

**UNIVERSITY OF VOCATIONAL TECHNOLOGY**

100, Kandawala Road, Ratmalana

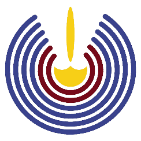
Bachelor of Technology in Software Technology

Computer Programming

1st Semester

Assignment 01

|  |
| --- |
| M.N.M. Nifraz |
| SOF/21/B2/21 |

**UNIVERSITY OF VOCATIONAL TECHNOLOGY**

**SRI LANKA**

Assignment Template & Feedback Form

|  |  |  |  |
| --- | --- | --- | --- |
| Course Title | Bachelor of Technology in Software Technology | Module | IT104021 |
| Student’s Name | M.N.M. Nifraz | Batch No. | B2 |
| Reg. No. | SOF/21/B2/21 | Semester | 01 |
| Resource Person | Mr. Nishantha Anuruddha | Assignment No. | 01 |
| Issued on | 21/06/2022 |  |  |
| Submitted on | 08/07/2022 | Received By |  |

Feedback from the Resource Person:

|  |  |
| --- | --- |
| Key Strength: |  |
| To Improve Report: |  |
| General Comments: |  |

Grade:

Date: …………………………………………………

………………………………………………

Resource Person’s Signature

………………………………………………

Resource Person’s Signature

1. Ask User to enter length & unit of measure of a cube. Print perimeter of a side, volume & surface area.

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

// WinAPI - to support exponent symbols

#include <windows.h>

using namespace std;

int main()

{

     // set console output code page to CP-1252

     // reference - http://zuga.net/articles/text-ascii-vs-cp-1252-vs-cp-437/

     SetConsoleOutputCP(1252);

     cout << "Qubic Measurements Calculator\n"

          << setfill('=') << setw(30) << "\n";

     // read length

     cout << "Enter the length of a side : ";

     double length;

     cin >> length;

     if (length <= 0) // check if length is negative or 0

     {

          cout << "Length must be positive!\n"

               << "The program will exit now...";

          return 0; // return main

     }

     // read unit

     cout << "Enter the measurement unit ( m | cm | mm ) : ";

     string unit;

     cin >> unit;

     // calculations

     double squarePerimeter = length \* 4;         // calculate perimeter

     double cubeVolume = pow(length, 3);          // calculate volume

     double cubeSurfaceArea = pow(length, 2) \* 6; // calculate surface area

     // print results (x)² = (char)178, (x)³ = (char)179

     cout << "\n"

          << "Results\n"

          << setfill('=') << setw(8) << "\n";

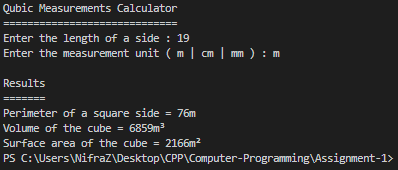
     cout << "Perimeter of a square side = " << squarePerimeter << unit << "\n"

          << "Volume of the cube = " << cubeVolume << unit << (char)179 << "\n"

          << "Surface area of the cube = " << cubeSurfaceArea << unit << (char)178 << "\n";

}

Sample Output:



2. Write a program that accepts an integer (n) and computes the value of n+nn+nnn+nnnn+nnnnn.

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    // read digit

    cout << "Enter a digit (1-9) : ";

    int digit;

    cin >> digit;

    if (digit < 1 || digit > 9) // check if valid single digit

    {

        cout << "Input must be a valid single digit (1-9)!\n"

             << "The program will exit now...";

        return 0; // return main

    }

    cout << "\n";

    int term;

    int sum = 0;

    // generate 5 terms

    for (int i = 1; i <= 5; i++)

    {

        // build term

        term = 0;

        for (int j = 0; j < i; j++)

        {

            term += digit \* pow(10, j);

        }

        // print term

        cout << "+" << setw(8) << term << "\n";

        sum += term; // add term to sum

    }

    // print result

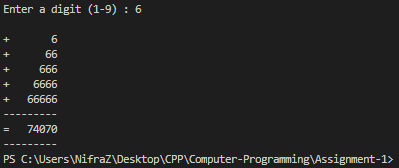
    cout << setfill('-') << setw(10) << "\n"

         << setfill(' ') << "=" << setw(8) << sum << "\n"

         << setfill('-') << setw(10) << "\n";

}

Sample Output:



3. User will enter a value (year). Print if that is a leap year or not.

a. Any number which can be divided by 4 is a leap year (2016, 2020, 2024)

b. Except if that can be divided by 100 (2100, 2200, 2300 are not leap year)

c. But if that is possible to divided by 400 then it is a leap year (2000, 2400, 2800)

d. Why this is → Earth will take 365.242375 days to rotate, but we say 365 days for a year

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    cout << "Leap Year Checker\n"

         << setfill('=') << setw(18) << "\n";

    // read year

    cout << "Enter the year : ";

    int year;

    cin >> year;

    if (year <= 0) // check if year is negative or 0

    {

        cout << "Year must be positive!\n"

             << "The program will exit now...";

        return 0; // return main

    }

    bool isLeapYear = year % 4 == 0; // year must be divided by 4

    isLeapYear &= year % 100 != 0;   // and not divided by 100

    isLeapYear |= year % 400 == 0;   // or if divided by 400

    // print results

    cout << "\n"

         << year;

    if (isLeapYear) // if leap year

    {

        cout << " is a leap year.";

    }

    else

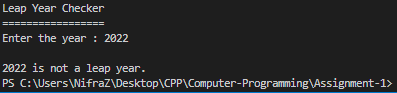
    {

        cout << " is not a leap year.";

    }

}

Sample Output:



4. Create a program to build a simple calculator using switch Statement. Initially, user will enter 2 values. Then ask for the operation (+,-,\*,/,%). Then print the result. Handle the errors (like % can’t use with doubles…).

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    cout << "Simple Calculator\n"

         << setfill('=') << setw(18) << "\n";

    // read number 1

    cout << "Enter Number 1 : ";

    double number1;

    cin >> number1;

    // read number 2

    cout << "Enter Number 2 : ";

    double number2;

    cin >> number2;

    // read operator symbol

    cout << "Enter the symbol for operation ( + | - | \* | / | % ) : ";

    char operatorSymbol;

    cin >> operatorSymbol;

    // print answer

    cout << "\n"

         << "Answer\n"

         << setfill('=') << setw(7) << "\n";

    switch (operatorSymbol)

    {

    case '+': // addition

        cout << number1 << " + " << number2 << " = " << (number1 + number2);

        break;

    case '-': // subtraction

        cout << number1 << " - " << number2 << " = " << (number1 - number2);

        break;

    case '\*': // multiplication

        cout << number1 << " \* " << number2 << " = " << (number1 \* number2);

        break;

    case '/': // division

    {

        if (number2 != 0) // check if divisor is not 0

        {

            cout << number1 << " / " << number2 << " = " << (number1 / number2);

        }

        else

        {

            cout << "Divisor (Number 2) cannot be zer0!";

        }

    }

    break;

    case '%': // modulus

    {

        int number1IntValue = (int)number1; // convert number 1 to int

        int number2IntValue = (int)number2; // convert number 2 to int

        // check if both the numbers are integers

        bool isInteger = (number1 == number1IntValue) && (number2 == number2IntValue);

        if (isInteger)

        {

            cout << number1IntValue << " % " << number2IntValue << " = " << (number1IntValue % number2IntValue);

        }

        else

        {

            cout << "Numbers must be integers to perform Modulus operation!";

        }

    }

    break;

    default: // other

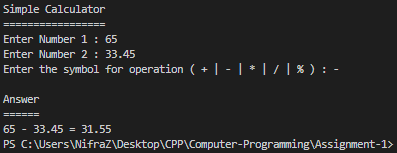
        cout << operatorSymbol << " - Invalid operator symbol!";

        break;

    }

}

Sample Output:



5. Find the lucky number by taking user’s birthday as an input. Use following format to input birthday YYYYMMDD -19850217. → lucky number is 6.

(19850217 → 1 + 9 + 8 +5+0+2+1+7= 33 → 3+3 = 6)

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    cout << "Lucky Number Calculator\n"

         << setfill('=') << setw(24) << "\n";

    // read birthday

    cout << "Enter your birthday (YYYYMMDD) : ";

    int birthdayNumber;

    cin >> birthdayNumber;

    if (birthdayNumber <= 0) // check if number is negative or 0

    {

        cout << "Number must be positive!\n"

             << "The program will exit now...";

        return 0; // return main

    }

    int digit;

    int luckyNumber = 0;

    while (birthdayNumber > 0)

    {                                // repeat until 0

        digit = birthdayNumber % 10; // get last digit from bd no.

        luckyNumber += digit;        // add digit to lucky no.

        birthdayNumber /= 10;        // remove last digit off bd no.

        // check if lucky number has more than one digit

        // after adding all digits from bd no.

        if (birthdayNumber < 1 && luckyNumber > 9)

        {

            // assign lucky no. to bd no. and reset lucky no. to 0

            // then continue loop to start adding the digits again

            birthdayNumber = luckyNumber;

            luckyNumber = 0;

        }

    }

    // print lucky no.

    cout << "\n"

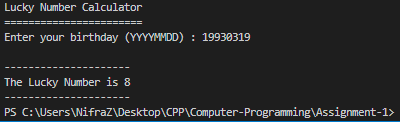
         << setfill('-') << setw(22) << "\n"

         << "The Lucky Number is " << luckyNumber << "\n"

         << setfill('-') << setw(22) << "\n";

}

Sample Output:



6. There are 45 students in a classroom & 25 are boys. 80% of the total students has passed the exam. Also, it says 2 girls are failed. Now find out how many boys has passed the exam.

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    // perform calculations

    int totalStudents = 45;

    int boys = 25;

    int girls = totalStudents - boys;

    int passedStudents = totalStudents \* 80 / 100;

    int failedStudents = totalStudents - passedStudents;

    int failedGirls = 2;

    int passedGirls = girls - failedGirls;

    int passedBoys = passedStudents - passedGirls;

    int failedBoys = boys - passedBoys;

    // print results

    cout << "Total Students : " << totalStudents << "\n"

         << "\t> passed : " << passedStudents << "\n"

         << "\t> failed : " << failedStudents << "\n"

         << "Total Boys : " << boys << "\n"

         << "\t> passed : " << passedBoys << "\n"

         << "\t> failed : " << failedBoys << "\n"

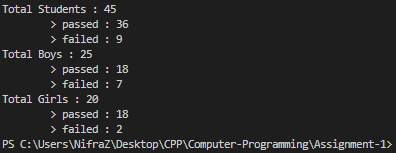
         << "Total Girls : " << girls << "\n"

         << "\t> passed : " << passedGirls << "\n"

         << "\t> failed : " << failedGirls << "\n";

}

Sample Output:



7. Write a program that accept integers from user. Whenever user enter 0 it prints the output & exit the program. If the input is greater than 10 add reminder of 10 to final answer, if the input is even number add 2 more final answer. If the input is odd add 1 more to final answer. If number is negative ignore it. If number is divisible by 3 add 3 to final answer. Any positive number will be added to final answer. Design the algorithm to support above scenario & create a program.

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    cout << "(Enter 0 to show the final answer)\n";

    int integerInput;

    int finalAnswer = 0;

    int count = 1;

    do

    {

        // read integer

        cout << "Enter Integer #" << count++ << " : ";

        cin >> integerInput;

        if (integerInput > 0) //check if positive

        {

            if (integerInput > 10) //check if greater than 10

            {

                //add modulus remainder of 10

                finalAnswer += integerInput % 10;

            }

            if (integerInput % 2 == 0) //if even

            {

                finalAnswer += 2; //add 2

            }

            else //if odd

            {

                finalAnswer += 1; //add 1

            }

            if (integerInput % 3 == 0) //if divisible by 3

            {

                finalAnswer += 3; // add 3

            }

        }

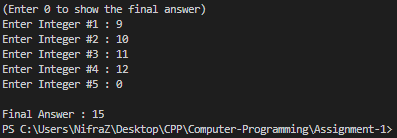
    } while (integerInput); // loop until user enters 0

    cout << "\n"

         << "Final Answer : " << finalAnswer;

}

Sample Output:



8. There is a requirement to create a new Student Evaluation System for a school to verify students’ marks by themselves. First, system will ask username (String) & student ID (number). Then it asks you to enter marks for 3 subjects. Then system should print some messages according to entered vales. If student scores less than 50 for any subject, then he is repeated all 3 subjects (Means that he has got failed the exam). Therefore, print whether he should do repeat exam or not. Then, if average mark is higher than 75 then print “Very Good”, 50 ~ 74 print “Good”, 35~49 print “average” otherwise, print “Fail”. Also, if he scored above 90 for all subjects, he would get a first class. System should handle errors, like if user enter invalid marks (-10, 120) for a given subject, system will show error message and ask to enter again. But if the 2nd time also user entered an invalid number, then system shows an error message & value will be set to zero for that subject automatically. Finally print the details in “nice” way. (You may or may not use array support)

Source Code:

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main()

{

    cout << "Student Evaluation System\n"

         << setfill('=') << setw(26) << "\n";

    // read username

    cout << "Enter your username : ";

    string username;

    cin >> username;

    // read student id

    cout << "Enter your student ID : ";

    int studentId;

    cin >> studentId;

    int const SUBJECT\_COUNT = 3;

    int marks[SUBJECT\_COUNT]; // int array for subjects

    int sum = 0;

    bool isPassed = true;

    bool isFirstClass = true;

    cout << "Enter your marks :\n";

    for (int i = 0; i < SUBJECT\_COUNT; i++)

    {

        // infinite loop

        while (true)

        {

            // read marks

            cout << "\tSubject #" << (i + 1) << " : ";

            cin >> marks[i];

            // if valid marks (0 - 100)

            if (marks[i] >= 0 && marks[i] <= 100)

            {

                break; // break loop

            }

            else

            {

                // print error message and continue

                cout << "Invalid marks! Please enter a value between 0 - 100.\n";

            }

        }

        // add to total

        sum = sum + marks[i];

        // if marks greater than or equal to 50 in all 3 subjects

        isPassed &= (marks[i] >= 50);

        // if marks greater than or equal to 90 in all 3 subjects

        isFirstClass &= (marks[i] >= 90);

    }

    // print results

    cout << "\n"

         << "Your Results\n"

         << setfill('=') << setw(13) << "\n";

    cout << "Username : " << username << "\n"

         << "Student ID : " << studentId << "\n"

         << "Marks :\n";

    for (int i = 0; i < SUBJECT\_COUNT; i++)

    {

        cout << "\tSubject #" << (i + 1) << " : " << marks[i] << "\n";

    }

    if (isPassed) // if passed

    {

        cout << "You have passed the exam.\n";

    }

    else // if failed

    {

        cout << "You have failed the exam.\n"

             << "You should repeat all " << SUBJECT\_COUNT << " subjects!\n";

    }

    if (isFirstClass) // if first class

    {

        cout << "Congratz! You got first class. :)\n";

    }

    // calculate and print average

    double average = (double)sum / SUBJECT\_COUNT;

    cout << "You have the average of " << average << ".\n";

    // print grade

    cout << "You have received the grade of ";

    if (average >= 75)

    {

        cout << "Very Good.";

    }

    else if (average >= 50)

    {

        cout << "Good.";

    }

    else if (average >= 35)

    {

        cout << "Average.";

    }

    else

    {

        cout << "Fail.";

    }

}

Sample Output:

