. **if**(n<0)
16. **return**(-1); /\*Wrong value\*/
17. **if**(n==0)
18. **return**(1);  /\*Terminating condition\*/
19. **else**
20. {
21. **return**(n\*factorial(n-1));
22. }
23. }

**Output:-**

**E**nter any number: 6

Factorial of a number is: 720

4

**Experiment No. :-5**

**Name of The Experiment:-**Armstrong Number in C++

**Armstrong number** is a number that is equal to the sum of cubes of its digits. For example 0, 1, 153, 370, 371 and 407 are the Armstrong numbers. Let's try to understand why **371** is an Armstrong number.

1. 371 = (3\*3\*3)+(7\*7\*7)+(1\*1\*1)
2. So:
3. 27+343+1=371

**Syntax:-**Features of Java - J

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n,r,sum=0,temp;
6. cout<<"Enter the Number=  ";
7. cin>>n;
8. temp=n;
9. **while**(n>0)
10. {
11. r=n%10;
12. sum=sum+(r\*r\*r);
13. n=n/10;
14. }
15. **if**(temp==sum)
16. cout<<"Armstrong Number."<<endl;
17. **else**
18. cout<<"Not Armstrong Number."<<endl;
19. **return** 0;
20. }

**Output:-**

Enter the Number= 371

Armstrong Number.

Enter the Number= 342

Not Armstrong Number.

5

**Experiment No. :-6**

**Name of The Experiment:-** Sum of digits program in C++

Sum of digits algorithm

* **Step 1:** Get number by user
* **Step 2:** Get the modulus/remainder of the number
* **Step 3:** sum the remainder of the number
* **Step 4:** Divide the number by 10
* **Step 5:** Repeat the step 2 while number is greater than 0.

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** n,sum=0,m;
6. cout<<"Enter a number: ";
7. cin>>n;
8. **while**(n>0)
9. {
10. m=n%10;
11. sum=sum+m;
12. n=n/10;
13. }
14. cout<<"Sum is= "<<sum<<endl;
15. **return** 0;
16. }

**Output:-**

OOPs Concepts in Java

Enter a number: 23

Sum is= 5

Enter a number: 624

Sum is= 12

6

**Experiment No. :-7**

**Name of The Experiment:-** C++ Program to reverse number

We can reverse a number in C++ using loop and arithmetic operators. In this program, we are getting number as input from the user and reversing that number.

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main() {
4. **int** n, reverse=0, rem;
5. cout<<"Enter a number: ";
6. cin>>n;
7. **while**(n!=0){
8. rem=n%10;
9. reverse=reverse\*10+rem;
10. n/=10; }
11. cout<<"Reversed Number: "<<reverse<<endl;
12. **return** 0;
13. }

**Output:-**

Enter a number: 234

Reversed Number: 432

**Experiment No. :-8**

**Name of The Experiment:-** Program to swap two numbers without third variable

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main() {
4. **int** a=5, b=10;
5. cout<<"Before swap a= "<<a<<" b= "<<b<<endl;
6. a=a\*b; //a=50 (5\*10)
7. b=a/b; //b=5 (50/10)
8. a=a/b; //a=10 (50/5)
9. cout<<"After swap a= "<<a<<" b= "<<b<<endl;
10. **return** 0; }

**Output:-**

Before swap a= 5 b= 10

After swap a= 10 b= 5

7

**Experiment No. :-9**

**Name of The Experiment:-** Matrix multiplication in C++

We can add, subtract, multiply and divide 2 matrices. To do so, we are taking input from the user for row number, column number, first matrix elements and second matrix elements. Then we are performing multiplication on the matrices entered by the user.

Let's try to understand the matrix multiplication of **3\*3 and 3\*3** matrices by the figure.



**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** a[10][10],b[10][10],mul[10][10],r,c,i,j,k;
6. cout<<"enter the number of row=";
7. cin>>r;
8. cout<<"enter the number of column=";
9. cin>>c;
10. cout<<"enter the first matrix element=\n";
11. **for**(i=0;i<r;i++)
12. {
13. **for**(j=0;j<c;j++)
14. {
15. cin>>a[i][j];
16. }
17. }
18. cout<<"enter the second matrix element=\n";
19. **for**(i=0;i<r;i++)
20. {
21. **for**(j=0;j<c;j++)
22. {
23. cin>>b[i][j];     8
24. }
25. }
26. cout<<"multiply of the matrix=\n";
27. **for**(i=0;i<r;i++)
28. {
29. **for**(j=0;j<c;j++)
30. {
31. mul[i][j]=0;
32. **for**(k=0;k<c;k++)
33. {
34. mul[i][j]+=a[i][k]\*b[k][j];
35. }
36. }
37. }
38. //for printing result
39. **for**(i=0;i<r;i++)
40. {
41. **for**(j=0;j<c;j++)
42. {
43. cout<<mul[i][j]<<" ";
44. }
45. cout<<"\n";
46. }
47. **return** 0;
48. }

**Output:-**

enter the number of row=3

enter the number of column=3

enter the first matrix element=

1 2 3

1 2 3

1 2 3

enter the second matrix element=

1 1 1

2 1 2

3 2 1

multiply of the matrix=

14 9 8

14 9 8

14 9 8

9

**Experiment No. :-10**

**Name of The Experiment:-** C++ Program to convert Decimal to Binary

Decimal to Binary Conversion Algorithm

**Step 1:** Divide the number by 2 through % (modulus operator) and store the remainder in array

**Step 2:** Divide the number by 2 through / (division operator)

**Step 3:** Repeat the step 2 until the number is greater than zero

**Syntax:-**

1. #include <iostream>
2. **using** **namespace** std;
3. **int** main()
4. {
5. **int** a[10], n, i;
6. cout<<"Enter the number to convert: ";
7. cin>>n;
8. **for**(i=0; n>0; i++)
9. {
10. a[i]=n%2;
11. n= n/2;
12. }
13. cout<<"Binary of the given number= ";
14. **for**(i=i-1 ;i>=0 ;i--)
15. {
16. cout<<a[i];
17. }
18. }

**Output:-**

Enter the number to convert: 9

Binary of the given number= 1001

10

**Experiment No. :-11**

**Name of The Experiment:-**  Program to find greatest of four numbers

In this tutorial, we will write a C++ program to find the greatest of four numbers.

**Syntax:-**

1. #include <bits/stdc++.h>
2. **using** **namespace** std;
3. **void** find\_greatest(**int** a, **int** b, **int** c, **int** d)  {
4. **if** (a > b) {
5. **if** (a > c) {
6. **if** (a > d) {
7. cout << "a is greatest";  }
8. **else** {
9. cout << "d is greatest";  }}}
10. **else** **if** (b > c) {
11. **if** (b > d) {
12. cout << "b is greatest";  }
13. **else** {
14. cout << "d is greatest";  }}
15. **else** **if** (c > d) {
16. cout << "c is greatest";  }
17. **else** {
18. cout << "d is greatest";  }}
19. **int** main()  {
20. **int** a = 10, b = 50, c = 20, d = 25;
21. cout << "a=" << a << " b=" << b << " c=" << c << " d=" << d;
22. cout << "\n";
23. find\_greatest(a, b, c, d);
24. a = 35, b = 50, c = 99, d = 2;
25. cout << "\n";
26. cout << "a=" << a << " b=" << b << " c=" << c << " d=" << d;
27. cout << "\n";
28. find\_greatest(a, b, c, d);
29. **return** 0; }

**Output:-**

a=10 b=50 c=20 d=25

b is greatest

a=35 b=50 c=99 d=2

c is greatest

11