

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

**CS-118 Programming Fundamentals Spring-2021
ASSIGNMENT-5**

Section (A, B, C, D, E and F)

Instructions:

1. Assignments are to be done individually. You must complete this assignment by yourself. You cannot work with anyone else in the class or with someone outside of the class. The code you write must be your own and you must understand each part of your code. You are encouraged to get help from the instructional staff through google classroom.
2. Use appropriate data types, operations, and conditional structures, and loops for each problem. You cannot use arrays or advanced topics (for example Chapter 7 of the textbook and onward) for this assignment.
3. Do not use any String or math libraries (such as cmath etc) and also do not use built-in function (such as pow etc). **Caution:** zero marks will be awarded.
4. Do not edit **Function Prototypes**.
5. Your code must be **generic**.
6. Marks distribution and test examples are provided with each question. Your code will be evaluated with **similar test examples**. If the required output is generated, you will be awarded full marks. Failing to generate the correct output will result in zero marks. Total Marks: 70.
7. No late assignments will be accepted.
8. The output should be properly displayed and well presented. One mark will be deducted in each question if appropriate comments and indentation not done in source code (Total of seven marks can be deducted).
9. **Plagiarism:** Plagiarism of any kind (copying from others, copying from the internet, etc) is not allowed. If found plagiarized, you will be awarded zero marks in the assignment. Repeating such an act can lead to strict disciplinary actions and failure in the course.
10. **Submission Guidelines:** Dear students, we will be using auto-grading tools, so failure to submit according to the below format would result in zero marks in the relevant evaluation instrument.
 - a. For each question in your assignment if necessary, make a separate .cpp file e.g. for question 1, make ROLL-NUM_SECTION_q1.cpp, and so on. Each file that you submit must contain your name, student-id, and assignment # on the top of the file in the comments.
 - b. Combine all your work in one folder. The folder must contain only .cpp files (no binaries, no exe files etc.). If we unable to download your submission due to any reason you will be awarded zero mark.
 - c. Run and test your program on a lab machine before submission. If there is a syntax error, zero marks will be awarded in that specific question.
 - d. Rename the folder as ROLL-NUM_SECTION (e.g. 20i-0001_A) and compress the folder as a zip file. (e.g. 20i-0001_A.zip). Only zip file will be acceptable.
 - e. Submit the .zip file on Google Classroom within the deadline.
 - f. Submission other than Google classroom (e.g. email etc.) will not be accepted.
 - g. The student is solely responsible to check the final zip files for issues like corrupt files, viruses in the file, mistakenly exe sent. If we cannot download the file from Google classroom due to any reason it will lead to zero marks in the assignment.

Note: Follow the given instruction to the letter, failing to do so will result in a zero.

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Total Marks: 06

1. Write a C++ function that evaluates polynomials $a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x^1 + a_0 x^0$. It should take following inputs:
 - Value of x.
 - Highest power n.
 - Coefficients in descending order $a_n, a_{n-1}, \dots, a_2, a_1, a_0$ (coefficients will be $n+1$)

For instance, if $x=4$, $n=3$ & coefficients are [2,3,1,2], then output should be 182 which is obtained by evaluating the polynomial $2 * 4^3 + 3 * 4^2 + 1 * 4^1 + 2 * 4^0$

Function Prototype: double evaluatePolynomial();

Example01:*****Marks: 02

x = 4

p = 3

Enter 4 coefficients in descending order of power

2

3

1

2

output: 182

Example02:*****Marks: 02

x = -2.5

p = 7

Enter 8 coefficients in descending order of power

-2

0

0

-4.6

0

1

6

0

output: 1032.265625

Example03:*****Marks: 01

x = -2.5

p = 0

Enter 1 coefficients in descending order of power

7

output: 7

Example04:*****Marks: 01

x = -2.5

p = -4

output: Error (power must be ≥ 0)

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Note:

- Do not use built-in power function. And power must be ≥ 0 and integer.
- Do not use array or string etc.
- Read inputs inside evaluatePolynomial function.
- Display output in main function.

Total Marks: 06

2. **Happy Numbers:** A number (>0) is called a happy number, if you start with the given number and arrive at 1 by repeating the following process (as illustrated in the below example): (a) compute the sum of the squares of given number digits (b) if the resultant value is 1, then the number is happy number, else execute point (a) for the newly produced number.

Note that if a number is not a happy number, there will be an endless loop to this execution.

Goal: In this question, you are required to write Function that checks whether the number entered by the user is happy number or not for 100 cycles/iterations only.

Example: Assume a number 19

Number	Computation	Result	cycle/iterations
19	$1^2 + 9^2$	82	1
82	$8^2 + 2^2$	68	2
68	$6^2 + 8^2$	100	3
100	$1^2 + 0^2 + 0^2$	1	4

Stop computing further, as the Result obtained is 1. Hence, the number '19' , is a happy number. We achieved this in 4th cycles only.

Function Prototype: bool isHappyNumber(int num);

Example

01.*****marks: 02

Input:

Number: 19

Output: Happy Number

Example

02.*****marks: 02

Input:

Number: 7

Output: Happy Number

Example

03.*****marks: 02

Input:

Number: 18

Output: Sad Number

Note: Take input and display output in main function.

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Total Marks 06

3. Given a number N (>0). Write a c++ function to find a digit whose frequency is maximum.

Function Prototype: void maxFrequencyOfNumber(int num,int & digit, int &maxFreq);

Examples:

Marks: 02

Input: N = 1122322 Output: digit: 2 Max Frequency:4

Marks: 02

Input: N = 1002023 Output: digit: 0 Max Frequency:3

Note: For duplicate max frequency display any digit. For Example:

Marks: 02

N=10022

Output: digit: 0 Max Frequency 2 //OR

Output: digit: 2 Max Frequency 2

Note: Take input and display output in main function.

Total Marks: 06

4. Draw a bar chart of 5 values entered by the user (≥ 0), where the y-axis is along the rows and the x-axis is along the columns of the screen (as shown in the figure below).

Function Prototype: void barChart(int N1, int N2, int N3, int N4, int N5)

Example

01.*****marks: 02

Input:

N1=>8 , N2=>6 , N3=>2 , N4=>10 , N5=>9

```
10          *
09          * *
08 *        * *
07 *        * *
06 * *      * *
05 * *      * *
04 * *      * *
03 * *      * *
02 * * *    * *
01 * * *    * *
   1  2  3  4  5
```

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Example

02:

*****marks: 02

Input:

N1=>8 , N2=>6 , N3=>0 , N4=>11 , N5=>7

```

11      *
10      *
09      *
08 *      *
07 *      * *
06 * *      * *
05 * *      * *
04 * *      * *
03 * *      * *
02 * *      * *
01 * *      * *
    1  2  3  4  5

```

Example

03:

*****marks: 02

Input:

N1=>4 , N2=>3 , N3=>1 , N4=>0 , N5=>0

```

04 *
03 * *
02 * *
01 * * *
    1  2  3  4  5

```

Note: Take input in main function.

Total Marks: 06

5. Write and test the “digit” function:

Function Prototype: int digit(int n,int k)

This function returns the kth digit of the positive integer n. For example, if n is the integer 29,415, then the call digit(n,0) would return the digit 2, and the call digit(n,2) would return the digit 4.

Examples:

Marks: 02

Input: n = 29415 , k = 1 output: 9

Marks: 02

Input: n = 2 , k = 0 output: 2

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Marks: 03

Input: $n = 2$, $k = 1$ output: index out of bound (return -1)

Note:

- The digits are numbered from left to right beginning with the “zeroth” digit.
- Take input and display output in main function.

Total Marks: 06

6. Write a function named "reduce" that takes two positive integer arguments (as reference), call them "num" and "denom", treats them as the numerator and denominator of a fraction, and reduces the fraction.

The function should return the value 0 (to indicate failure to reduce: if either of the two arguments is zero) and should return the value 1 otherwise.

Function Prototype: `bool reduce(int & num, int & denom);`

Example01: *****Marks: 02

num= 10

denom=50

after calling reduce function

1. function must return 1
2. updated value of num will be 1
3. updated value of denom will be 5

Example02: *****marks: 01

num= 38

denom= 6

after calling reduce function

1. function must return 1
2. updated value of num will be 19
3. updated value of denom will be 3

Example03: *****marks: 01

num= 3

denom= -27

after calling reduce function

1. function must return 1
2. num will be 1
3. denom will be -9

Example04: *****marks: 01

num= -3

denom= -27

after calling reduce function

4. function must return 1
5. num will be 1
6. denom will be 9

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

Example05: *****marks: 01

num= 3

denom= 0

after calling reduce function

1. function must return 0
2. num will be 3
3. denom will be 0

Note:

- Do not use any built-in (math) functions (cmath library).
- Take input and display output in main function.

Total Marks: 34

7. In this problem, you are required to use for/while loop only to implement a calculator using ++ (increment) and -- (decrement) operators only. The operations that your calculator shall perform are limited, i.e. **addition, subtraction, multiplication, division, remainder and absolute**. Your program shall ask the user about the two input values and an operator, and then use the switch-case structure to run the required operation (addition, subtraction, multiplication, division or absolute (symbol ~)).

Prototype are:

- int addition(int n1,int n2);
- int subtraction(int n1,int n2);
- int multiplication(int n1,int n2);
- int division(int num,int denom);
- int remainder(int num,int denom);
- int absolute(int num);

op	n1	n2	Answer	Marks
+	10	12	22	1
+	10	-12	-2	2
+	-10	12	2	2
+	-10	-12	-22	1
-	10	12	-2	1
-	10	-12	22	1
-	-10	12	-22	2
-	-10	-12	2	2
*	10	12	120	1
*	10	-12	-120	2
*	-10	12	-120	2

**NATIONAL UNIVERSITY OF COMPUTER & EMERGING
SCIENCES ISLAMABAD CAMPUS**

*	-10	-12	120	1
/	20	3	6	1
/	20	-3	-6	1
/	-20	3	-6	1
/	-20	-3	6	1
/	3	5	0	1
/	3	0	error	1
%	20	3	2	3
%	3	5	3	1
~	5		5	1
~	-5		5	3

Note:

1. **Any operation that uses arithmetic operators (+, -, *, /, %) other than ++ or -- will be awarded zero marks.**
2. Take only one input for absolute function.
3. All functions inputs can be positive or negative except remainder function, which accept only positive values (>0).
4. Take input and display output in main function.
5. **02 Marks for Menu** (*menu must display your aesthetic sense*).