**PF Fall 2021 - Assignment 4**

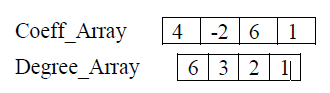
**Submission Instructions:**

1. You can email me ([aamina.batool@lhr.nu.edu.pk](mailto:aamina.batool@lhr.nu.edu.pk)) or your TA Salman ([l181127@lhr.nu.edu.pk](mailto:l181127@lhr.nu.edu.pk)), if you have any query.
2. Plagiarism tool will be used to check if you have cheated off of internet or any other source. If someone’s assignment is plagiarized, that student will be given F straight away so beware.
3. Make separate .cpp files for every question. Kindly name your .cpp files as Q1\_21L-XXXX. (XXXX here would be replaced by your roll number).
4. Add cpp files of all the questions in a folder. Name your folder by including your roll number as 21L-XXXX-A2 (A2 here means assignment 2) and then compress the folder as .rar, and submit it.
5. And kindly note that write your roll no. in the format given above and not as L21-XXXX or 21-XXXX because it makes it hard to find the submission of specific student.
6. There is straight**20 Percent deduction** of marks for students who submit more than 1 folder.
7. If any student resubmits make sure they remove the folder they last submitted from drive, by clicking the cross that shows when you have attached a file at Submission folder.
8. You have to upload your assignments in google classroom. Don’t forget to turn in after uploading your assignment.
9. The deadline for this assignment is **December 12, 2021, 11:59 pm**. No late submission will be accepted.

**Question 1.**

Write a C++ program called polynomial calculator which will perform some operations on polynomials. For each polynomial there are two important things which you need to store, first is degree of polynomial and second is coefficient of all terms. You can consider that maximum degree of all polynomials will be not more than 15 (You can define fixed size arrays of maximum size). So, you will store degrees of polynomial in one array and coefficients in second array accordingly.

For example, a polynomial: 4x6-2x3+ 6x2+1x would be stored as: Terms = 4



Make sure that the degree array is sorted in descending order so that the highest power term appears first. Also, there should be no duplicates in the degree array and it should only be allocated according to the total terms present in the polynomial.

**Functions:**

**1. Input:** This function will take a polynomial as input from user. It will populate values for degree and coefficient arrays according to number of terms, by taking all values as input from user. Prototype of function in given below.

void Input( int terms, int deg [], int coef []);

**2. Solve:** This function will take a polynomial and a variable as input. It will solve the polynomial for given value of variable and will return computed result. Prototype of function in given below.

int Solve( int terms, int deg [], int coef [], int variable);

**3. Add:** This function will take three polynomials as input. It will add first two polynomials and store the result of addition (which will also a polynomial) in third polynomial. Prototype of function in given below.

void Add ( int aterms, int adeg [], int acoef [],

int bterms, int bdeg [], int bcoef [],

int rterms, int rdeg [], int rcoef [] );

**4. Multiply:** This function will take three polynomials as input. It will multiply first two polynomials and store the result of multiplication (which will also a polynomial) in third polynomial. Prototype of function in given below.

void Multiply( int aterms, int adeg [], int acoef [],

int bterms, int bdeg [], int bcoef [],

int rterms, int rdeg [], int rcoef [] );

**5. Subtract:** This function will take three polynomials as input. It will subtract second polynomial (b) from first polynomial (a) and store the result of subtraction (which will also a polynomial) in third polynomial. Prototype of function in given below.

void Subtract( int aterms, int adeg [], int acoef [],

int bterms, int bdeg [], int bcoef [],

int rterms, int rdeg [], int rcoef [] );

**6. Print:** This function will take a polynomial as input, and will print that polynomial.

void Print( int terms, int deg [], int coef []);

For Example if polynomial has 4 terms and following data in coefficient and degree array.



Then output will be: 4x^6 -2x^3 +6x^2 +1x^1

**7. Equal:** This function will take two polynomials as input. It will check if both polynomials have same number of terms, degrees and corresponding coefficients then return true otherwise will return false. Prototype of function in given below.

void Equal(int aterms, int adeg [], int acoef [],

int bterms, int bdeg [], int bcoef []);

**8. Menu:** Test the polynomial functions via a menu based system. Design a Menu function which will take input an integer value from user and will execute all operations accordingly.

Example of menu options for different values are given below:

0. Exit the program.

1. Input one polynomial from user, first take number of terms from user and then for further data call Input function.

2. Take input of variable from user and call solve function. Then print result returned by solve function.

3. Create a polynomial for storage of result, call add function and then call print function for printing result.

4. Create a polynomial for storage of result, call multiply function and then call print function for printing result.

5. Create a polynomial for storage of result, call Subtract function and then call print function for printing result.

6. Ask user to print first or second polynomial and then call print function accordingly.

7. Call the Equal function and print **“Both polynomial are same”** if function returns true and **“Both are different”** if function returns false.

**Note:**

The arrays should be filled according to the total terms of the polynomial. You have to find out how addition, multiplication and subtraction of polynomials takes place. Also, you have to make sure that the resulting polynomial's degrees are sorted in descending order without any duplicates. You can implement any required helper private functions.

Finally design a main function and call the menu function.