

תכנות מונחה עצמים תש"ף - סמסטר ק'

**תרגיל בית מס' 1**

Class composition, C’tors, D’tors, static, const, reference, private, public, this

בתרגיל זה נעסוק בשברים של פולינומים בעל משתנה יחיד x.   
פולינום הוא סכום של מונומים בעל משתנה יחיד x.

דוגמא:

פולינום לדוגמא 1 +*x*2 − 2 · *x*  ·p(x)=3 .

שבר לדוגמא

כל פולינום ייוצג על ידי עצם במחלקה Polynomial. לכל עצם מסוג פולינום מוגדרת הדרגה המקסימלית שלו. הפולינום שבדוגמא יכול להיות מיוצג על ידי עצם מדרגה 2 ומעלה.

שבר (עצם במחלקה Rational) מיוצג על ידי שני פולינומים – המכנה והמונה.

○ העתיקו את התכנית הראשית המופיעה מטה לקובץ בשם Main.cpp. אין לשנות אותו במהלך העבודה.

○ כתבו את המחלקה Polynomial כך שהפלט של התכנית הראשית יהיה זהה לפלט שמופיע אחריו בקובץ זה.

דרישות נוספות:

○ המנעו משכפול קוד וקוד מיותר.

○ הדרגה של פולינום לא ניתנת לשינוי לאחר יצירתו, יש להגדיר אותה כ - const.

○ יש להשתמש ב const ו references בכל מקום שראוי להשתמש בהם.

**אין להשתמש במחלקה מסוג מונום. המחלקה פולינום לא מכילה אף אוביקט ששיך למחלקה אחרת.**

בנוסף המחלקה Polynomial תשמור את הדרגה המקסימלית של כל הפולינומים שהוגדרו. ניתן יהיה להדפיס ערך זה בכל נקודת זמן.

**הוראות הגשה**

* יש להגיש קובץ ZIP בשם
* 1\_HW1\_123456789\_987654321.zip

(יש להחליף את המספרים עם מספרי ת.ז. של המגישים.)

* על הקובץ להכיל את כל קבצי ה cpp ו h שכתבתם.
* שימו לב: הקובץ לא צריך להכיל את התיקיה שבה הקבצים נמצאים, אלא רק את הקבצים עצמם.
* רק אחד מהשותפים לתרגיל יגיש אותו ב MOODLE.

**תכנית ראשית**

// Polynomial.cpp : Defines the entry point for the console application.

//

//#include "stdafx.h"

#include <iostream>

#include "Polynom.h"

#include "rational.h"

void testPolynomial()

{

cout << "----- start testPolynomial ---------" << endl;

cout << "test0 :" << Polynomial::getMaxDegree() <<endl;

Polynomial p1;

cout << "test1: " <<p1;

Polinomial p2(6);

p2.setCoeff(2, 0.4);

cout << "test2: " << p2 << "degree=" << p2.getDegree(false)<<endl;

double c[5] = { 0.1, 0.2, 0 ,11.5, 1.3};

Polinomial p3(c,4);

cout << "test3: " << p3;

p2.setCoeff(2, 0.0);

cout << "test4: " << p2;

cout << "test6 :maxDegree = " << Polinomial::getMaxDegree() << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<< endl;

}

void testRational() {

cout << "----- start testRational ---------" << endl;

cout << "test7" << endl;

Rational r1;

r1.print();

cout << r1;

double c[] = { 0,2,2,3,4,5 };

Polinomial p1(c, 5);

c[0] = 1.2;

Polinomial p2(c, 3);

Rational r2(p1, p2);

cout << "test8" << endl;

r2.print();

cout << endl<< r2<<endl;

cout << "test9" << endl;

cout << "nom = " << r2.getNom();

cout << "denom = " << r2.getDenom();

cout << "test10 :maxDegree = " << Polinomial::getMaxDegree() << endl;

cout << "\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_"<<endl;

}

int main()

{

testPolynomial();

testRational();

}

----- start testPolynomial ---------

test0 :0

test1: Polinomial = 0

test2: Polinomial = 0+0X^1+0.4X^2

degree=6

test3: Polinomial = 0.1+0.2X^1+0X^2+11.5X^3+1.3X^4

test4: Polinomial =0

test6 :maxDegree = 4

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

----- start testRational ---------

test7

Polinomial = 0

--------------------------

Polinomial = 1

Polinomial = 0

--------------------------

Polinomial = 1

test8

Polinomial = 0+2X^1+2X^2+3X^3+4X^4+5X^5

--------------------------

Polinomial = 1.2+2X^1+2X^2+3X^3

Polinomial = 0+2X^1+2X^2+3X^3+4X^4+5X^5

--------------------------

Polinomial = 1.2+2X^1+2X^2+3X^3

test9

nom = Polinomial = 0+2X^1+2X^2+3X^3+4X^4+5X^5

denom = Polinomial = 1.2+2X^1+2X^2+3X^3

test10 :maxDegree = 5

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

בהצלחה !!