LBYEC2A

Computer Fundamentals and Programming 1

Laboratory Activity 5

C - Conditional Statements

By

John Carlo Theo S. Dela Cruz | LBYEC2A - EQ1

OBJECTIVES

- Learn the different looping statements: while loop and do-while loop
- Develop algorithms and flowcharts for use in programming applications.
- Design, compile, test, run, and implement the C language program
- Creating a flowchart based on the programs needed
- Maximize and explore the features of Visual Studio Code

MATERIALS

1. Visual Studio Code
    a. C Compiler (mingw64)
    b. Draw.io (Flowchart Extension)
    c. W3schools
2. Module Guide: DLSU Canva
3. Snipping Tool

PROCEDURE

1. **Lab Activity 6.1 Countdown:** Create a program that creates a countdown based on the user's input.
    a. Open Visual Studio Code
    b. Create a new file and name it 'countdown.c' or whatever filename the user prefers as long as it ends with '.c' to ensure you have a c program file.
    c. Since the program won't require the function of math we will Input the default format of the c program. **#include <stdio.h>**
    d. **DECLARE FUNCTIONS and VARIABLE -** *int main(void) is the main function that will be called without any parameter. Void functions are used just like value-returning functions except the void does not return a value when the function is executed.* Inside the main function, we declare the variable level that will be an indication as the user's input and the program itself will have a set condition to follow.

```
int main(void){
//variable declaration
int x, i;
```

e. **INPUT AND PRINT -** After declaring the variable we now need to print out the title and the instructions that will require the user to input a specific number and print out its descending order.

```
//intro
printf("==COUNTDOWN==");

// input
printf("\n\nInput a Number: ");
scanf("\n%d", &x);
printf("\n\nLift off in...\n");
```

f. **LOOPING STATEMENT -** The goal of this program is to subtract by 1 and print out the value until it reaches 0. By using a for loop and setting a condition that the loop starts from the value inputted and the last value will end with 0. After it reaches 0, it prints out a “Blast-Off” statement that indicates the end of the program.

```
// for-loop
for (i = x; i >= 0 ; i--){ // (starting value, condition, decrement)
    printf("\n%d", i);
}

    printf("\n\nBlast-Off !!!");
    return 0;

}
```

2. **Lab Activity 6.2 Series:** Create a program that Converts a specific value to another. In this program, we require the user to choose an option of what kind of conversion would they prefer.
    a. Open Visual Studio Code
    b. Create a new file and name it ‘series.c’ or whatever filename the user prefers as long as it ends with ‘.c’ to ensure you have a c program file.
    c. Since the program won’t require the function of math we will Input the default format of the c program. **#include <stdio.h>**
    d. **DECLARE FUNCTIONS and VARIABLE -** Inside the main function, we

declare the function of the input value (x), summation (total), and the starting value (i).

```
int x; // input variable
int total = 0; // total or summation
int i = 1; // starting value / placeholder
```

e. **INPUT AND PRINT -** After declaring the variable we now need to print out the title and instructions that will require the user to input a specific value.

```
//print and input
printf("==SUMMATION==");
printf("\n\nEnter a Value: ");
scanf(" %d", &x);
printf("\n");
```

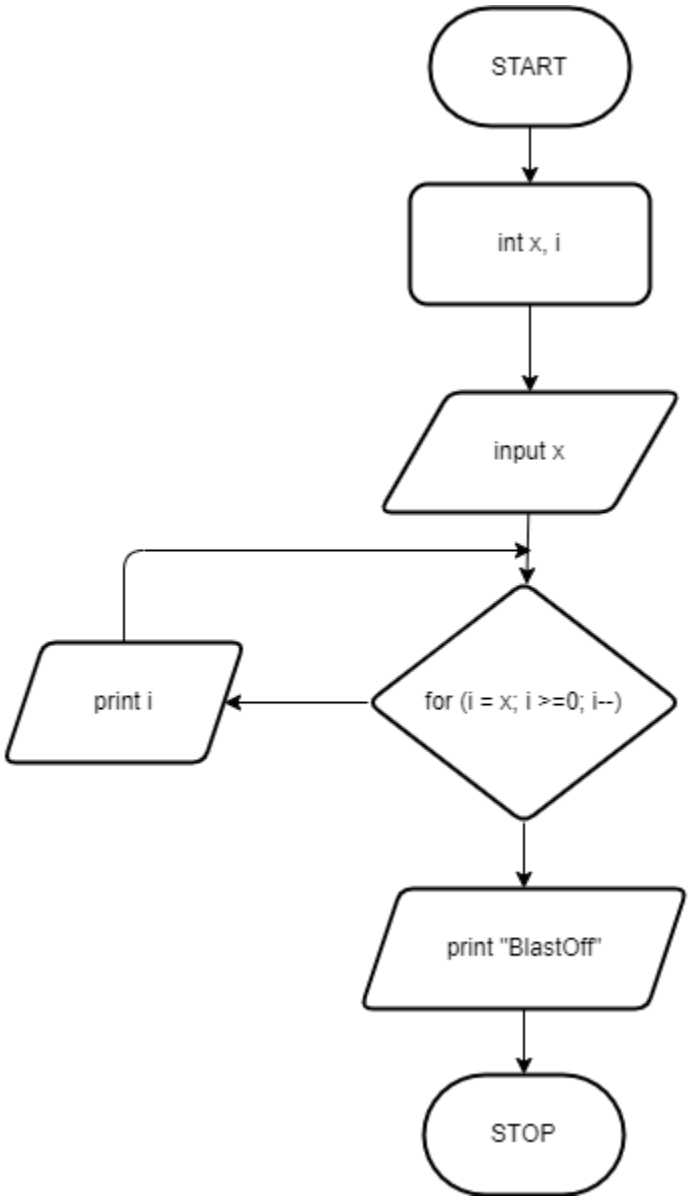
f. WHILE LOOP - Using the While Loop and setting a condition that prints out all the values below the input. After adding every value of i, it prints out the summation based on the input of the user.

```
// it prints out value i (or 1) until the value of x
while(i <= x)
{
    total += i; //Adds every value of i
    printf("%d\n", i );
    i++; // increment - to ascending order
}
    //print the total / summation
    printf("\nSUMMATION IS: %d", total);
    return 0;
}
```

ALGORITHM FLOWCHARTS (*Include the code flowcharts here)*

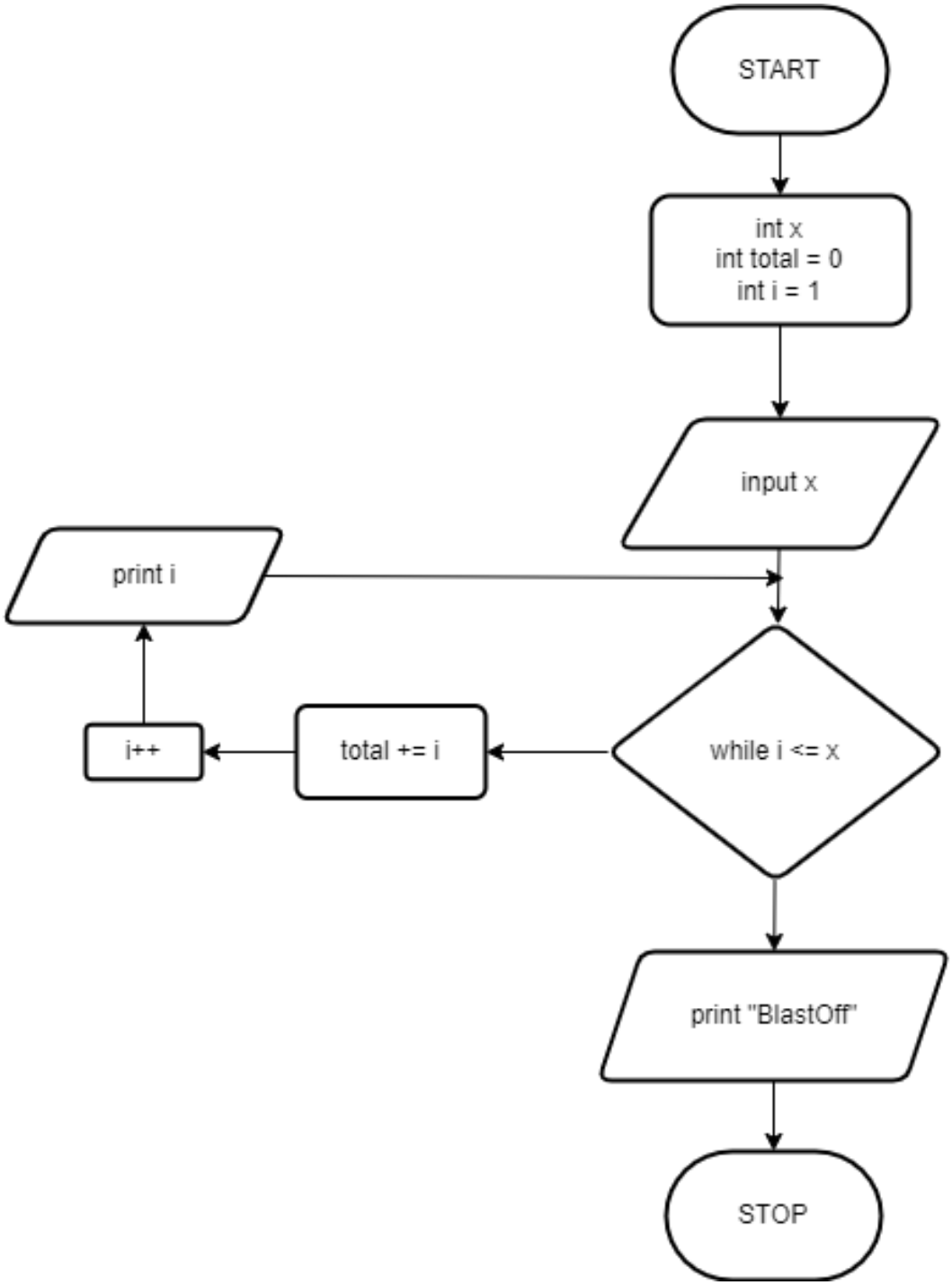
1. **Lab Activity 6.1 Countdown:**

Flowcharts:



2. **Lab Activity 6.2 Series:**

Flowcharts:



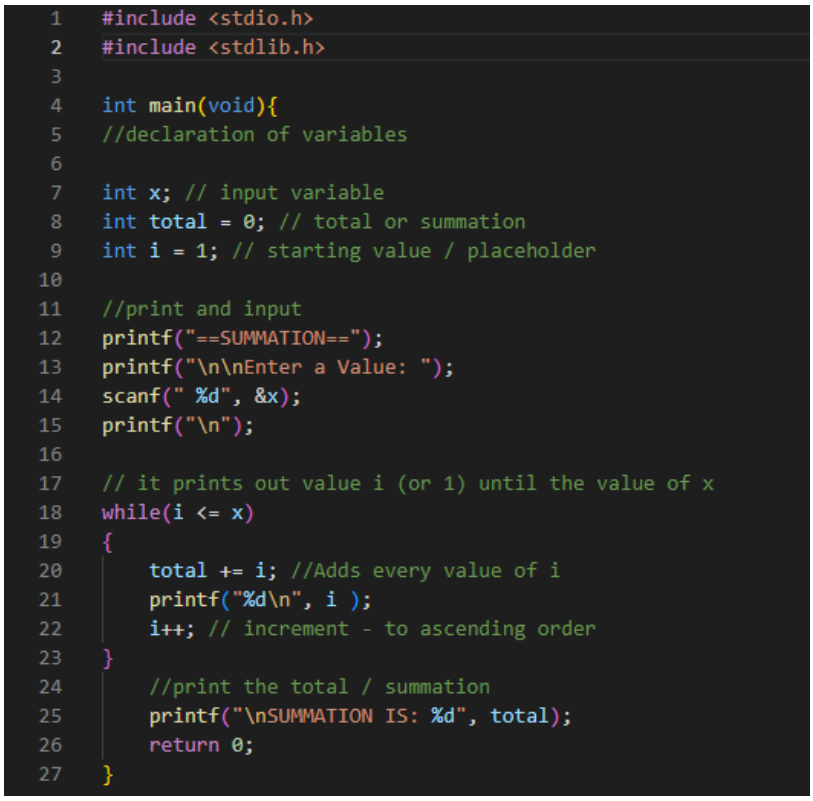
RESULTS AND DISCUSSION (*Include the screenshots and discussions per problem solution*)

1. **Lab Activity 6.1 Countdown:**

```c
#include <stdio.h>
#include <stdlib.h>

int main(void){
//variable declaration
int x, i;

//intro
printf("==COUNTDOWN==");

// input
printf("\n\nInput a Number: ");
scanf("\n%d", &x);
printf("\n\nLift off in...\n");

// for-loop
for (i = x; i >= 0 ; i--){ // (starting value, condition, decrement)
    printf("\n%d", i);
}

    printf("\n\nBlast-Off !!!");
    return 0;

}
```

In this program, we are required to use one of the functions of the looping statements which is a for-loop; a for-loop is used for a definite loop and the loop stops at a certain/set condition. In this case, we utilized how we can use range in looping. Inside the condition statement is the starting value (which is the user's input), the condition of the last value (zero), and lastly the decrementing value to create and print the numbers in descending order. Once it reaches 0 then the loop ends and prints out "Blast Off" which indicates the end of the program.

2. **Lab Activity 6.2 Series:**



```c
#include <stdio.h>
#include <stdlib.h>

int main(void){
//declaration of variables

int x; // input variable
int total = 0; // total or summation
int i = 1; // starting value / placeholder

//print and input
printf("==SUMMATION==");
printf("\n\nEnter a Value: ");
scanf(" %d", &x);
printf("\n");

// it prints out value i (or 1) until the value of x
while(i <= x)
{
    total += i; //Adds every value of i
    printf("%d\n", i );
    i++; // increment - to ascending order
}
    //print the total / summation
    printf("\nSUMMATION IS: %d", total);
    return 0;
}
```

In this program, we are required to use a while-loop statement; a while loop statement loops a certain block of codes as long as the condition is true or satisfied. In this Activity, the required output is to show the ascending order of the user's input as well as their summation. The variable x will be the input, the total is set to 0 because we will be adding values of i, and variable i will be the starting value of the ascending order. We used an increment in the while loop because we need the value to add by 1 every loop, and total +=i or total = total + i, will add every i value. The condition of the loop is less than or equal than the user's input which is x. After it is completely satisfied, the program will print out the summation of the user's input and indicates the end of the program.

SUMMARY and WHAT I LEARNED (*Sample: around 5 or more sentences; include*)

In Module 6 / Lab Activity 6, we were introduced and focused on Looping Statements (For and While Loop). We utilized each statement in our Lab Activities. The first activity which is the Countdown, For loop was used because we need to repeat a certain block and decrement the value to display a descending order of the user's input. 2nd activity which is the Series, While loop was used to print out the user's input in ascending order and to be able to get the summation of the user's input. Looping Statements are very important, due to the fact that loops can allow the programmers to shorten their codes and allows them to write a block of code once a repeat the same code at a certain condition. One challenge encountered while performing this activity was not knowing what to do. Loop statements may be confusing at first but practice makes perfect. Understanding and executing them at the same time can increase your knowledge about a certain topic. There are multiple sources on the internet to get a piece of certain information for us to understand Loop Statements.

REFERENCES

1. Cabatuan, M. (n.d.) LBYEC2A C - Looping Statement. Retrieved from. https://dlsu.instructure.com/courses/91770/discussion_topics/780628?module_item_id=2195644

APPENDIX (*For coding problems, copy all codes here*)(C codes)

1. **EARTHQUAKE.C**

```
#include <stdio.h>
#include <stdlib.h>

int main(void){
int x, i;

printf("==COUNTDOWN==");
printf("\n\nInput a Number: ");
scanf("\n%d", &x);
printf("\n\nLift off in...\n");

// for-loop
for (i = x; i >= 0 ; i--){ // (starting value, condition, decrement)
   printf("\n%d", i);
}

   printf("\n\nBlast-Off !!!");
   return 0;

}
```

2. **CONVERSION.C**

```
#include <stdio.h>
#include <stdlib.h>

int main(void){

int x; // input variable
int total = 0; // total or summation
int i = 1; // starting value / placeholder

printf("==SUMMATION==");
printf("\n\nEnter a Value: ");
scanf(" %d", &x);
printf("\n");

while(i <= x)
{
   total += i; //Adds every value of i
   printf("%d\n", i );
   i++; // increment - to ascending order
}

   printf("\nSUMMATION IS: %d", total);
   return 0;
```

```
}
```