Submitted by: Sajjal Fayyaz

Submitted to: Lecturer Salman Irfan

Task 1:

Code:

reversnumber.cpp > ...

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

int main() {
    // Step 1: Ask user for an integer input
    int number;
    cout << "Please enter an integer: ";
    cin >> number;

    // Step 2: Initialize variables for reversed number and sum of digits
    int reversedNumber = 0;
    int digitSum = 0;

    // Step 3: Extract digits, reverse the number and calculate the sum of digits
    int tempNumber = number;  // Store the original number for future reference
    while (tempNumber != 0) {
        int digit = tempNumber % 10;  // Get the last digit
        reversedNumber = reversedNumber * 10 + digit;  // Build the reversed number
        digitSum += digit;  // Add the digit to the sum
        tempNumber /= 10;  // Remove the last digit from tempNumber
    }

    // Step 4: Output the reversed number and the sum of its digits
    cout << "Reversed number: " << reversedNumber << endl;
    cout << "Sum of digits: " << digitSum << endl;

    return 0;
```

Figure 1: reverse number and sum of digits

Compiler:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                    powershell

PS C:\Users\LENOVO\OneDrive\Desktop\cpp> g++ .\reversnumber.cpp -o .\reversnumber.exe
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> .\reversnumber.exe
Please enter an integer: 965
Reversed number: 569
Sum of digits: 20
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 2: compile a code

Lab task 2:

Code:

```
task2.cpp > ...
3
4   int main() {
5       // Declare a variable to hold the user input
6       int n;
7
8       // Ask the user to input a number up to which we will check even and odd numbers
9       cout << "Please enter a number: ";
10      cin >> n;
11
12      // Loop through numbers from 1 to n
13      for (int i = 1; i <= n; ++i) {
14          // Determine if the (const char [20])" is an even number."
15          if (i % 2 == 0) {
16              cout << i << " is an even number." << endl;  // If i is divisible by 2, it is even
17          } else {
18              cout << i << " is an odd number." << endl;  // Otherwise, it is odd
19          }
20      }
21
22      return 0;
23  }
24
```

Figure 3: code of even and old check

Compiler:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                    powershell

PS C:\Users\LENOVO\OneDrive\Desktop\cpp> g++ .\task2.cpp -o .\task2.exe
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> .\task2.exe
Please enter a number: 4
1 is an odd number.
2 is an even number.
3 is an odd number.
4 is an even number.
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 4: compile a code

Lab task 3:

Code:

task3.cpp > ...

```cpp
#include <iostream>
#include <cstdlib>  // For rand() and srand()
#include <ctime>    // For time()
using namespace std;

int main() {
    srand(time(0)); // Seed random number generator with current time

    // Generate a random number between 1 and 100
    int secretNumber = rand() % 100 + 1;
    int userGuess;
    int attemptsCounter = 0; // Counter to track number of attempts

    cout << "Welcome to the Number Guessing Game!\n";
    cout << "I have selected a number between 1 and 100.\n";
    cout << "Try to guess the number!\n";

    // Start guessing loop
    while (true) {
        cout << "Enter your guess: ";
        cin >> userGuess;  // Take user's guess as input
        attemptsCounter++;  // Increment attempts counter

        // Check if the guess is correct, too high, or too low
        if (userGuess < secretNumber) {
            cout << "Your guess is too low! Try again.\n";
        } else if (userGuess > secretNumber) {
            cout << "Your guess is too high! Try again.\n";
        } else {
            // User guessed correctly
            cout << "Congratulations! You've guessed the number in "
                 << attemptsCounter << " attempts.\n";
            break; // Exit the loop when the guess is correct
        }
    }

    return 0;
}
```

Figure 5: code of guessing game

Compiler:

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell

```
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> g++ .\task3.cpp -o .\task3.exe
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> .\task3.exe
Welcome to the Number Guessing Game!
I have selected a number between 1 and 100.
Try to guess the number!
Enter your guess: 50
Your guess is too low! Try again.
Enter your guess: 80
Your guess is too low! Try again.
Enter your guess: 90
Congratulations! You've guessed the number in 3 attempts.
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 6: compile a code

Lab task 4:

Code:

task4.cpp > ...

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

int main() {
    int number, limit;

    // Prompt the user to input a base number for the multiplication table
    cout << "Enter a number for the multiplication table: ";
    cin >> number;

    // Ensure the number is positive (greater than 0)
    if (number > 0) {
        // Ask for the limit (where the multiplication should stop)
        cout << "Enter the limit (up to where the table should be displayed): ";
        cin >> limit;

        // Display the multiplication table
        cout << "Multiplication table for " << number << " up to " << limit << ":\n";
        for (int i = 1; i <= limit; ++i) {
            // Output each line of the multiplication table
            cout << number << " x " << i << " = " << number * i << endl;
        }
    } else {
        // Handle invalid input where the number is not positive
        cout << "Error: Please enter a number greater than 0.\n";
    }

    return 0;
}
```

Figure 7: code of multiplication

Compiler:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                powershell

Enter the limit (up to where the table should be displayed): 10
Multiplication table for 6 up to 10:
6 x 1 = 6
6 x 2 = 12
6 x 3 = 18
6 x 4 = 24
6 x 5 = 30
6 x 6 = 36
6 x 7 = 42
6 x 8 = 48
6 x 9 = 54
6 x 10 = 60
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 8: compile a code

Lab task5:

Code:

task5.cpp > ...

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

int main() {
    int number;

    // Ask the user to input a number
    cout << "Please enter a number: ";
    cin >> number;

    int digitSum = 0;
    int workingNumber = number;  // Store the number for digit extraction

    // Extract digits and calculate the sum of the digits
    while (workingNumber > 0) {
        digitSum += workingNumber % 10;  // Add the last digit to the sum
        workingNumber /= 10;  // Remove the last digit
    }

    // Output the sum of the digits
    cout << "The sum of the digits is: " << digitSum << endl;

    // Check if the sum of the digits is divisible by 3
    if (digitSum % 3 == 0) {
        cout << "The sum is divisible by 3: Yes" << endl;
    } else {
        cout << "The sum is divisible by 3: No" << endl;
    }

    return 0;
}
```

Figure 9: code of sum and divisible of 3

Compiler:

Figure 10: compile a code

Lab task6:

Code:

Figure 11: code of number comparison

Compiler:

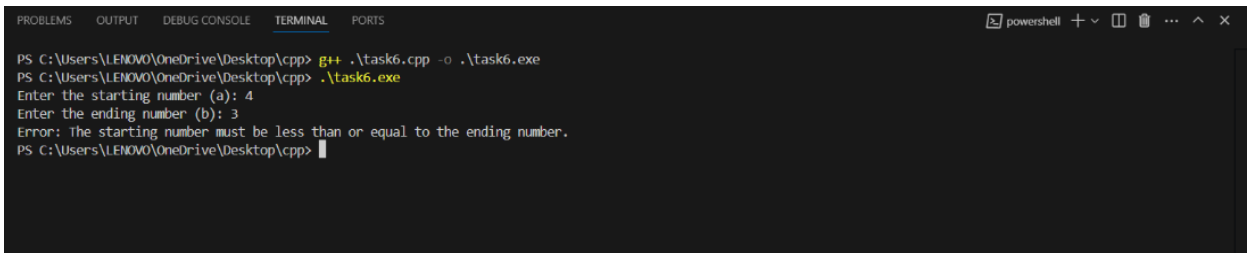

Figure 12: compile a code

Lab task 7:

Code:

reversnumber.cpp > ...

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

int main() {
    // Step 1: Ask user for an integer input
    int number;
    cout << "Please enter an integer: ";
    cin >> number;

    // Step 2: Initialize variables for reversed number and sum of digits
    int reversedNumber = 0;
    int digitSum = 0;

    // Step 3: Extract digits, reverse the number and calculate the sum of digits
    int tempNumber = number;  // Store the original number for future reference
    while (tempNumber != 0) {
        int digit = tempNumber % 10;  // Get the last digit
        reversedNumber = reversedNumber * 10 + digit;  // Build the reversed number
        digitSum += digit;  // Add the digit to the sum
        tempNumber /= 10;  // Remove the last digit from tempNumber
    }

    // Step 4: Output the reversed number and the sum of its digits
    cout << "Reversed number: " << reversedNumber << endl;
    cout << "Sum of digits: " << digitSum << endl;

    return 0;
}
```

Figure 13: code of count divisible number

Compiler:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                  powershell

PS C:\Users\LENOVO\OneDrive\Desktop\cpp> g++ .\task7.cpp -o .\task7.exe
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> .\task7.exe
Enter a positive number (n): 4
Numbers divisible by 3 between 1 and 4 are:
3
Total count of numbers divisible by 3: 1
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 14: compile a code

Lab task 8:

Code:

task8.cpp > ...

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

int main() {
    int largest;  // Variable to store the largest number
    int currentNumber;  // Variable to store each number input by the user

    // Prompt the user to enter 5 numbers
    cout << "Please enter 5 numbers to find the largest:" << endl;

    // Accept the first number to initialize 'largest' as a baseline
    cout << "Enter number 1: ";
    cin >> largest;

    // Loop to accept 4 more numbers
    for (int i = 2; i <= 5; ++i) {
        cout << "Enter number " << i << ": ";
        cin >> currentNumber;

        // Compare current number with 'largest' and update if needed
        if (currentNumber > largest) {
            largest = currentNumber;  // Update 'largest' if currentNumber is greater
        }
    }

    // Output the largest number found
    cout << "The largest number is: " << largest << "." << endl;

    return 0;
}

```

Figure 15: code of find the largest number

Compiler:

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS                                  powershell

PS C:\Users\LENOVO\OneDrive\Desktop\cpp> g++ .\task8.cpp -o .\task8.exe
PS C:\Users\LENOVO\OneDrive\Desktop\cpp> .\task8.exe
Please enter 5 numbers to find the largest:
Enter number 1: 8
Enter number 2: 9
Enter number 3: 2
Enter number 4: 5
Enter number 5: 11
The largest number is: 11.
PS C:\Users\LENOVO\OneDrive\Desktop\cpp>
```

Figure 16: compile a code

Lab task 9:

Code:

task9.cpp > ...

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

int main() {
    int n;
    int sum = 0;

    // Prompt the user to input a number N
    cout << "Please enter a positive number (N): ";
    cin >> n;

    // Check if the input is positive
    if (n <= 0) {
        cout << "Please enter a number greater than 0." << endl;
        return 1;  // Exit the program if the input is invalid
    }

    // Loop through all numbers from 1 to N
    for (int i = 1; i <= n; ++i) {
        // Check if the current number is odd
        if (i % 2 != 0) {
            sum += i;  // Add the odd number to the sum
        }
    }

    // Display the result: sum of odd numbers
    cout << "The sum of odd numbers between 1 and " << n << " is: " << sum << endl;

    return 0;
}

```

Figure 17: code of sum of number

Compiler:

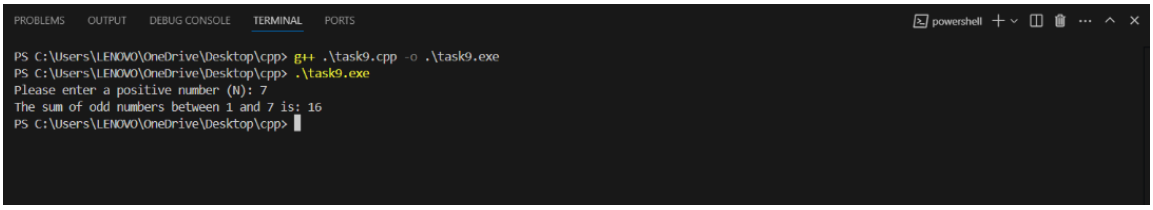

Figure 18: compile a code

Lab task 10:

Code:

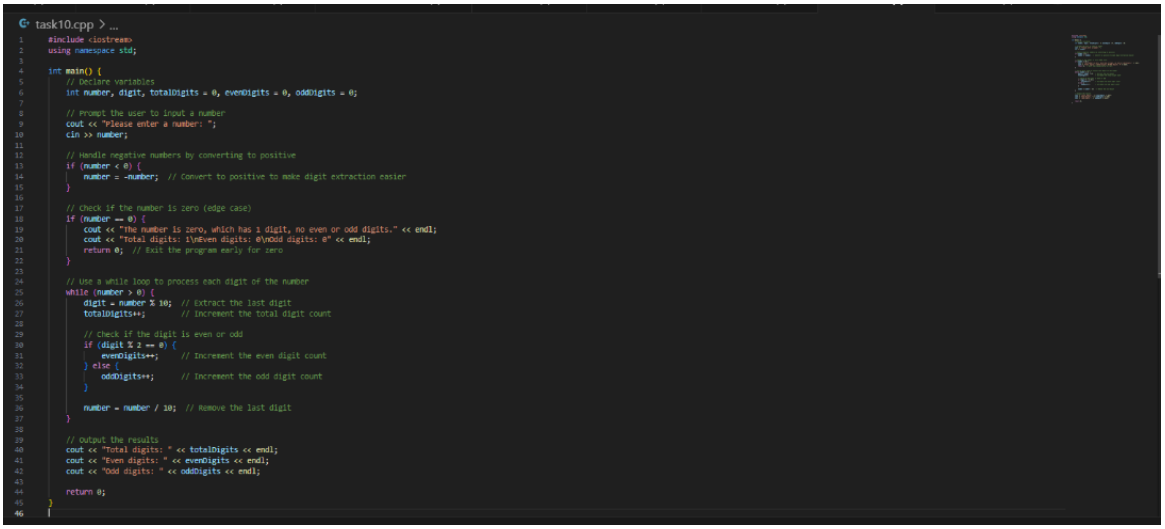

Figure 19: code of identify even and odd

compiler:

Figure 20: compile a code