

OOP: Assignment No. 1

Name: Naman Bharsakale

PRN No: 255200003

Branch: Computer Science and Engineering (AIDED)

Batch: S8

1) Check Number is Even or Odd

Input:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int num;
```

```
    cout << "Enter a number: ";
```

```
    cin >> num;
```

```
    if (num % 2 == 0)
```

```
        cout << num << " is Even";
```

```
    else
```

```
        cout << num << " is Odd";
```

```
    return 0;
```

```
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ evenodd.cpp -o evenodd.exe
```

```
PS D:\DS\CPP_Assignment> ./evenodd.exe
```

```
Enter a number: 4
```

```
4 is Even
```

```
PS D:\DS\CPP_Assignment> ./evenodd.exe
```

```
Enter a number: 5
```

```
5 is Odd
```

2) Find Largest Number Among 3 Numbers.

Input:


```
#include <iostream>
using namespace std;

int main() {
    int a, b, c;
    cout << "Enter three numbers: ";
    cin >> a >> b >> c;

    if (a >= b && a >= c)
        cout << "Largest: " << a;
    else if (b >= a && b >= c)
        cout << "Largest: " << b;
    else
        cout << "Largest: " << c;

    return 0;
}
```

Output:

A terminal window with a dark background. The prompt is 'PS D:\DS\CPP_Assignment>'. The command './largest.exe' is entered and executed. The output shows 'Enter three numbers: 12 233 22' followed by 'Largest: 233' on the next line.

```
PS D:\DS\CPP_Assignment> ./largest.exe
Enter three numbers: 12 233 22
Largest: 233
```

3) Check Whether a Character is a Vowel or Consonant

Input:

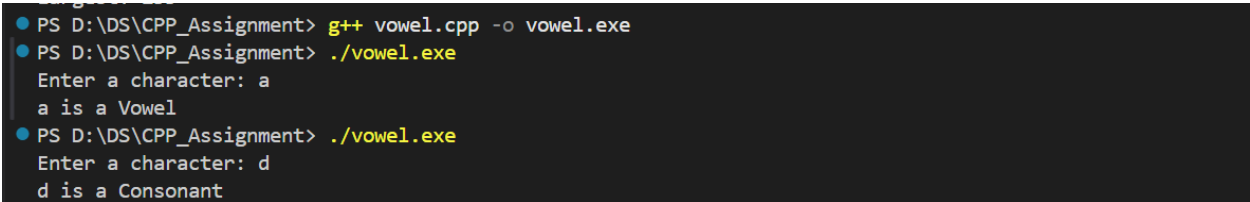
```
#include <iostream>
using namespace std;

int main() {
    char ch;
    cout << "Enter a character: ";
    cin >> ch;

    if (ch=='a' || ch=='e' || ch=='i' || ch=='o' || ch=='u' ||
        ch=='A' || ch=='E' || ch=='I' || ch=='O' || ch=='U')
        cout << ch << " is a Vowel";
    else
        cout << ch << " is a Consonant";

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ vowel.cpp -o vowel.exe
PS D:\DS\CPP_Assignment> ./vowel.exe
Enter a character: a
a is a Vowel
PS D:\DS\CPP_Assignment> ./vowel.exe
Enter a character: d
d is a Consonant
```

4)Check if a Given Year is a Leap Year

Input:

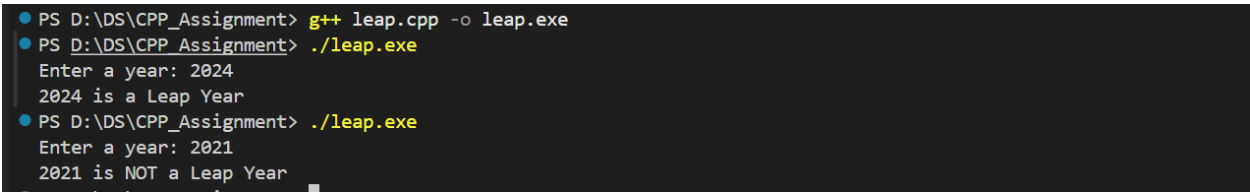
```
#include <iostream>
using namespace std;

int main() {
    int year;
    cout << "Enter a year: ";
    cin >> year;

    if ((year % 4 == 0 && year % 100 != 0) || (year % 400 == 0))
        cout << year << " is a Leap Year";
    else
        cout << year << " is NOT a Leap Year";

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ leap.cpp -o leap.exe
PS D:\DS\CPP_Assignment> ./leap.exe
Enter a year: 2024
2024 is a Leap Year
PS D:\DS\CPP_Assignment> ./leap.exe
Enter a year: 2021
2021 is NOT a Leap Year
```

5)Print Multiplication Table of a Number

Input:

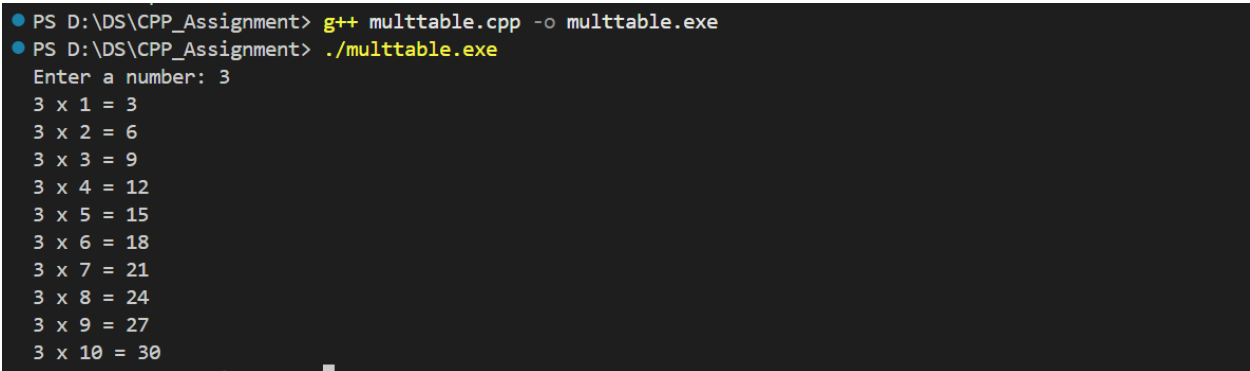
```
#include <iostream>
using namespace std;
```

```
int main() {
    int num;
    cout << "Enter a number: ";
    cin >> num;

    for (int i = 1; i <= 10; i++) {
        cout << num << " x " << i << " = " << num * i << endl;
    }

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ multtable.cpp -o multtable.exe
PS D:\DS\CPP_Assignment> ./multtable.exe
Enter a number: 3
3 x 1 = 3
3 x 2 = 6
3 x 3 = 9
3 x 4 = 12
3 x 5 = 15
3 x 6 = 18
3 x 7 = 21
3 x 8 = 24
3 x 9 = 27
3 x 10 = 30
```

6)Calculate Sum of First n Natural Numbers

Input:

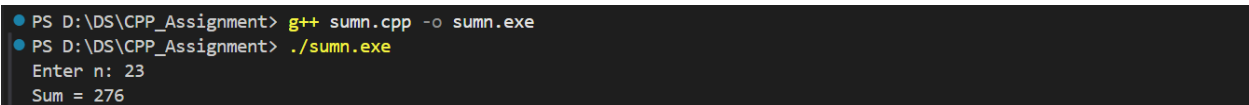
```
#include <iostream>
using namespace std;
```

```
int main() {
    int n, sum = 0;
    cout << "Enter n: ";
    cin >> n;

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

    cout << "Sum = " << sum;
    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ sumn.cpp -o sumn.exe
PS D:\DS\CPP_Assignment> ./sumn.exe
Enter n: 23
Sum = 276
```

7)Find Factorial of a Number

Input:

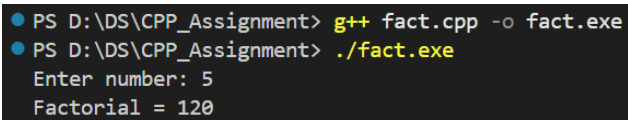
```
#include <iostream>
using namespace std;

int main() {
    int n, fact = 1;
    cout << "Enter number: ";
    cin >> n;

    for (int i = 1; i <= n; i++) {
        fact *= i;
    }

    cout << "Factorial = " << fact;
    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ fact.cpp -o fact.exe
PS D:\DS\CPP_Assignment> ./fact.exe
Enter number: 5
Factorial = 120
```

8) Reverse a Number.

Input:

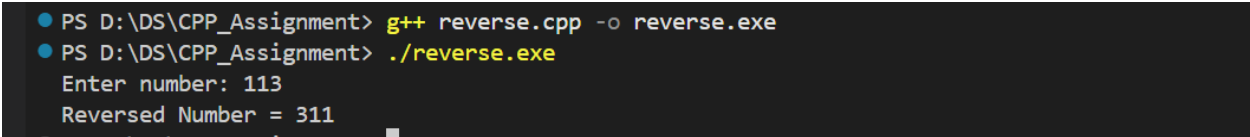
```
#include <iostream>
using namespace std;

int main() {
    int n, rev = 0;
    cout << "Enter number: ";
    cin >> n;

    while (n > 0) {
        rev = rev * 10 + (n % 10);
        n /= 10;
    }

    cout << "Reversed Number = " << rev;
    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ reverse.cpp -o reverse.exe
PS D:\DS\CPP_Assignment> ./reverse.exe
Enter number: 113
Reversed Number = 311
```


9)Find GCD

Input:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int a, b;
```

```
    cout << "Enter two numbers: ";
```

```
    cin >> a >> b;
```

```
    while (b != 0) {
```

```
        int temp = b;
```

```
        b = a % b;
```

```
        a = temp;
```

```
    }
```

```
    cout << "GCD = " << a;
```

```
    return 0;
```

```
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ gcd.cpp -o gcd.exe
```

```
PS D:\DS\CPP_Assignment> ./gcd.exe
```

```
Enter two numbers: 30 15
```

```
GCD = 15
```

10)Find LCM

Input:

```
#include <iostream>
using namespace std;

int main() {
    int a, b, lcm, gcd, x, y;
    cout << "Enter two numbers: ";
    cin >> a >> b;

    x = a; y = b;

    while (b != 0) {
        int temp = b;
        b = a % b;
        a = temp;
    }
    gcd = a;
    lcm = (x * y) / gcd;

    cout << "LCM = " << lcm;
    return 0;
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ lcm.cpp -o lcm.exe
PS D:\DS\CPP_Assignment> ./lcm.exe
Enter two numbers: 12 18
LCM = 36
```

11)Check Whether a Number is a Palindrome or Not

Input:

```
#include <iostream>
using namespace std;

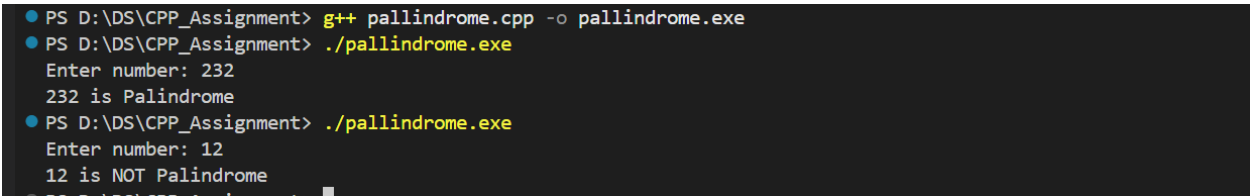
int main() {
    int n, rev = 0, temp;
    cout << "Enter number: ";
    cin >> n;
    temp = n;

    while (temp > 0) {
        rev = rev * 10 + (temp % 10);
        temp /= 10;
    }

    if (rev == n)
        cout << n << " is Palindrome";
    else
        cout << n << " is NOT Palindrome";

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ pallindrome.cpp -o pallindrome.exe
PS D:\DS\CPP_Assignment> ./pallindrome.exe
Enter number: 232
232 is Palindrome
PS D:\DS\CPP_Assignment> ./pallindrome.exe
Enter number: 12
12 is NOT Palindrome
```

12)Check Whether a Number is Prime or Not

Input:

```
#include <iostream>
using namespace std;

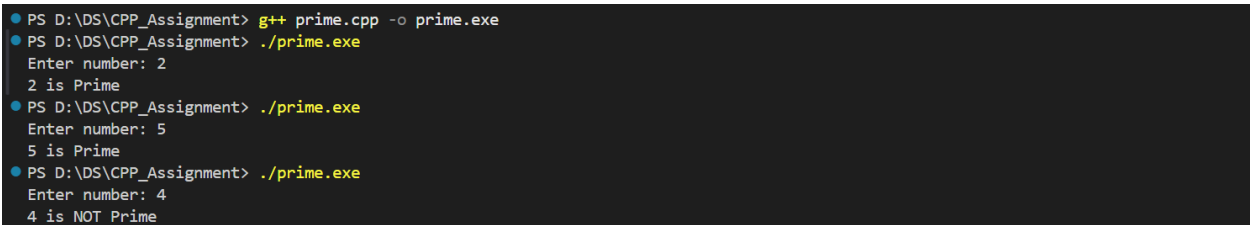
int main() {
    int n, flag = 1;
    cout << "Enter number: ";
    cin >> n;

    if (n <= 1) flag = 0;
    for (int i = 2; i <= n / 2; i++) {
        if (n % i == 0) {
            flag = 0;
            break;
        }
    }

    if (flag) cout << n << " is Prime";
    else cout << n << " is NOT Prime";

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ prime.cpp -o prime.exe
PS D:\DS\CPP_Assignment> ./prime.exe
Enter number: 2
2 is Prime
PS D:\DS\CPP_Assignment> ./prime.exe
Enter number: 5
5 is Prime
PS D:\DS\CPP_Assignment> ./prime.exe
Enter number: 4
4 is NOT Prime
```

13)Display Prime Numbers Between Two Intervals

Input:

```
#include <iostream>
using namespace std;

int main() {
    int low, high;
    cout << "Enter range: ";
    cin >> low >> high;

    for (int n = low; n <= high; n++) {
        int flag = 1;
        if (n <= 1) flag = 0;
        for (int i = 2; i <= n / 2; i++) {
            if (n % i == 0) {
                flag = 0;
                break;
            }
        }
        if (flag) cout << n << " ";
    }

    return 0;
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ primerange.cpp -o primerange.exe
PS D:\DS\CPP_Assignment> ./primerange.exe
Enter range: 20 50
23 29 31 37 41 43 47
```

14)Check Neon Numbers in a Given Range

Input:

```
#include <iostream>
using namespace std;

int main() {
    int start, end;
    cout << "Enter range: ";
    cin >> start >> end;

    for (int n = start; n <= end; n++) {
        int sq = n * n, sum = 0, temp = sq;
        while (temp > 0) {
            sum += temp % 10;
            temp /= 10;
        }
        if (sum == n)
            cout << n << " ";
    }

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ neon.cpp -o neon.exe
PS D:\DS\CPP_Assignment> ./neon.exe
Enter range: 1 10
1 9
```

15)Check Armstrong Number

Input:

```
#include <iostream>
using namespace std;

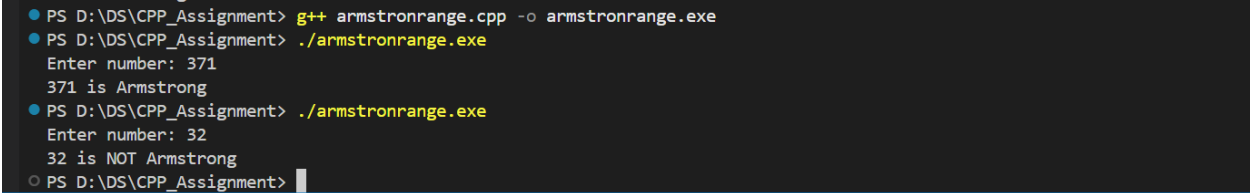
int main() {
    int n, sum = 0, temp, digits = 0;
    cout << "Enter number: ";
    cin >> n;

    temp = n;
    while (temp > 0) {
        int d = temp % 10;
        sum += d*d*d;
        temp /= 10;
    }

    if (sum == n) cout << n << " is Armstrong";
    else cout << n << " is NOT Armstrong";

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ armstronrange.cpp -o armstronrange.exe
PS D:\DS\CPP_Assignment> ./armstronrange.exe
Enter number: 371
371 is Armstrong
PS D:\DS\CPP_Assignment> ./armstronrange.exe
Enter number: 32
32 is NOT Armstrong
PS D:\DS\CPP_Assignment> |
```

16) Display Armstrong Numbers Between 1 to 1000

Input:

```
#include <iostream>
using namespace std;

int main() {
    for (int n = 1; n <= 1000; n++) {
        int sum = 0, temp = n, digits = 0;
        temp = n;
        while (temp > 0) {
            int d = temp % 10;
            sum += d*d*d;
            temp /= 10;
        }
        if (sum == n) cout << n << " ";
    }

    return 0;
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ armrange.cpp -o armrange.exe
PS D:\DS\CPP_Assignment> ./armrange.exe
1 2 3 4 5 6 7 8 9 370 371 407
```


17)Find nth Fibonacci Number

Input:

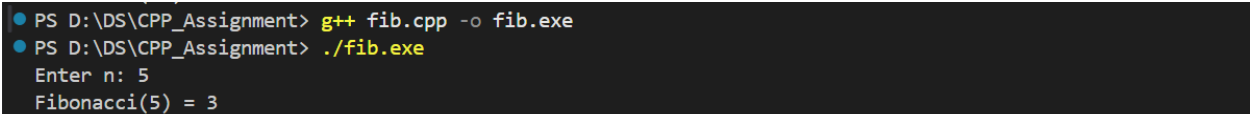
```
#include <iostream>
using namespace std;

int main() {
    int n;
    cout << "Enter n: ";
    cin >> n;

    int a = 0, b = 1, fib;
    if (n == 1) fib = 0;
    else if (n == 2) fib = 1;
    else {
        for (int i = 3; i <= n; i++) {
            fib = a + b;
            a = b;
            b = fib;
        }
    }

    cout << "Fibonacci(" << n << ") = " << fib;
    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ fib.cpp -o fib.exe
PS D:\DS\CPP_Assignment> ./fib.exe
Enter n: 5
Fibonacci(5) = 3
```

18)Sum of Fibonacci Numbers at Even Indexes up to N Terms

Input:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int n;
```

```
    cout << "Enter n: ";
```

```
    cin >> n;
```

```
    int a = 0, b = 1, sum = 0;
```

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

```
        if (i % 2 == 0) sum += b;
```

```
        int temp = b;
```

```
        b = a + b;
```

```
        a = temp;
```

```
    }
```

```
    cout << "Sum = " << sum;
```

```
    return 0;
```

```
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ fibsum.cpp -o fibsum.exe
PS D:\DS\CPP_Assignment> ./fibsum.exe
Enter n: 10
Sum = 88
```

19) Calculate the Power of a Number

Input:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int base, exp, result = 1;
```

```
    cout << "Enter base and exponent: ";
```

```
    cin >> base >> exp;
```

```
    for (int i = 1; i <= exp; i++) {
```

```
        result *= base;
```

```
    }
```

```
    cout << "Result = " << result;
```

```
    return 0;
```

```
}
```

Output:

```
Sam > cd
PS D:\DS\CPP_Assignment> g++ power.cpp -o power.exe
PS D:\DS\CPP_Assignment> ./power.exe
Enter base and exponent: 2 3
Result = 8
```

20)Display Factors of a Natural Number

Input:

```
#include <iostream>
using namespace std;
```

```
int main() {
    int n;
    cout << "Enter number: ";
    cin >> n;

    cout << "Factors: ";
    for (int i = 1; i <= n; i++) {
        if (n % i == 0)
            cout << i << " ";
    }

    return 0;
}
```

Output:

```
PS D:\DS\CPP_Assignment> g++ factors.cpp -o factors.exe
PS D:\DS\CPP_Assignment> ./factors.exe
Enter number: 45
Factors: 1 3 5 9 15 45
```

21) Make a Simple Calculator

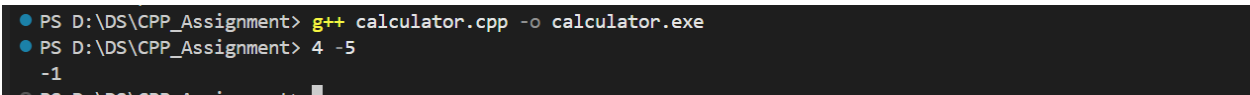
```
#include <iostream>
using namespace std;

int main() {
    char op;
    double a, b;
    cout << "Enter expression (a + b): ";
    cin >> a >> op >> b;

    switch (op) {
        case '+': cout << "Result = " << a + b; break;
        case '-': cout << "Result = " << a - b; break;
        case '*': cout << "Result = " << a * b; break;
        case '/':
            if (b != 0) cout << "Result = " << a / b;
            else cout << "Division by zero not allowed";
            break;
        default: cout << "Invalid operator";
    }

    return 0;
}
```

Output:



```
PS D:\DS\CPP_Assignment> g++ calculator.cpp -o calculator.exe
PS D:\DS\CPP_Assignment> 4 -5
-1
```