

Final Assignment

Summer 2024

Course Title: Introduction to Programming

Course Code: CSE 1102 (Summer 2024)

Submitted by:

Student Name and ID

Md. Tazminur Rahman Tanim (242014124)

Department of CSE

University of Liberal Arts Bangladesh (ULAB)

1 Question (a):

To calculate the total cost of apples, bread and milk, we can use the variables given below:

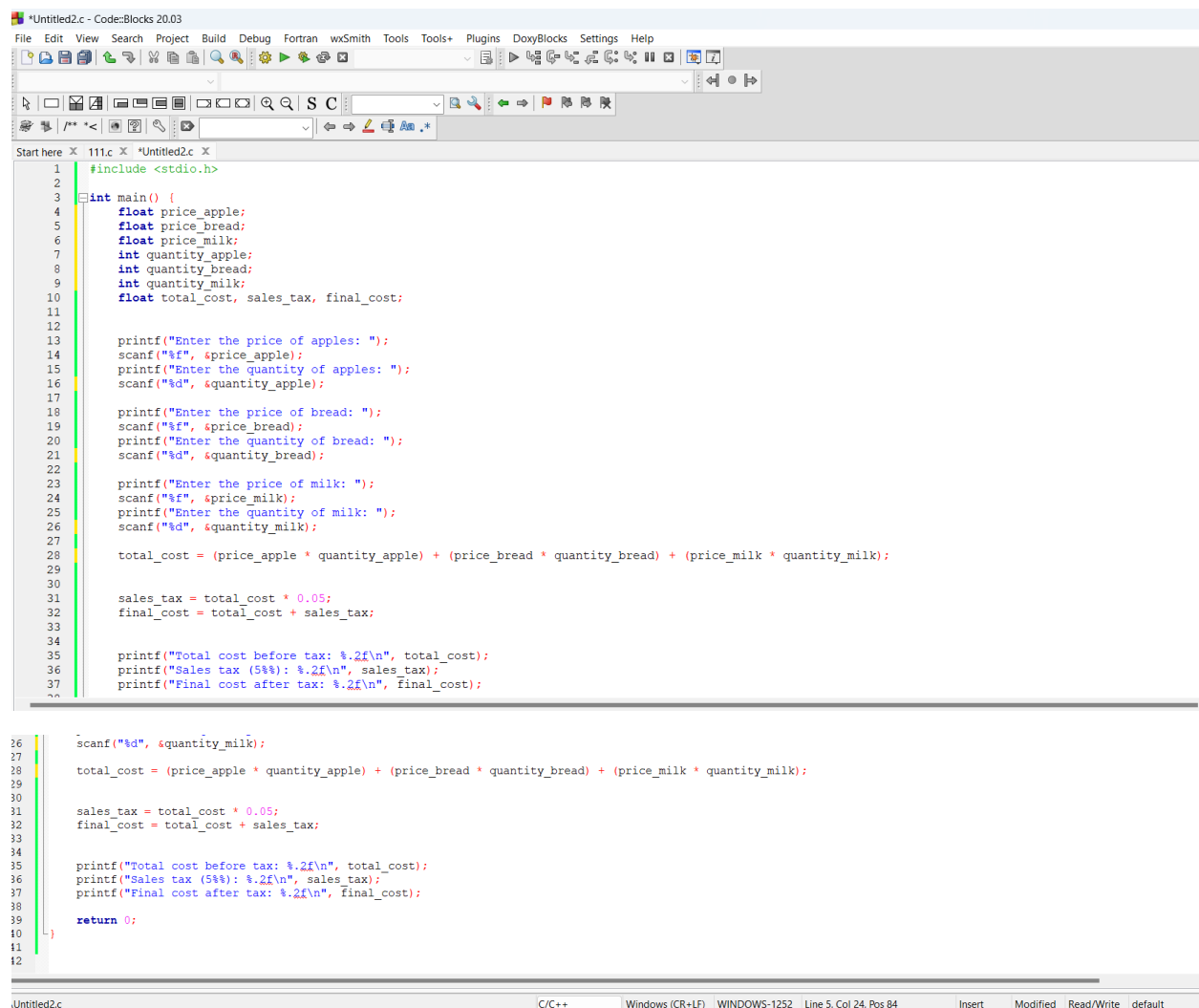
For the price of the items float, apple_cost, bread_cost, milk_cost.

For the quantity of the items, int apple_quantity, bread_quantity, milk_quantity.

The arithmetic operators to be used is given below:

1. We will use Multiplication (*) to calculate the cost. We will multiply the price with the quantity.
2. We will use Addition (+) to sum up the total cost all the items

Here's the Code,

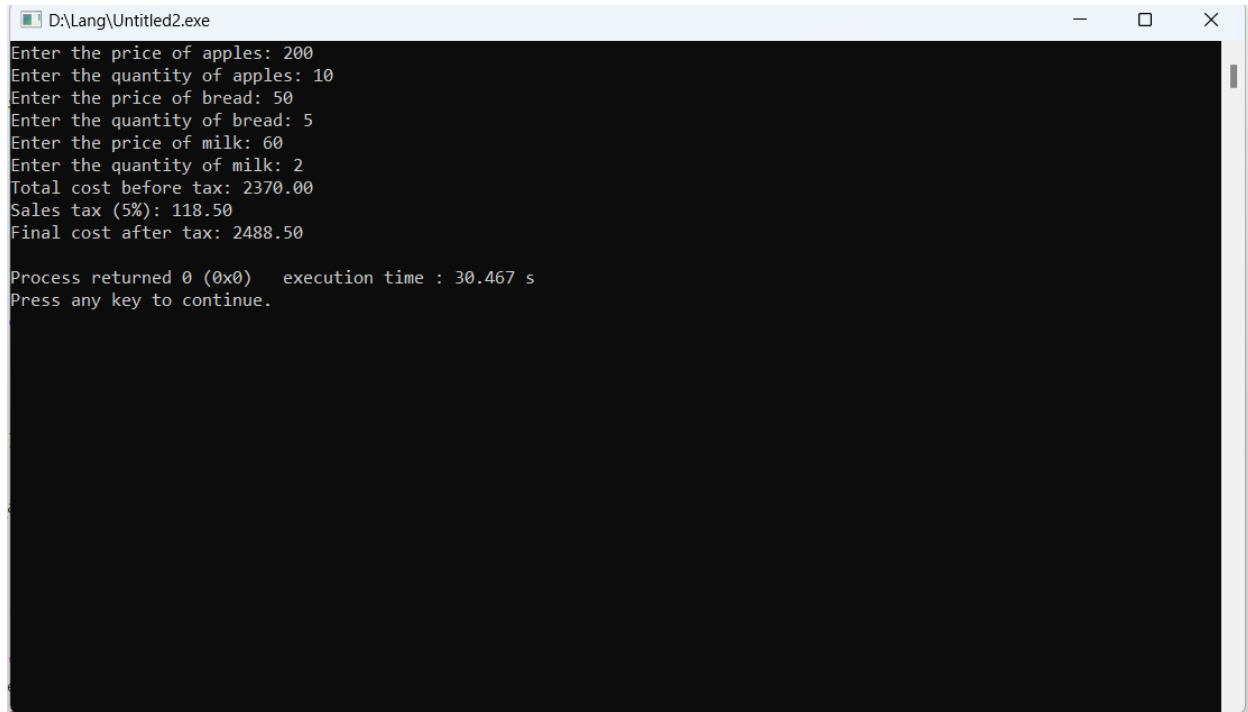


```
*Untitled2.c - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

1  #include <stdio.h>
2
3  int main() {
4      float price_apple;
5      float price_bread;
6      float price_milk;
7      int quantity_apple;
8      int quantity_bread;
9      int quantity_milk;
10     float total_cost, sales_tax, final_cost;
11
12     printf("Enter the price of apples: ");
13     scanf("%f", &price_apple);
14     printf("Enter the quantity of apples: ");
15     scanf("%d", &quantity_apple);
16
17     printf("Enter the price of bread: ");
18     scanf("%f", &price_bread);
19     printf("Enter the quantity of bread: ");
20     scanf("%d", &quantity_bread);
21
22     printf("Enter the price of milk: ");
23     scanf("%f", &price_milk);
24     printf("Enter the quantity of milk: ");
25     scanf("%d", &quantity_milk);
26
27     total_cost = (price_apple * quantity_apple) + (price_bread * quantity_bread) + (price_milk * quantity_milk);
28
29     sales_tax = total_cost * 0.05;
30     final_cost = total_cost + sales_tax;
31
32     printf("Total cost before tax: %.2f\n", total_cost);
33     printf("Sales tax (5%): %.2f\n", sales_tax);
34     printf("Final cost after tax: %.2f\n", final_cost);
35
36     return 0;
37 }
```

Untitled2.c C/C++ Windows (CR+LF) WINDOWS-1252 Line 5, Col 24, Pos 84 Insert Modified Read/Write default

Output Result



```
D:\Lang\Untitled2.exe
Enter the price of apples: 200
Enter the quantity of apples: 10
Enter the price of bread: 50
Enter the quantity of bread: 5
Enter the price of milk: 60
Enter the quantity of milk: 2
Total cost before tax: 2370.00
Sales tax (5%): 118.50
Final cost after tax: 2488.50

Process returned 0 (0x0)   execution time : 30.467 s
Press any key to continue.
```

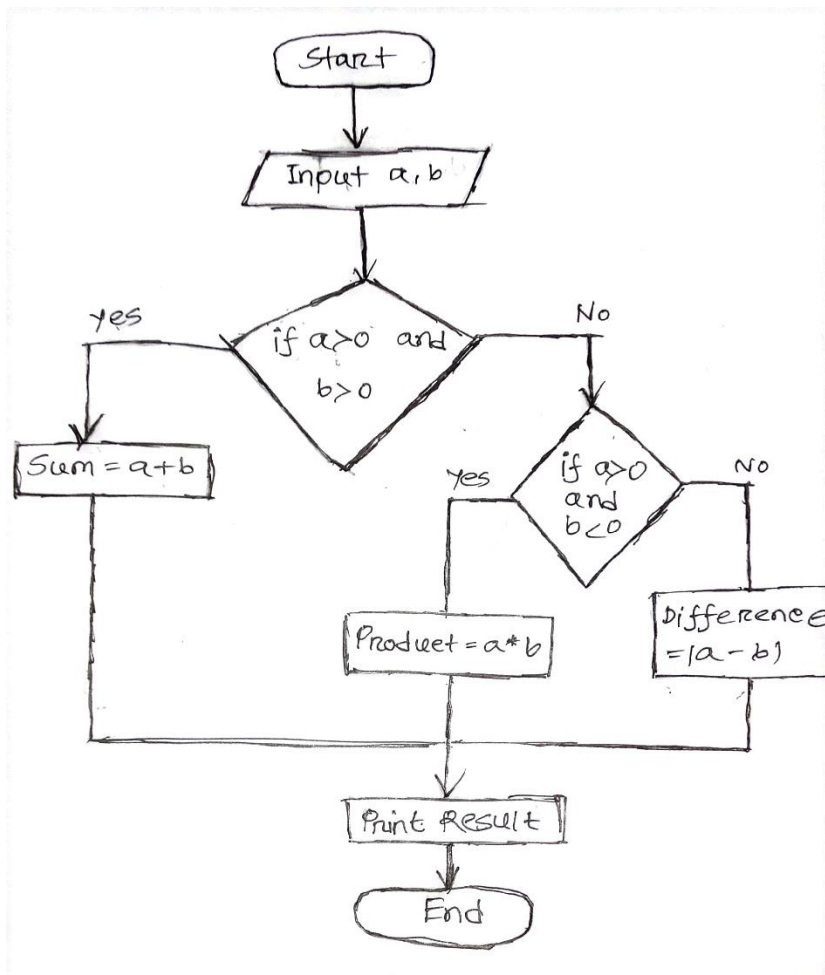
- To ensure accuracy, we have used floating-point numbers (float).
- We have inserted the formula very sincerely.

1 Question (b):

Flowchart Description:

1. Start: Begin the flowchart.
2. Input Step: Read two numbers a and b .
3. Condition 1: Check if $a > 0$ and $b > 0$. If yes, compute the sum $a + b$ and display it. If no, move to the next condition.
4. Condition 2: Check if $a > 0$ and $b < 0$. If yes, compute the product $a * b$ and display it. If no, move to the next condition.
5. Condition 3: Check if $a < 0$ and $b < 0$. If yes, compute the absolute difference $|a - b|$ and display it.

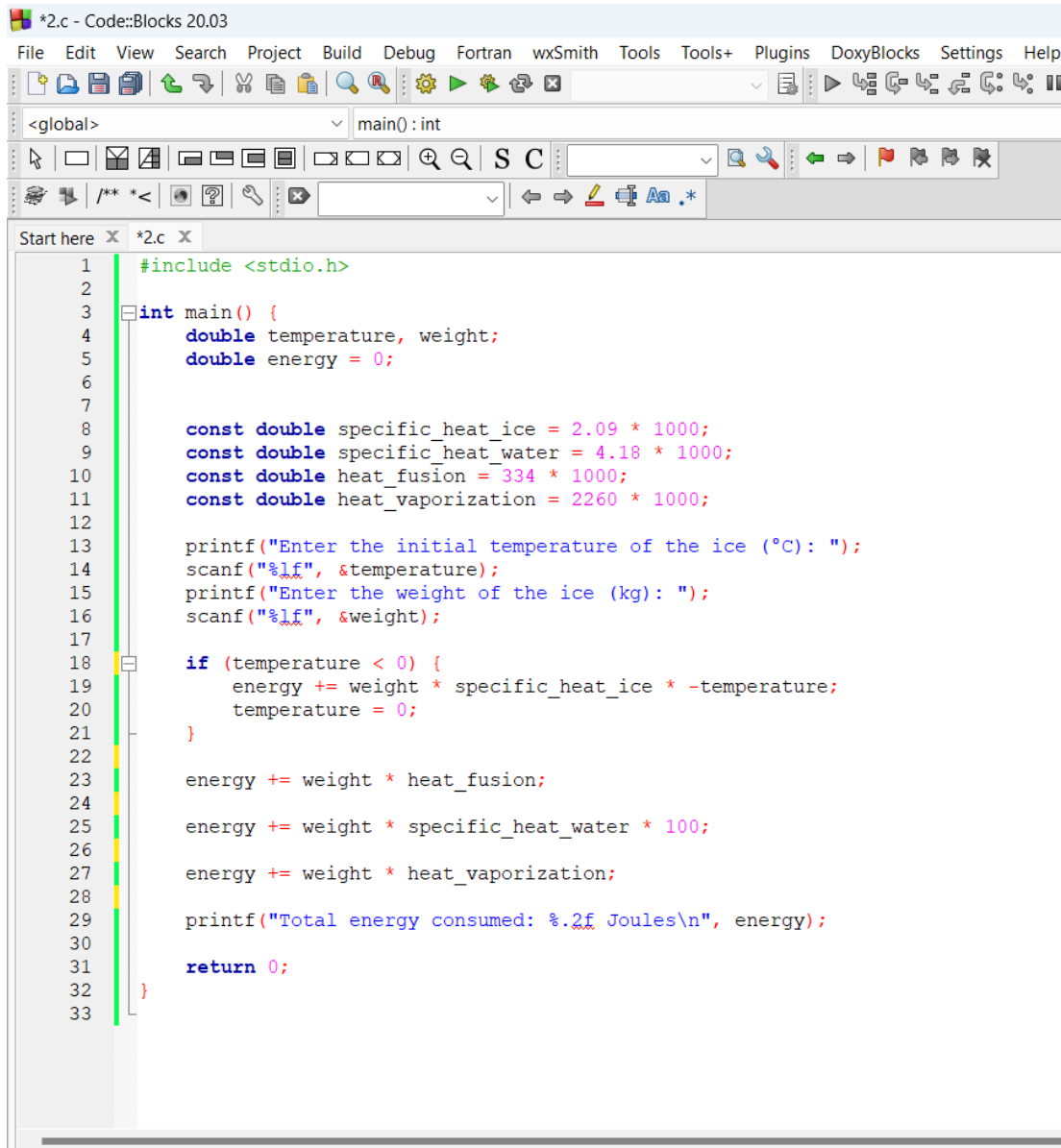
Here The drawing of the Flowchart,



2 Question:

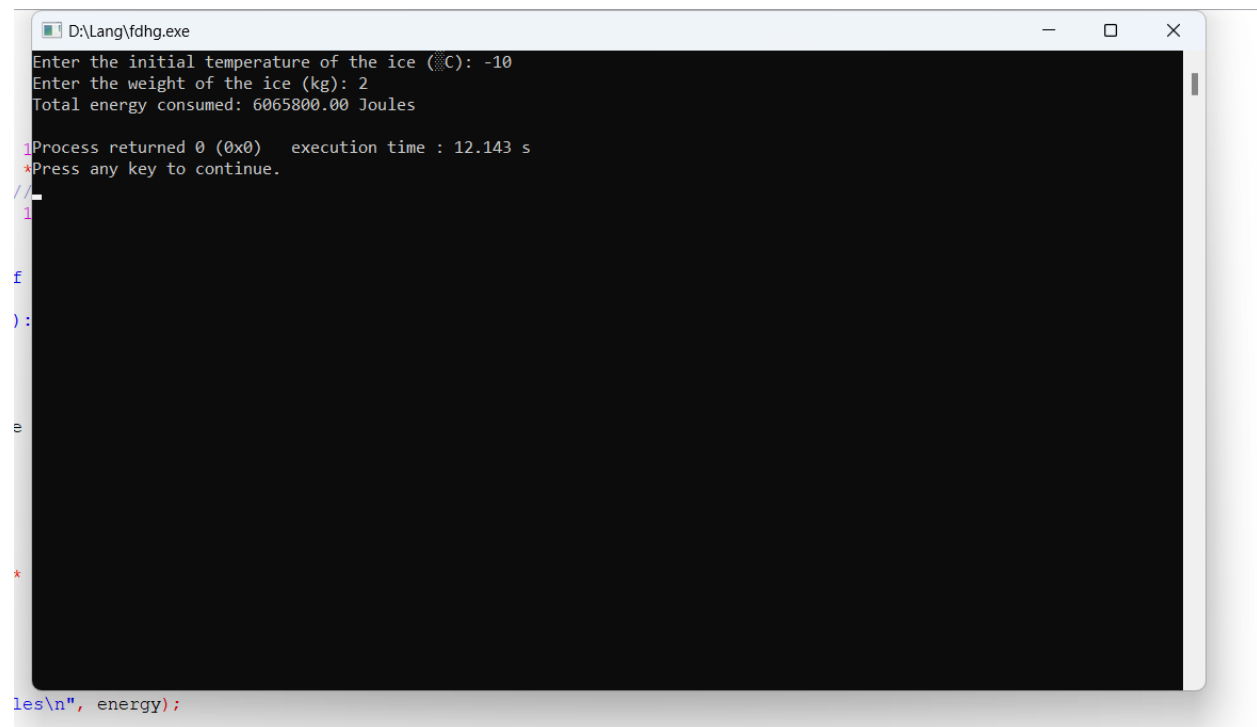
1. Heat for warming up the cool from unique temperature to 0°C
2. Energy needed to raise the temperature of water at 0°C
3. Heat energy needed to raise the water temperature from 0°C to 100°C
4. Energy needed to turn the water at 100°C into steam.

Here Code is



```
*2.c - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help
<global> main(): int
Start here x *2.c x
1 #include <stdio.h>
2
3 int main() {
4     double temperature, weight;
5     double energy = 0;
6
7
8     const double specific_heat_ice = 2.09 * 1000;
9     const double specific_heat_water = 4.18 * 1000;
10    const double heat_fusion = 334 * 1000;
11    const double heat_vaporization = 2260 * 1000;
12
13    printf("Enter the initial temperature of the ice (°C): ");
14    scanf("%lf", &temperature);
15    printf("Enter the weight of the ice (kg): ");
16    scanf("%lf", &weight);
17
18    if (temperature < 0) {
19        energy += weight * specific_heat_ice * -temperature;
20        temperature = 0;
21    }
22
23    energy += weight * heat_fusion;
24
25    energy += weight * specific_heat_water * 100;
26
27    energy += weight * heat_vaporization;
28
29    printf("Total energy consumed: %.2f Joules\n", energy);
30
31    return 0;
32 }
33
```

For the input temperature = -10°C and weight = 2 kg,
running this corrected program will output:



```
D:\Lang\fdhg.exe
Enter the initial temperature of the ice (C): -10
Enter the weight of the ice (kg): 2
Total energy consumed: 6065800.00 Joules

Process returned 0 (0x0)   execution time : 12.143 s
Press any key to continue.
//
1
f
):
e
*
les\n", energy);
```

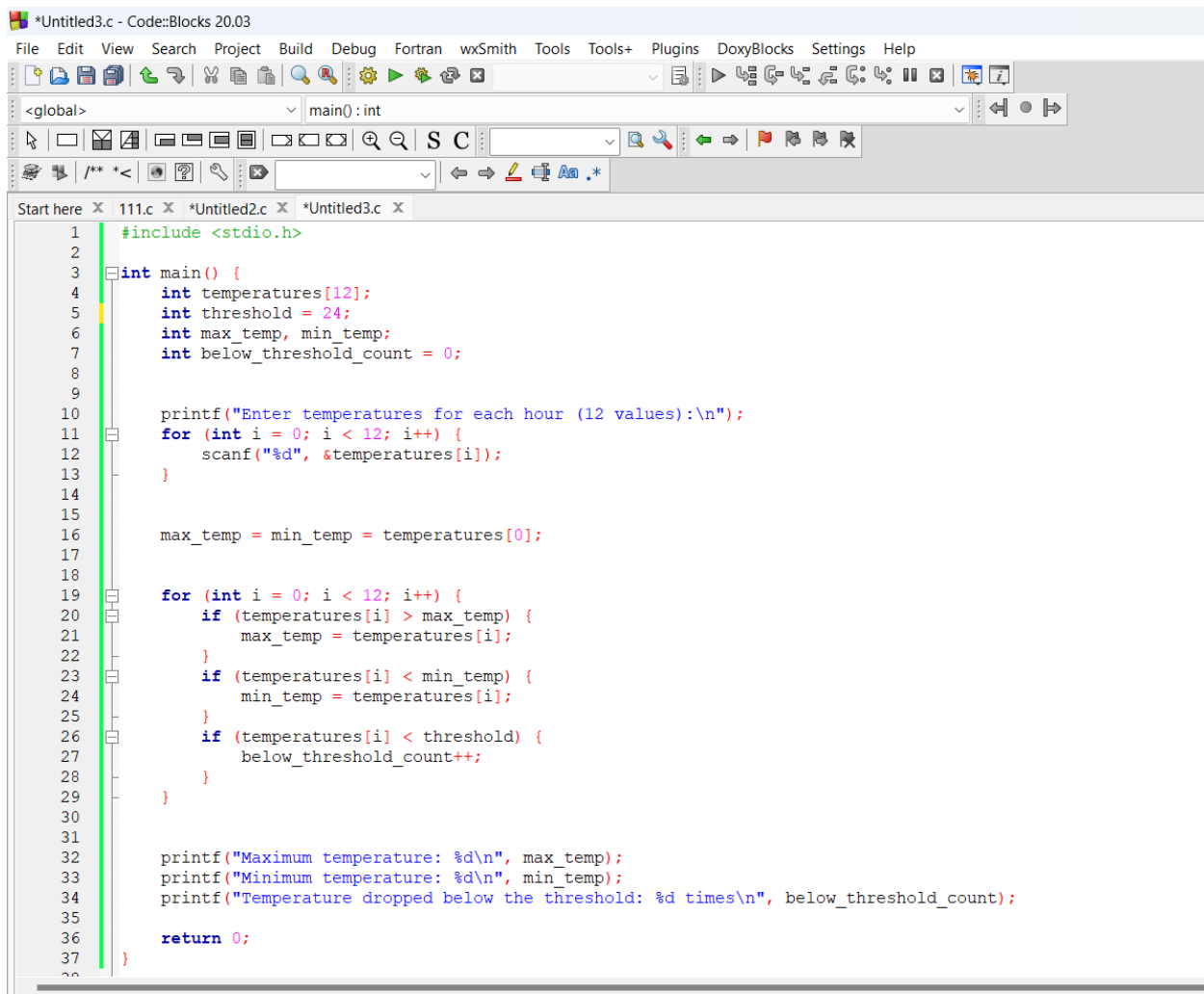
3 Question Answer:

Variables: max_temp and min_temp are initialized as very low and very high values respectively so that literally any plausible temperature input the user will provide would overwrite them.

Input: The first line of input is the threshold temperature—The program will get input one day worth 24. hours times temperatures.)

Processing: This is the max_temp and min_temp key-value pair that will set it, it does a comparison for not taking the highest and lowest temperatures of the day. It also keeps a count of the number of times the temperature goes below this threshold.

Here's Code



```
*Untitled3.c - Code::Blocks 20.03
File Edit View Search Project Build Debug Fortran wxSmith Tools Tools+ Plugins DoxyBlocks Settings Help

<global> main() : int

1  #include <stdio.h>
2
3  int main() {
4      int temperatures[12];
5      int threshold = 24;
6      int max_temp, min_temp;
7      int below_threshold_count = 0;
8
9
10     printf("Enter temperatures for each hour (12 values):\n");
11     for (int i = 0; i < 12; i++) {
12         scanf("%d", &temperatures[i]);
13     }
14
15
16     max_temp = min_temp = temperatures[0];
17
18
19     for (int i = 0; i < 12; i++) {
20         if (temperatures[i] > max_temp) {
21             max_temp = temperatures[i];
22         }
23         if (temperatures[i] < min_temp) {
24             min_temp = temperatures[i];
25         }
26         if (temperatures[i] < threshold) {
27             below_threshold_count++;
28         }
29     }
30
31
32     printf("Maximum temperature: %d\n", max_temp);
33     printf("Minimum temperature: %d\n", min_temp);
34     printf("Temperature dropped below the threshold: %d times\n", below_threshold_count);
35
36     return 0;
37 }
```

Output: Finally it writes the highest temperature, minimum temperature, and how many times the temperature has dropped below this threshold.

My University ID 242014124

Simple Input :

24

12 34 26 27 28 10 22 23 24

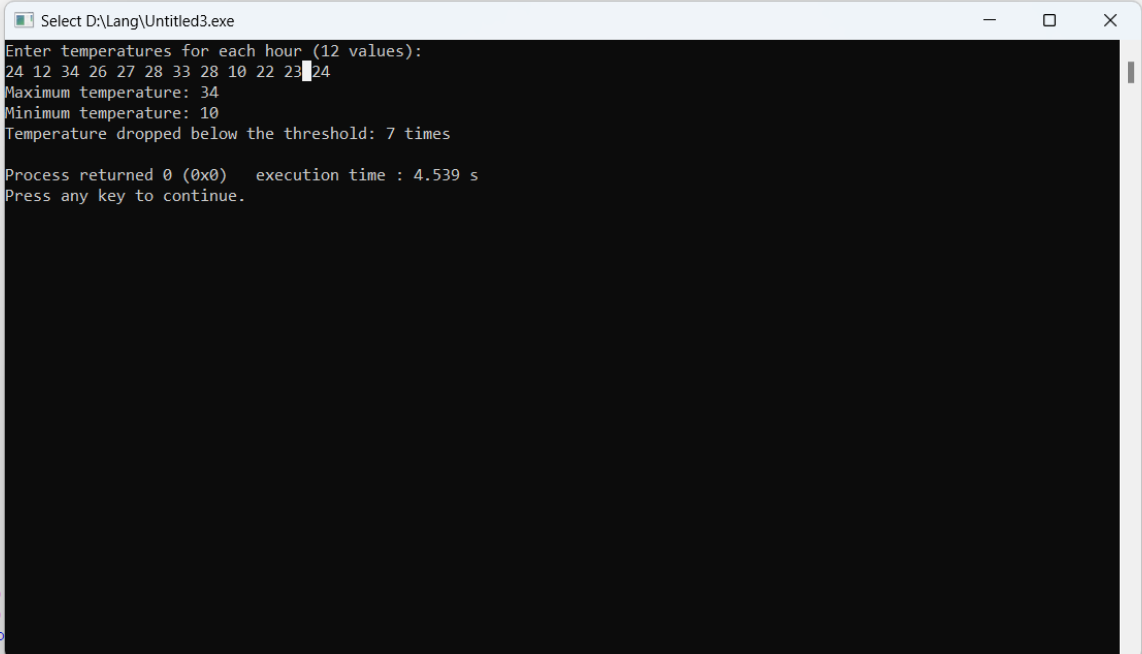
Sample output :

Maximum temperature: 34

Minimum temperature: 10

Temperature dropped below the threshold: 7 times

Output Result :



The screenshot shows a Windows command prompt window titled "Select D:\Lang\Untitled3.exe". The program prompts the user to "Enter temperatures for each hour (12 values):". The user has entered the sequence: 24 12 34 26 27 28 33 28 10 22 23 24. The program outputs the following results: "Maximum temperature: 34", "Minimum temperature: 10", and "Temperature dropped below the threshold: 7 times". Below the output, it shows "Process returned 0 (0x0) execution time : 4.539 s" and "Press any key to continue.". To the left of the command prompt, parts of the C++ source code are visible, including a loop for input and variables for tracking maximum temperature, minimum temperature, and the number of times the temperature dropped below a threshold.

```
hour (12)
};
{
{
{
max_temp)
min_temp)
e thresho
```


4 Question Answer :

Here, we need to write a C program that can simulate an online shopping cart system according to the requirements of the problem statement.

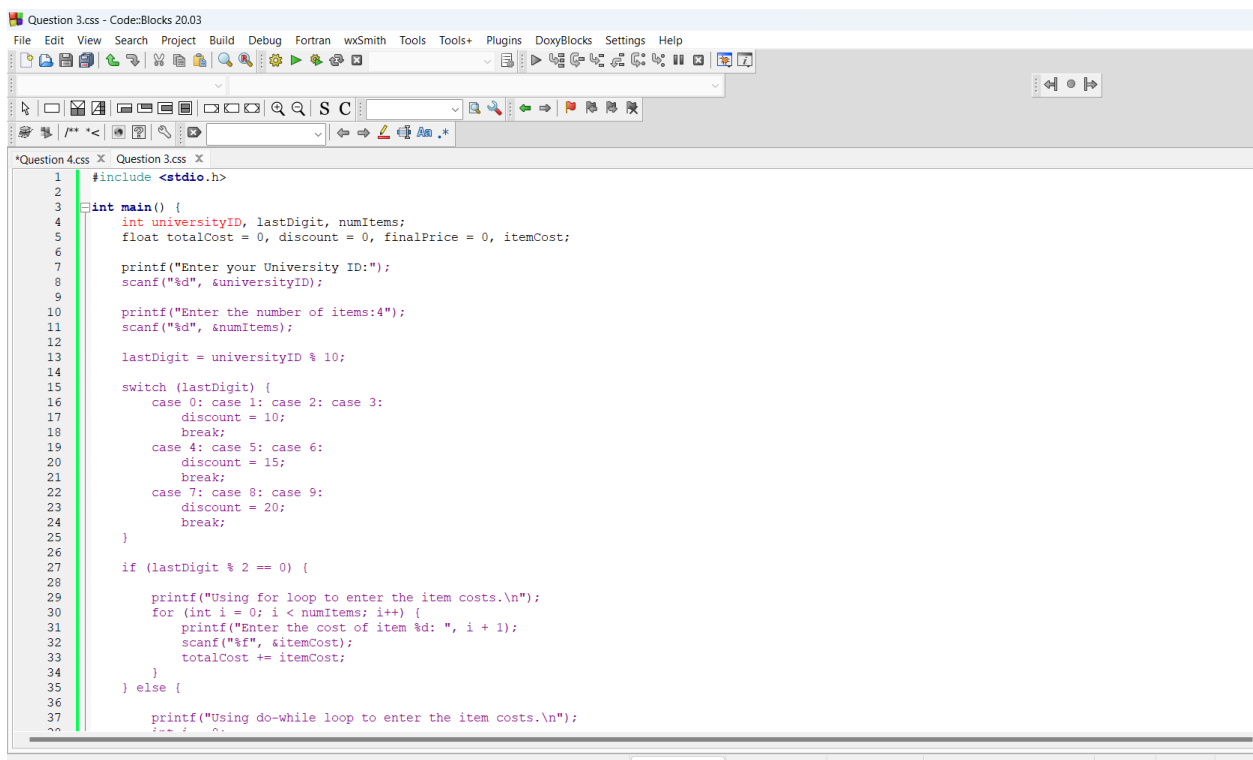
Input University ID : Read University ID.

Discount Check: By Switch-Case the function decides which discount should be set!

Loop through the Costs: Use either a for loop or while/do-while loop depending on whether the last digit is even to mimic adding costs of items to an accumulated amount.

Final Price Calculation: Deduct the discount that you have found from the total cost.

Here's Code

The image shows a screenshot of a code editor window titled "Question 3.css - Code::Blocks 20.03". The editor contains a C program for an online shopping cart simulation. The code includes a main function that prompts the user for a University ID and the number of items. It then uses a switch-case statement to determine a discount based on the last digit of the University ID. Finally, it uses a for loop or a do-while loop to calculate the total cost of the items, depending on whether the last digit is even or odd. The code is as follows:

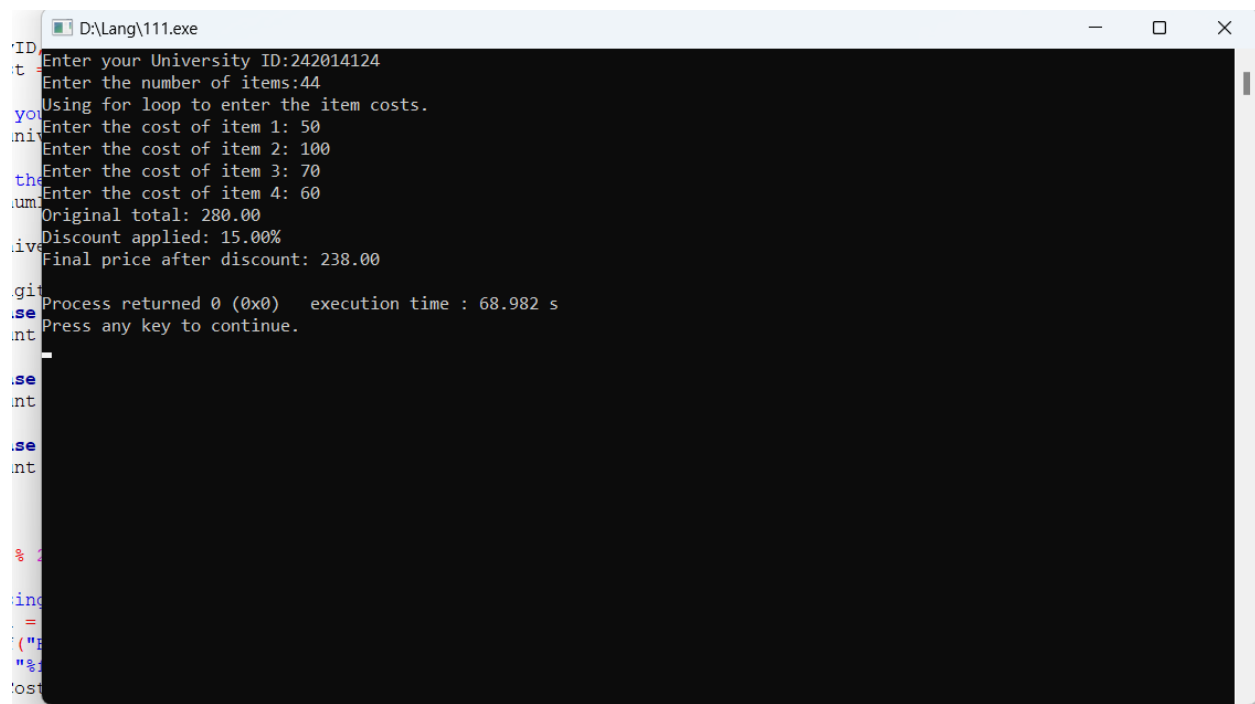
```
1 #include <stdio.h>
2
3 int main() {
4     int universityID, lastDigit, numItems;
5     float totalCost = 0, discount = 0, finalPrice = 0, itemCost;
6
7     printf("Enter your University ID:");
8     scanf("%d", &universityID);
9
10    printf("Enter the number of items:4");
11    scanf("%d", &numItems);
12
13    lastDigit = universityID % 10;
14
15    switch (lastDigit) {
16        case 0: case 1: case 2: case 3:
17            discount = 10;
18            break;
19        case 4: case 5: case 6:
20            discount = 15;
21            break;
22        case 7: case 8: case 9:
23            discount = 20;
24            break;
25    }
26
27    if (lastDigit % 2 == 0) {
28        printf("Using for loop to enter the item costs.\n");
29        for (int i = 0; i < numItems; i++) {
30            printf("Enter the cost of item %d: ", i + 1);
31            scanf("%f", &itemCost);
32            totalCost += itemCost;
33        }
34    } else {
35        printf("Using do-while loop to enter the item costs.\n");
36        do {
37            printf("Enter the cost of item %d: ", i + 1);
38            scanf("%f", &itemCost);
39            totalCost += itemCost;
40            i++;
41        } while (i < numItems);
42    }
43
44    finalPrice = totalCost - discount;
45    printf("Final Price: %f", finalPrice);
46}
```

```
31         printf("Enter the cost of item %d: ", i + 1);
32         scanf("%f", &itemCost);
33         totalCost += itemCost;
34     }
35 } else {
36     printf("Using do-while loop to enter the item costs.\n");
37     int i = 0;
38     do {
39         printf("Enter the cost of item %d: ", i + 1);
40         scanf("%f", &itemCost);
41         totalCost += itemCost;
42         i++;
43     } while (i < numItems);
44 }
45
46 finalPrice = totalCost - (totalCost * discount / 100);
47
48 printf("Original total: %.2f\n", totalCost);
49 printf("Discount applied: %.2f%%\n", discount);
50 printf("Final price after discount: %.2f\n", finalPrice);
51
52 return 0;
53 }
54
55
56
```

D:\Lang\Question 3.css CSS Windows (CR+LF) WINDOWS-1252 Line 47, Col 4, Pos 1220 Insert

Output: Display the original total, discount rate, discount amount, and actual price paid after the discount.

Output Result



```
D:\Lang\111.exe
Enter your University ID:242014124
Enter the number of items:44
Using for loop to enter the item costs.
Enter the cost of item 1: 50
Enter the cost of item 2: 100
Enter the cost of item 3: 70
Enter the cost of item 4: 60
Original total: 280.00
Discount applied: 15.00%
Final price after discount: 238.00
Process returned 0 (0x0)   execution time : 68.982 s
Press any key to continue.
```

