
# EE2028A

# C Programming

# Laboratory Exercise (LAB-II)

**Name: Anurag Chatterjee**

**Matriculation Number: A0200533Y**

## Submission instructions:

1. **Test your code on your computer first** before submitting.
2. You must name your functions exactly as the question states.
3. **DEADLINE**: **Tuesday11 Feb 2020 / Thursday 13 Feb 2020**
4. **LumiNUS "Lab 2 Assignment Submission Folder"**
5. **Grading: Your assignment will be graded out of 50 marks and the final weight of this assignment is 15%.**
6. You are expected to follow the guidelines given below:
    a. Use meaningful variable names while programming. It's a good practice to develop good programming skills and enables readability.
    b. Explain the code with proper comments; Comments must be meaningful and descriptive;
    c. Please adhere to the report deadlines and any late submissions are not accepted.
7. Please prepare the report in **PDF**format **only**.
8. Submit the following:

| Submit the compressed file | Contains |
|---|---|
| MATRICULATION_NUMBER_ASSIGNMENT 2_NAME (First Name).zip | REPORT_MATRICULATION_NUMBER_ASSIGNMENT2_NAME (First Name).**pdf** |
| | Your working C code, **ONLY .c file** |

**What you need to add into this report for submission? - YOUR OUTPUT:**

a. Program Code (attach in the ZIP file – see the guidelines for submission) addition to the .c file that you need to submit
   - Code should be well written with meaningful variables and comments.

b. In THIS report, **screenshot your results and paste.** Make sure it is visible, readable and clear.
   - **For Question1:**Your code and the screenshot of 3 different sample runs.
   - **For Question 2:**Your code and the screenshot of 3 different sample runsfor given n and m values.

**DO NOT FORGET TO SEND YOUR .C FILE WITH ALL YOUR WORKING CODES BESIDES ATTACHING THE CODES INTO THE REPORT.**

---

## NOTE: Start your answers from here. Use as much space as needed.

PROBLEM 1:

### CODE

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

int main() {
  int balance=1000;
  char input;
  int attempts=0;
  int deposit;
  int withdrawal;
  printf("Welcome to our Dream Bank!\n");
  printf("Your current balance: %d\n",balance);

  printf("==================\n");

  printf("What would you like to do?\n");
  printf("D - Deposit\n");
  printf("W - Withdrawal\n");
  printf("B - Balance Inquiry\n");
  printf("Q - Quit\n");

  printf("Please enter:");
  scanf("%c",&input);
  input = toupper(input);
    switch (input)
  {
    case 'D':
    printf("How much would you like to deposit?\n");
    scanf("%d",&deposit);
    if (deposit % 10 != 0) {
    printf("Unacceptable transcation, deposit only in increments of 10!\n");
    }
```

```
    printf("Unacceptable transcation, deposit only in increments of 10!\n");
    }
    else {
    printf("Transaction approved!");
    balance += deposit;
    printf("Your current balance: %d\n",balance);
    }
    break;

    case 'W':
    printf("How much would you like to withdraw?\n");
    scanf("%d",&withdrawal);
    if (balance > withdrawal) {
      if (withdrawal % 10 != 0) {
        printf("Unacceptable transcation, withdrawal only in increments of 10!\n");
        break;
        }
      else if (withdrawal % 10 == 0){
        printf("Transaction approved!");
        balance -= withdrawal;
        printf("Your current balance: %d\n",balance);
        break;
        }
      else {
        printf("Insufficient balance!");
        printf("Your current balance:%d\n",balance);
        break;
        }
      }
      break;
```

```
        }
      }
      break;

    case 'B':
      printf("Your current balance: %d\n",balance);
      break;

    case 'Q':
      printf("Your current balance: %d\n",balance);
      printf("Thanks for working with us. Have a nice day!");
      return 0;

    default:
      if (attempts >= 3) {
        printf("The number of trials is exceeded, your account is blocked!");
        return 0;
        }
      else {
        attempts ++;
        printf("Invalid option, please enter a valid option!");
        }
      }
    }
```

## OUTPUT 1:

```
clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_700/final)
> clang-7 -pthread -lm -o main main.c
> ./main
Welcome to our Dream Bank!
Your current balance: 1000
==================
What would you like to do?
D - Deposit
W - Withdrawal
B - Balance Inquiry
Q - Quit
Please enter:D
How much would you like to deposit?
35
Unacceptable transcation, deposit only in increments of 10!
```

## OUTPUT 2:

```
clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_700/final)
> clang-7 -pthread -lm -o main main.c
> ./main
Welcome to our Dream Bank!
Your current balance: 1000
==================
What would you like to do?
D - Deposit
W - Withdrawal
B - Balance Inquiry
Q - Quit
Please enter:D
How much would you like to deposit?
40
Transaction approved!Your current balance: 1040
```

## OUTPUT 3:

```
clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_700/final)
> clang-7 -pthread -lm -o main main.c
> ./main
Welcome to our Dream Bank!
Your current balance: 1000
==================
What would you like to do?
D - Deposit
W - Withdrawal
B - Balance Inquiry
Q - Quit
Please enter:W
How much would you like to withdraw?
800
Transaction approved!Your current balance: 200
```

## OUTPUT 4:

```
clang version 7.0.0-3~ubuntu0.18.04.1 (tags/RELEASE_700/final)
: clang-7 -pthread -lm -o main main.c
: ./main
Welcome to our Dream Bank!
Your current balance: 1000
==================
What would you like to do?
D - Deposit
W - Withdrawal
B - Balance Inquiry
Q - Quit
Please enter:W
How much would you like to withdraw?
155
Unacceptable transcation, withdrawal only in increments of 10!
```

PROBLEM 2:

## CODE

```c
#include <stdio.h>

int Fibonacci(int n)
{
  if (n == 0) {
    return 0;
  }
  else if (n == 1) {
    return 1;
  }
  else {
    return (Fibonacci(n-1) + Fibonacci(n-2));
  }
}

int main() {
  int start,end,i =1,product=0,sum=0,div2=0,div3=0,div5=0;
  int arrayseq[end-start+1];
  printf("Please enter n:");
  scanf("%d",&start);
  printf("Please enter m:");
  scanf("%d",&end);
  printf("=====\n");

  while (start <= end)
  {
    arrayseq[i-1]= Fibonacci(start-1);
    sum += arrayseq[i-1];
    if (i== 1){
      product = arrayseq[i-1] * Fibonacci(start -1);
    }
    else
      product = arrayseq[i-1] * arrayseq[i-2];
    printf("(%d,%d,%d)",i,arrayseq[i-1],product);
    if (product%2==0) {
      printf("(2,");
      div2++;
    }
    else {
```

```
            printf("(_,");
        }
        if (product%3==0) {
            printf("3,");
            div3++;
        }
        else {
            printf("_,");
        }
        if (product%5==0) {
            printf("5)");
            div5++;
        }
        else {
            printf("_)");
        }
        printf("\n");
        i ++;
        start ++;
    }
printf("Number of product terms divisible by 2: %d\n",div2);
printf("Number of product terms divisible by 3: %d\n",div3);
printf("Number of product terms divisible by 5: %d\n",div5);
printf("Sum of all Fibonnaci numbers: %d\n",sum);
return 0;
}
```

**OUTPUT 1:**

```
Please enter n:7
Please enter m:19
=====
(1,8,64)(2,_,_)
(2,13,104)(2,_,_)
(3,21,273)(_,3,_)
(4,34,714)(2,3,_)
(5,55,1870)(2,_,5)
(6,89,4895)(_,_,5)
(7,144,12816)(2,3,_)
Number of product terms divisible by 2: 5
Number of product terms divisible by 3: 3
Number of product terms divisible by 5: 2
Sum of all Fibonnaci numbers: 364
```

**OUTPUT 2:**

```
Please enter n:10
Please enter m:30
=====
(1,34,1156)(2,_,_)
(2,55,1870)(2,_,5)
(3,89,4895)(_,_,5)
(4,144,12816)(2,3,_)
(5,233,33552)(2,3,_)
(6,377,87841)(_,_,_)
(7,610,229970)(2,_,5)
Number of product terms divisible by 2: 5
Number of product terms divisible by 3: 2
Number of product terms divisible by 5: 3
Sum of all Fibonnaci numbers: 1542
```

**OUTPUT 3:**

```
Please enter n:3
Please enter m:25
=====
(1,1,1)(_,_,_)
(2,2,2)(2,_,_)
(3,3,6)(2,3,_)
(4,5,15)(_,3,5)
(5,8,40)(2,_,5)
(6,13,104)(2,_,_)
(7,21,273)(_,3,_)
(8,10946,240812)(2,_,_)
```