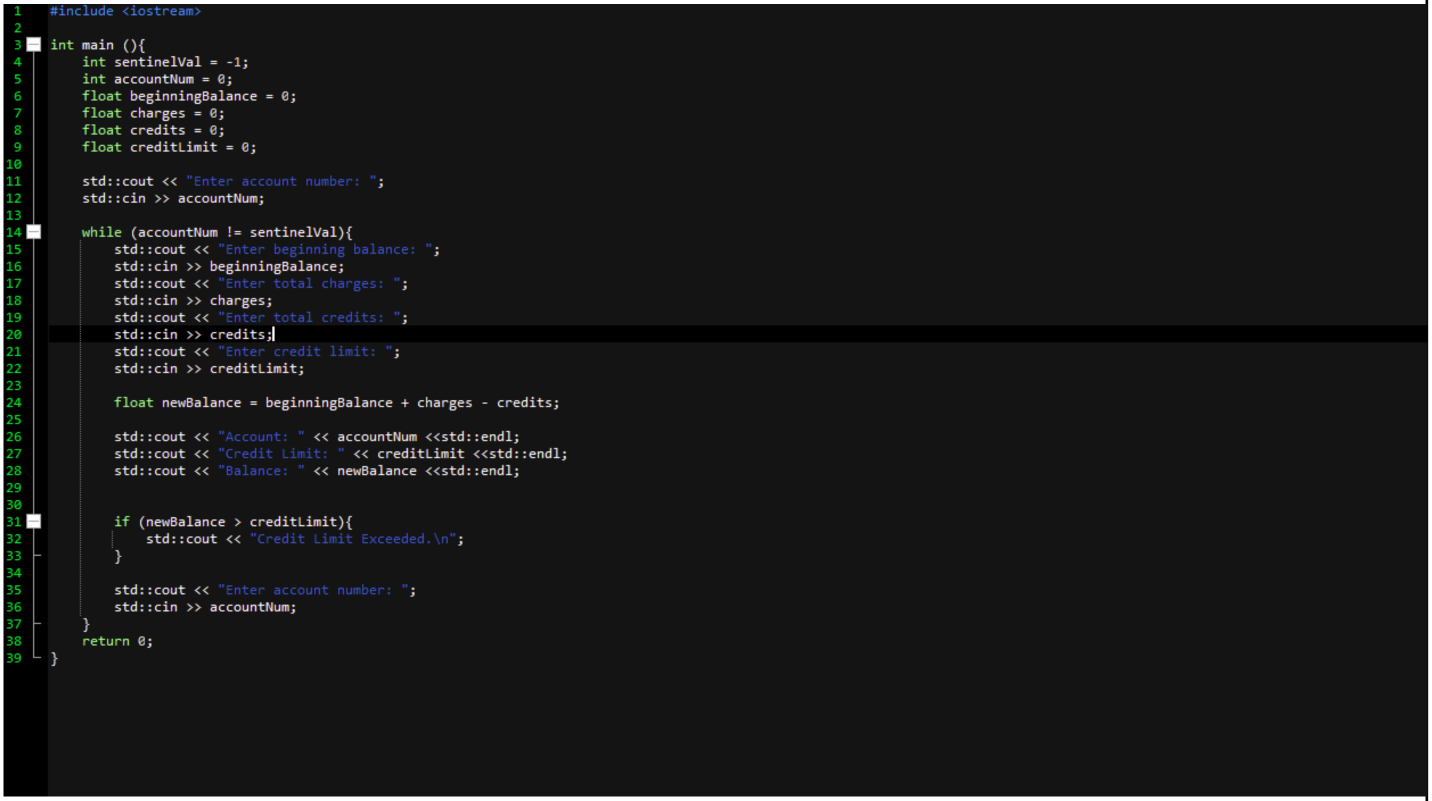
Activity No. 3.1	
Hands-on Activity 3.1: Control Structures (part 2)	
Course Code: CPE007	**Program:** Computer Engineering
Course Title: Programming Logic and Design	**Date Performed:8/12/25**
Section:CPE11S1	**Date Submitted:**
Name(s): Niel Vincent B. Condino	**Instructor: Engr. Jimlord M. Quejado**
6. Output	

1.Develop a C++ program that will determine if a department store customer has exceeded the credit limit on a charge account. For each customer, the following facts are available:

1. Account number
2. Balance at the beginning of the month
3. Total of all items charged by this customer this month
4. Total of all credits applied to this customer’s account this month
5. Allowed credit limit

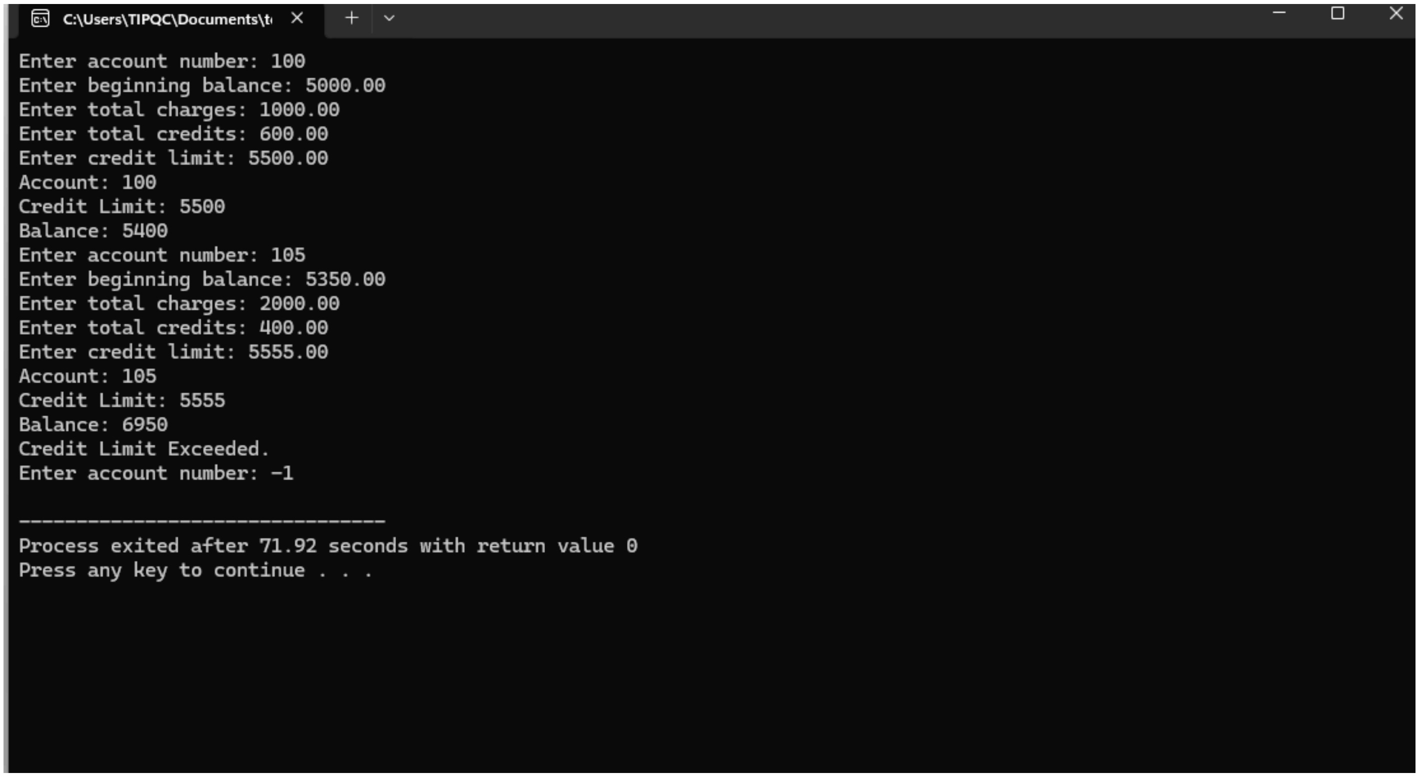
The program should input each of these facts, calculate the new balance (=*beginning balance +charges -credits),* and determine if the new balance exceeds the customer’s credit limit. For those customers whose credit limit is exceeded, the program should display the customer’s account number, credit limit, new balance and the message “Credit limit exceeded.”

Code:



```
#include <iostream>

int main (){
    int sentinelVal = -1;
    int accountNum = 0;
    float beginningBalance = 0;
    float charges = 0;
    float credits = 0;
    float creditLimit = 0;

    std::cout << "Enter account number: ";
    std::cin >> accountNum;

    while (accountNum != sentinelVal){
        std::cout << "Enter beginning balance: ";
        std::cin >> beginningBalance;
        std::cout << "Enter total charges: ";
        std::cin >> charges;
        std::cout << "Enter total credits: ";
        std::cin >> credits;
        std::cout << "Enter credit limit: ";
        std::cin >> creditLimit;

        float newBalance = beginningBalance + charges - credits;

        std::cout << "Account: " << accountNum <<std::endl;
        std::cout << "Credit Limit: " << creditLimit <<std::endl;
        std::cout << "Balance: " << newBalance <<std::endl;


        if (newBalance > creditLimit){
            std::cout << "Credit Limit Exceeded.\n";
        }

        std::cout << "Enter account number: ";
        std::cin >> accountNum;
    }
    return 0;
}
```

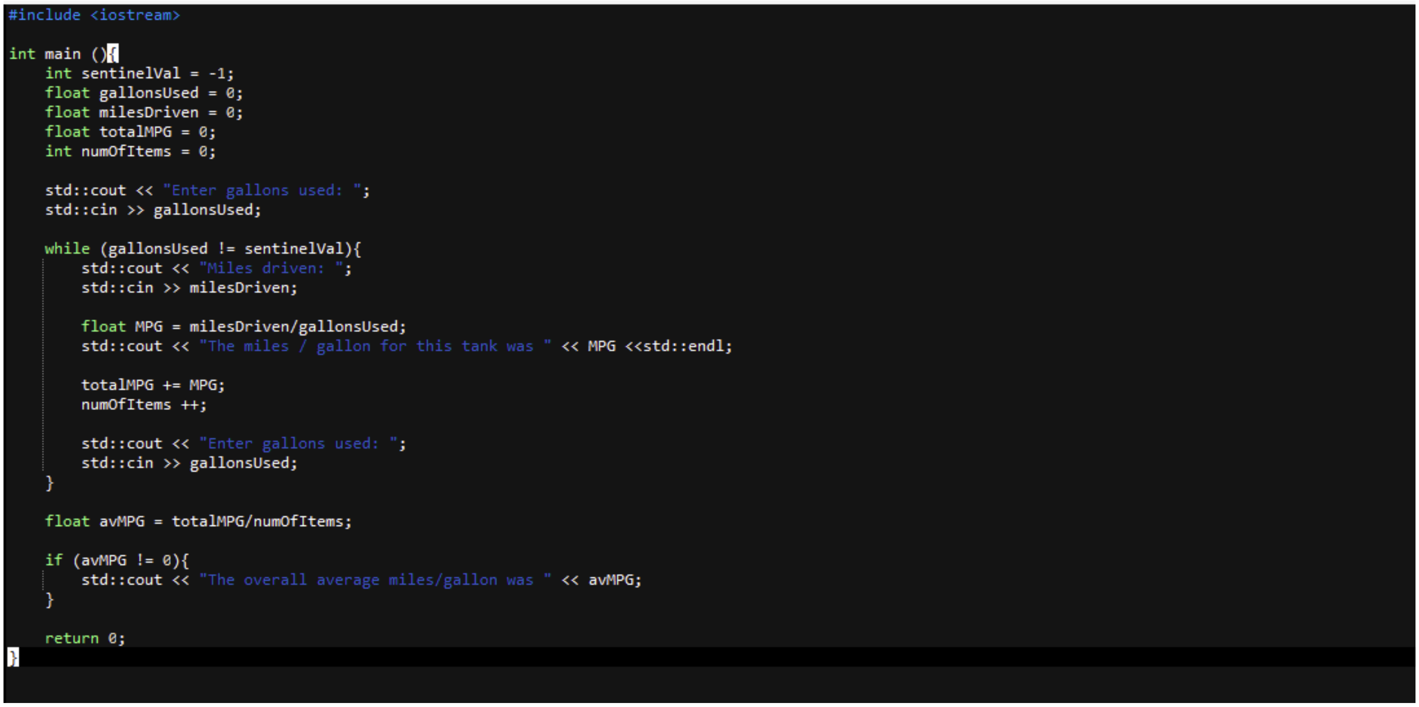
Output:



```
Enter account number: 100
Enter beginning balance: 5000.00
Enter total charges: 1000.00
Enter total credits: 600.00
Enter credit limit: 5500.00
Account: 100
Credit Limit: 5500
Balance: 5400
Enter account number: 105
Enter beginning balance: 5350.00
Enter total charges: 2000.00
Enter total credits: 400.00
Enter credit limit: 5555.00
Account: 105
Credit Limit: 5555
Balance: 6950
Credit Limit Exceeded.
Enter account number: -1

--------------------------------
Process exited after 71.92 seconds with return value 0
Press any key to continue . . .
```

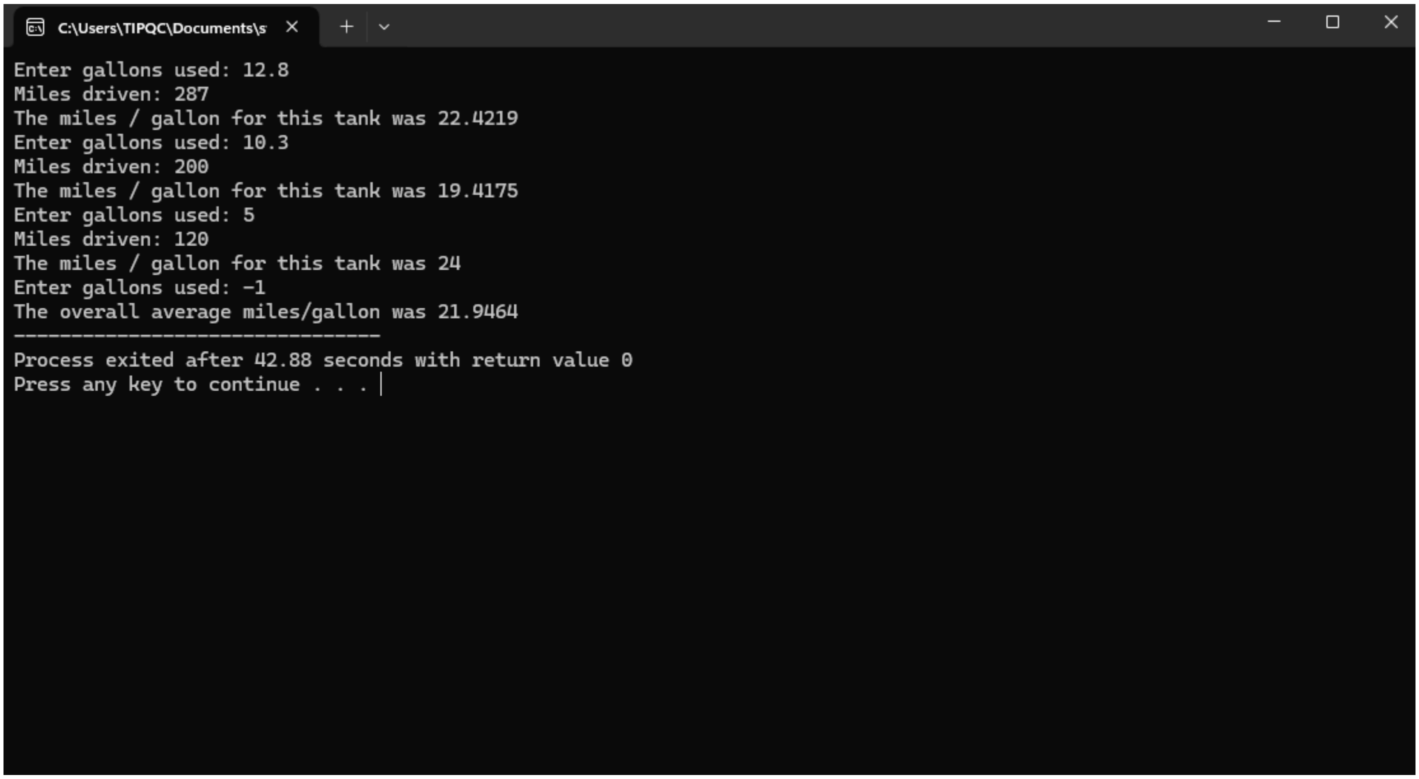
2.Because of the price of gasoline, drivers are concerned with the mileage obtained by their automobiles. One driver has kept track of several tankfuels of gasoline by recording miles driven and gallons used for each tankful. Develop a program that will input the miles driven and gallons used for each tankful. The program should calculate and display the miles per gallon obtained for each tankful. After processing all input information, the program should calculate and print the combined miles per gallon obtained for all tank fuels.

Code:



```
#include <iostream>

int main (){
    int sentinelVal = -1;
    float gallonsUsed = 0;
    float milesDriven = 0;
    float totalMPG = 0;
    int numOfItems = 0;

    std::cout << "Enter gallons used: ";
    std::cin >> gallonsUsed;

    while (gallonsUsed != sentinelVal){
        std::cout << "Miles driven: ";
        std::cin >> milesDriven;

        float MPG = milesDriven/gallonsUsed;
        std::cout << "The miles / gallon for this tank was " << MPG <<std::endl;

        totalMPG += MPG;
        numOfItems ++;

        std::cout << "Enter gallons used: ";
        std::cin >> gallonsUsed;
    }

    float avMPG = totalMPG/numOfItems;

    if (avMPG != 0){
        std::cout << "The overall average miles/gallon was " << avMPG;
    }

    return 0;
}
```

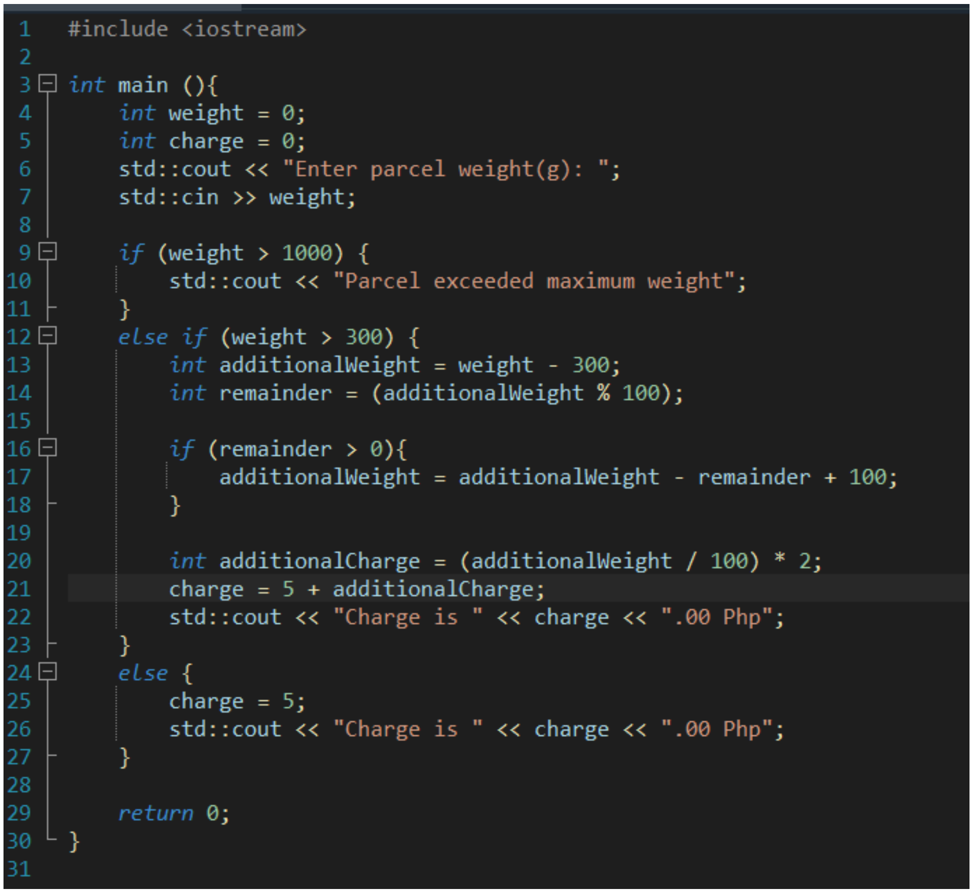
Output:



```
Enter gallons used: 12.8
Miles driven: 287
The miles / gallon for this tank was 22.4219
Enter gallons used: 10.3
Miles driven: 200
The miles / gallon for this tank was 19.4175
Enter gallons used: 5
Miles driven: 120
The miles / gallon for this tank was 24
Enter gallons used: -1
The overall average miles/gallon was 21.9464
--------------------------------
Process exited after 42.88 seconds with return value 0
Press any key to continue . . .
```

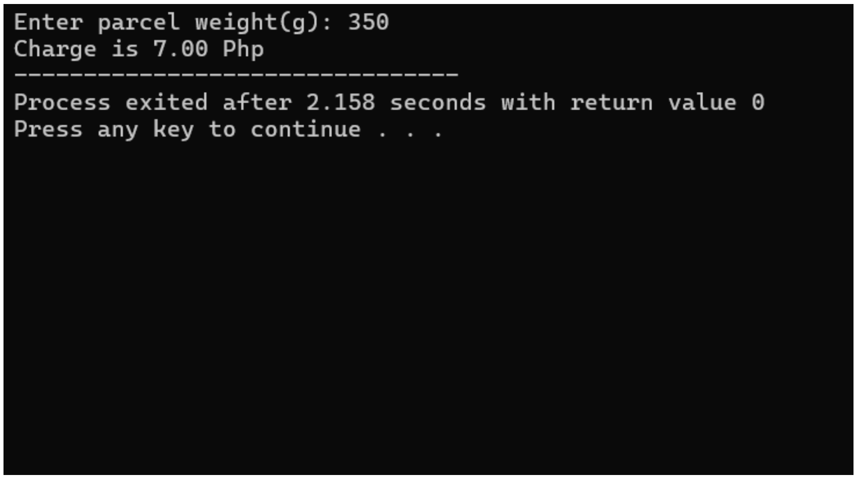
3.Create a program that will calculate the cost of sending a small parcel. The post office charges P5.00 for the first 300g, and P2.00 for every 100g thereafter (rounded up), up to a maximum weight of 1000g.

Code:

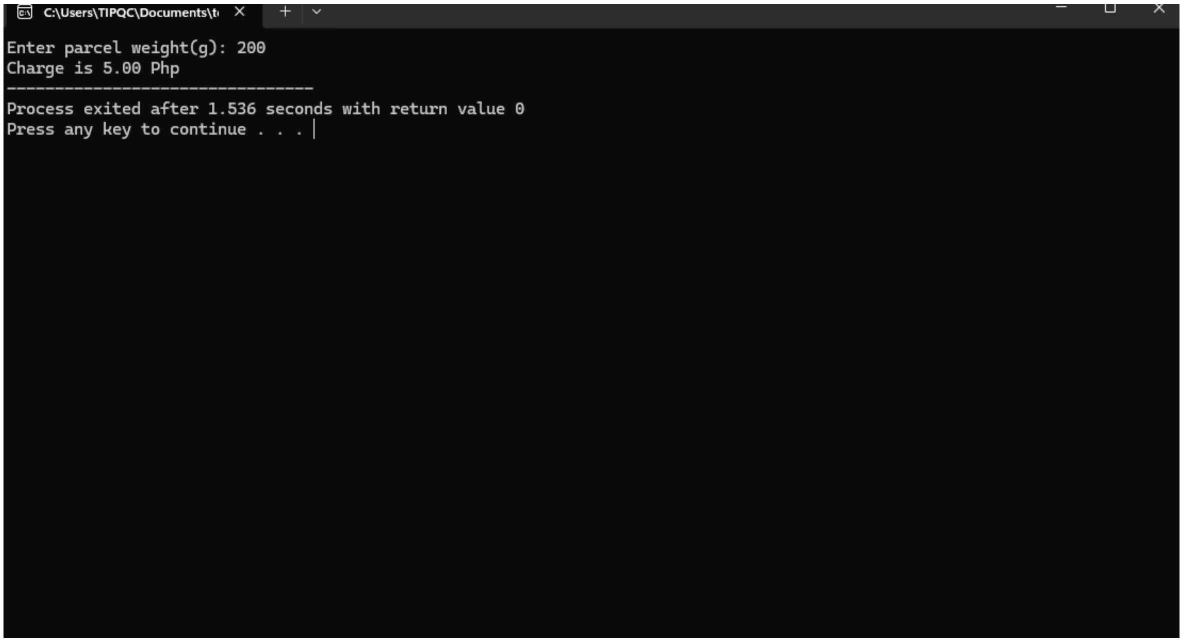


```
#include <iostream>

int main (){
    int weight = 0;
    int charge = 0;
    std::cout << "Enter parcel weight(g): ";
    std::cin >> weight;

    if (weight > 1000) {
        std::cout << "Parcel exceeded maximum weight";
    }
    else if (weight > 300) {
        int additionalWeight = weight - 300;
        int remainder = (additionalWeight % 100);

        if (remainder > 0){
            additionalWeight = additionalWeight - remainder + 100;
        }

        int additionalCharge = (additionalWeight / 100) * 2;
        charge = 5 + additionalCharge;
        std::cout << "Charge is " << charge << ".00 Php";
    }
    else {
        charge = 5;
        std::cout << "Charge is " << charge << ".00 Php";
    }

    return 0;
}

```

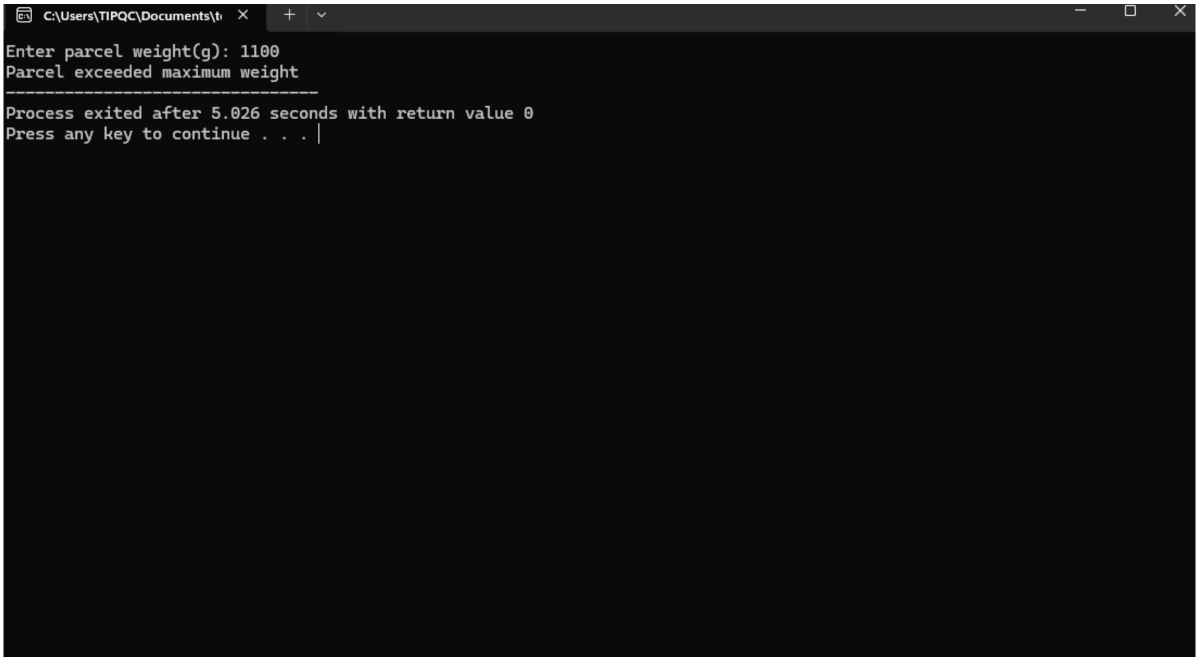
Output:



```
Enter parcel weight(g): 350
Charge is 7.00 Php
--------------------------------
Process exited after 2.158 seconds with return value 0
Press any key to continue . . .
```



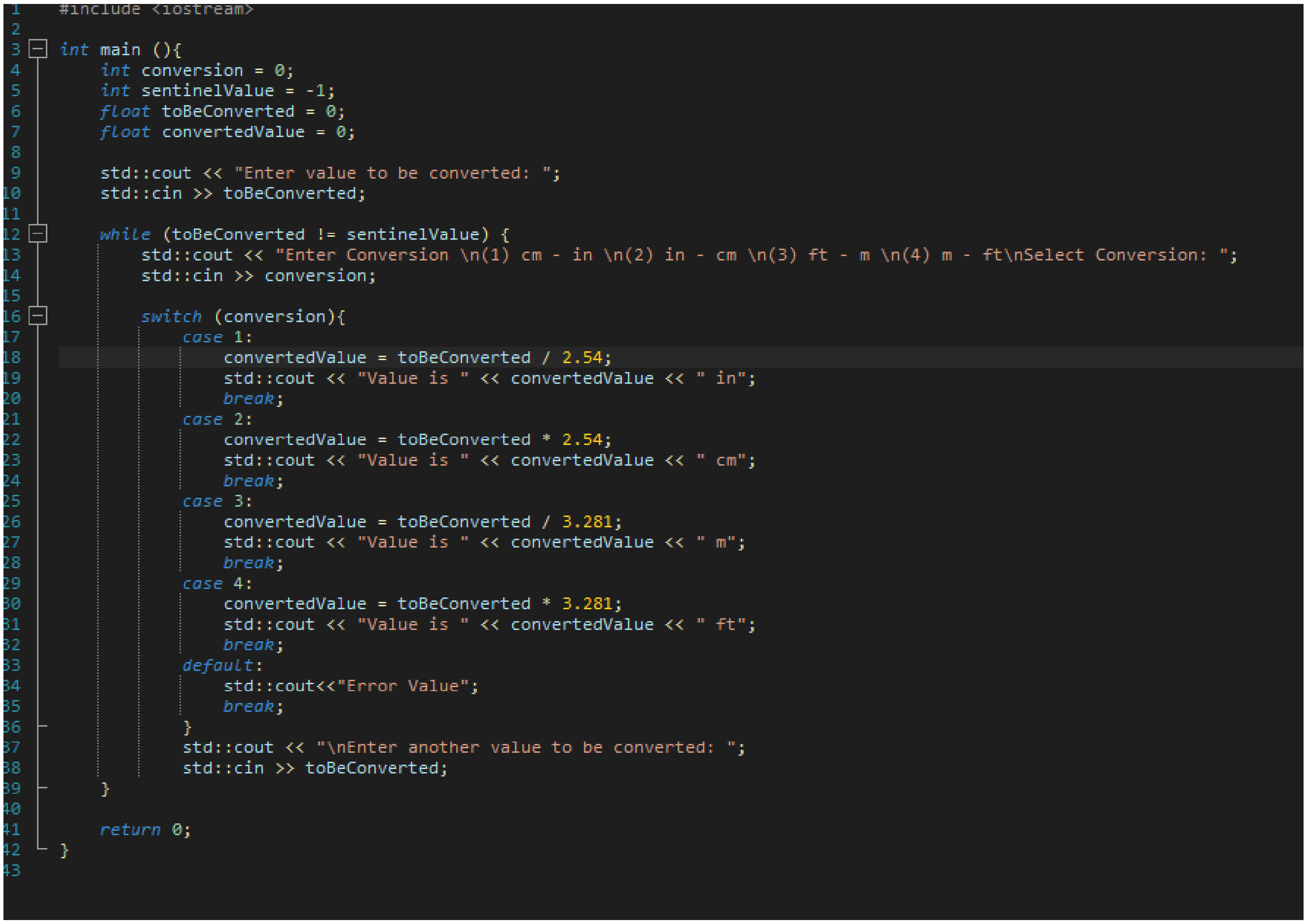
```
Enter parcel weight(g): 200
Charge is 5.00 Php
--------------------------------
Process exited after 1.536 seconds with return value 0
Press any key to continue . . .
```



```
Enter parcel weight(g): 1100
Parcel exceeded maximum weight
--------------------------------
Process exited after 5.026 seconds with return value 0
Press any key to continue . . .
```

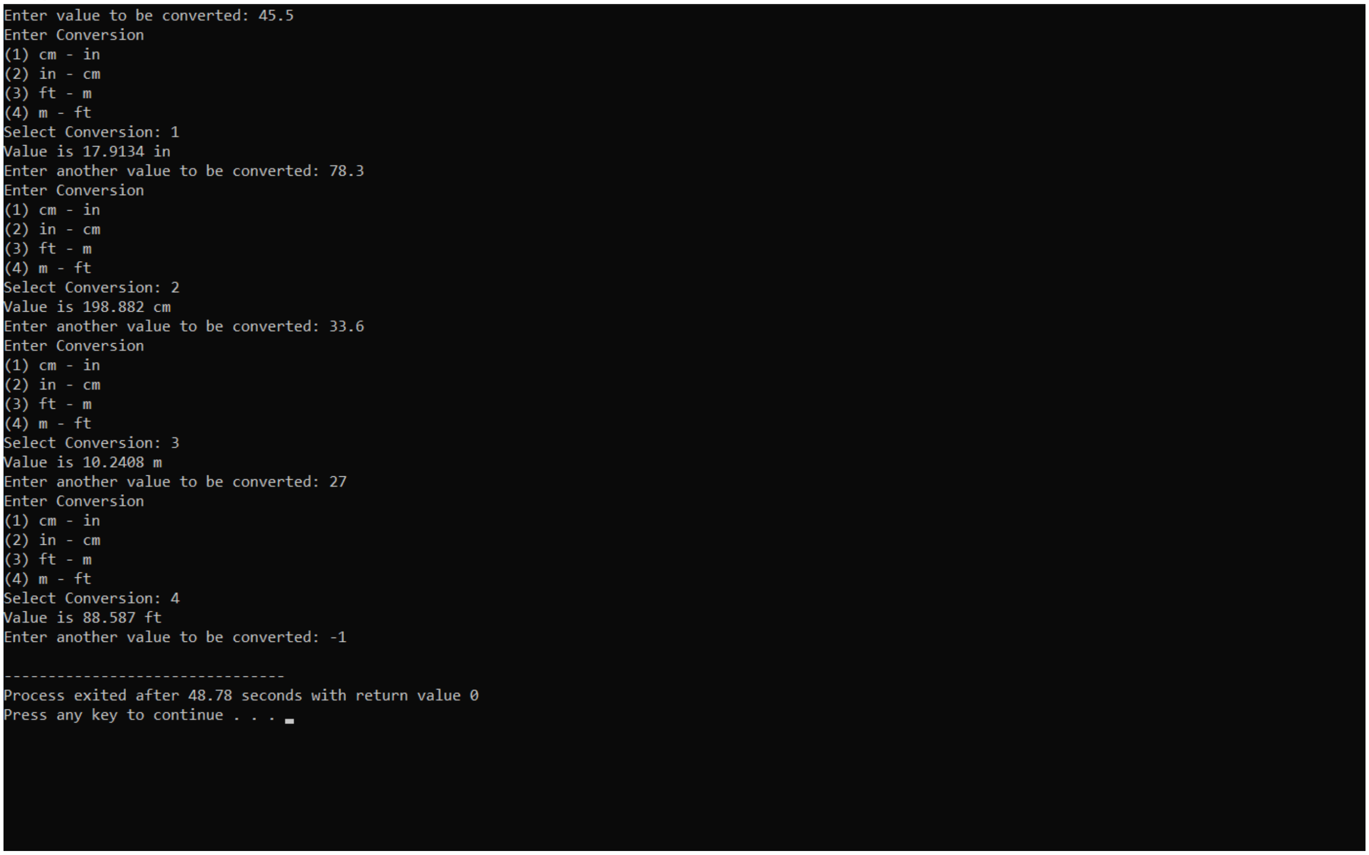
4.

Code:



```cpp
#include <iostream>

int main (){
    int conversion = 0;
    int sentinelValue = -1;
    float toBeConverted = 0;
    float convertedValue = 0;

    std::cout << "Enter value to be converted: ";
    std::cin >> toBeConverted;

    while (toBeConverted != sentinelValue) {
        std::cout << "Enter Conversion \n(1) cm - in \n(2) in - cm \n(3) ft - m \n(4) m - ft\nSelect Conversion: ";
        std::cin >> conversion;

        switch (conversion){
            case 1:
                convertedValue = toBeConverted / 2.54;
                std::cout << "Value is " << convertedValue << " in";
                break;
            case 2:
                convertedValue = toBeConverted * 2.54;
                std::cout << "Value is " << convertedValue << " cm";
                break;
            case 3:
                convertedValue = toBeConverted / 3.281;
                std::cout << "Value is " << convertedValue << " m";
                break;
            case 4:
                convertedValue = toBeConverted * 3.281;
                std::cout << "Value is " << convertedValue << " ft";
                break;
            default:
                std::cout<<"Error Value";
                break;
        }
        std::cout << "\nEnter another value to be converted: ";
        std::cin >> toBeConverted;
    }

    return 0;
}
```

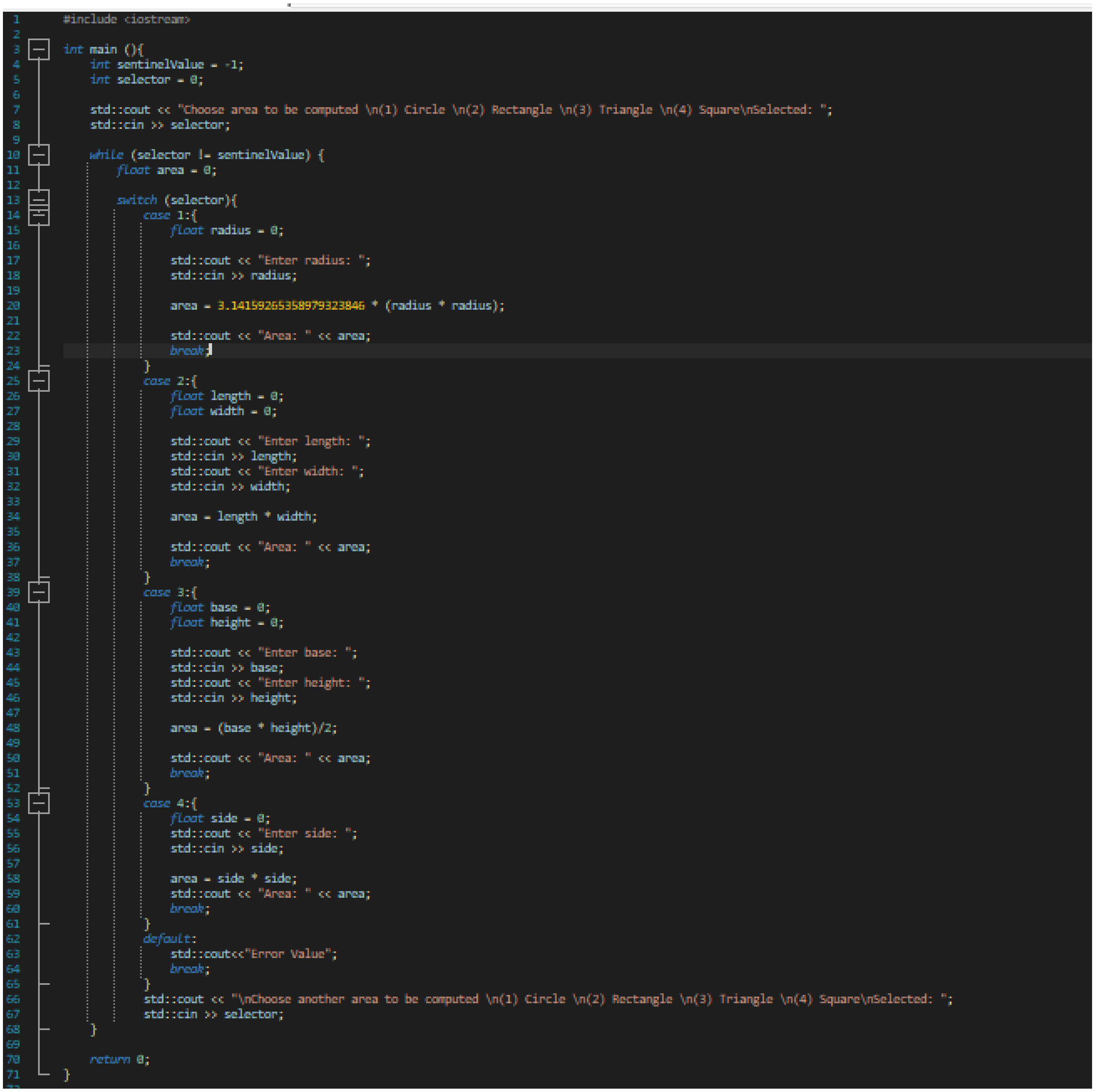
Output:



```
Enter value to be converted: 45.5
Enter Conversion
(1) cm - in
(2) in - cm
(3) ft - m
(4) m - ft
Select Conversion: 1
Value is 17.9134 in
Enter another value to be converted: 78.3
Enter Conversion
(1) cm - in
(2) in - cm
(3) ft - m
(4) m - ft
Select Conversion: 2
Value is 198.882 cm
Enter another value to be converted: 33.6
Enter Conversion
(1) cm - in
(2) in - cm
(3) ft - m
(4) m - ft
Select Conversion: 3
Value is 10.2408 m
Enter another value to be converted: 27
Enter Conversion
(1) cm - in
(2) in - cm
(3) ft - m
(4) m - ft
Select Conversion: 4
Value is 88.587 ft
Enter another value to be converted: -1

--------------------------------
Process exited after 48.78 seconds with return value 0
Press any key to continue . . .
```

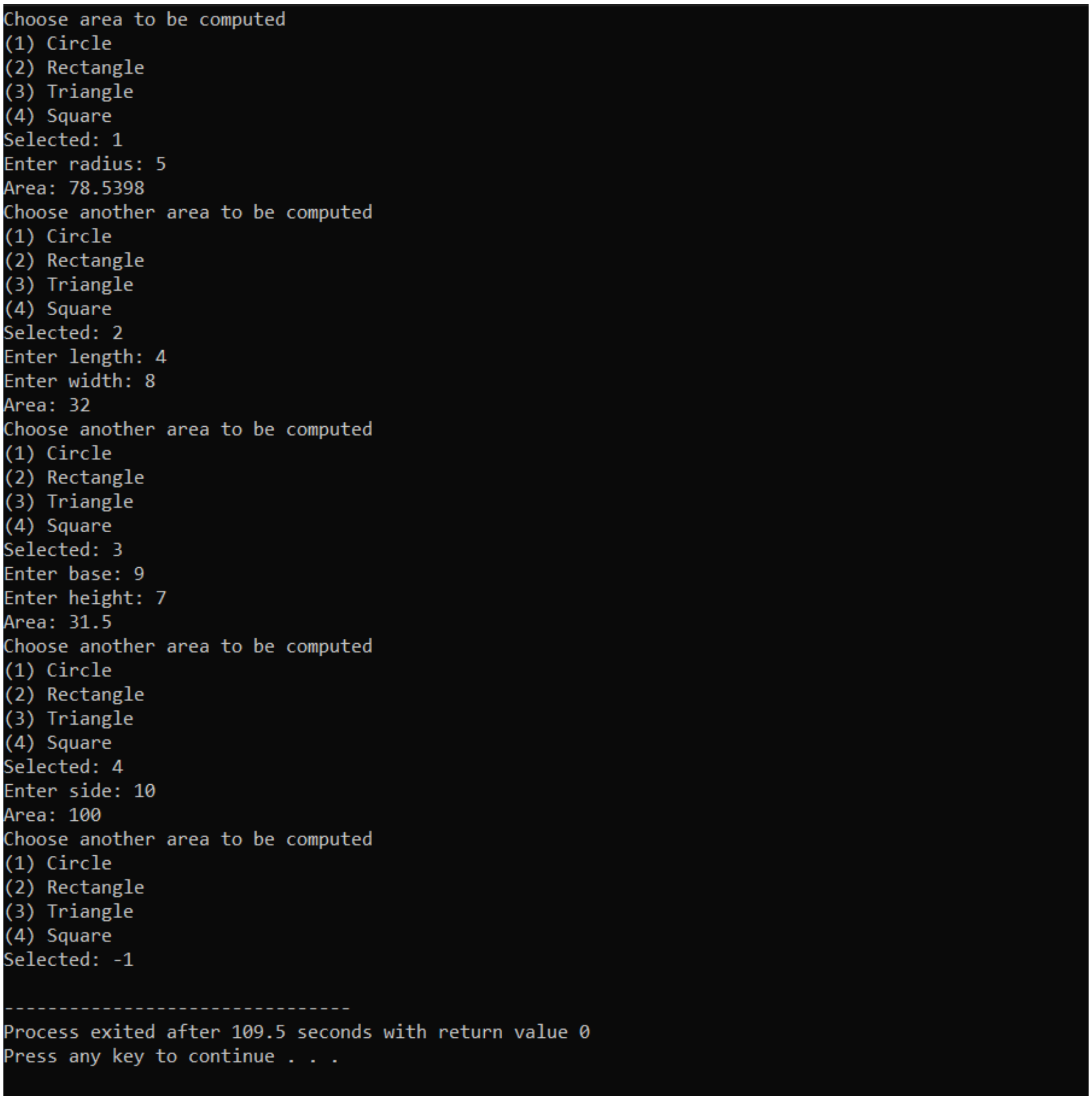
5.

Code:



```cpp
#include <iostream>

int main (){
    int sentinelValue = -1;
    int selector = 0;

    std::cout << "Choose area to be computed \n(1) Circle \n(2) Rectangle \n(3) Triangle \n(4) Square\nSelected: ";
    std::cin >> selector;

    while (selector != sentinelValue) {
        float area = 0;

        switch (selector){
            case 1:{
                float radius = 0;

                std::cout << "Enter radius: ";
                std::cin >> radius;

                area = 3.14159265358979323846 * (radius * radius);

                std::cout << "Area: " << area;
                break;
            }
            case 2:{
                float length = 0;
                float width = 0;

                std::cout << "Enter length: ";
                std::cin >> length;
                std::cout << "Enter width: ";
                std::cin >> width;

                area = length * width;

                std::cout << "Area: " << area;
                break;
            }
            case 3:{
                float base = 0;
                float height = 0;

                std::cout << "Enter base: ";
                std::cin >> base;
                std::cout << "Enter height: ";
                std::cin >> height;

                area = (base * height)/2;

                std::cout << "Area: " << area;
                break;
            }
            case 4:{
                float side = 0;
                std::cout << "Enter side: ";
                std::cin >> side;

                area = side * side;
                std::cout << "Area: " << area;
                break;
            }
            default:
                std::cout<<"Error Value";
                break;
            }
            std::cout << "\nChoose another area to be computed \n(1) Circle \n(2) Rectangle \n(3) Triangle \n(4) Square\nSelected: ";
            std::cin >> selector;
    }

    return 0;
}
```

Output:



```
Choose area to be computed
(1) Circle
(2) Rectangle
(3) Triangle
(4) Square
Selected: 1
Enter radius: 5
Area: 78.5398
Choose another area to be computed
(1) Circle
(2) Rectangle
(3) Triangle
(4) Square
Selected: 2
Enter length: 4
Enter width: 8
Area: 32
Choose another area to be computed
(1) Circle
(2) Rectangle
(3) Triangle
(4) Square
Selected: 3
Enter base: 9
Enter height: 7
Area: 31.5
Choose another area to be computed
(1) Circle
(2) Rectangle
(3) Triangle
(4) Square
Selected: 4
Enter side: 10
Area: 100
Choose another area to be computed
(1) Circle
(2) Rectangle
(3) Triangle
(4) Square
Selected: -1

--------------------------------
Process exited after 109.5 seconds with return value 0
Press any key to continue . . .
```

7. Supplementary Activity

8. Conclusion

Prior to the activity, I already knew counter controlled and sentinel repetition. I also have knowledge in the past on if else statements and while loops. But in the lessons I learned about the difference between while and do while loops. In this activity, even though I have prior knowledge on these, some of the problems need you to find a way to approach it. Some of the questions are like small puzzles to get the desired outputs.

9. Assessment Rubric