МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Інститут прикладної математики та фундаментальних наук

Кафедра прикладної математики

**ЗВІТ**

про виконання лабораторних робіт

з дисципліни

***«*Алгоритми та програмування, частина 2*»***

Виконала:  
студентка  
групи ПМ-11  
Вітюк Аліна  
Прийняв:   
Гладун В.Р.

**Лабораторна робота №3**

**Тема:** робота з масивами

**Мета:** набуття практичних навичок у роботі з масивами даних та відлагодження відповідних програм.

**Хід роботи:**

1. Постановку задачі: написати програму обчислення значення многочлена n-го степеня за заданими значеннями коефіцієнтів многочлена та аргументу х, а також значення похідної вказаного многочлена у цій же точці х.
2. Математичне обґрунтування алгоритму:

Для знаходження значення многочлена в точці х використала схему Горнера:

Знаходження похідної:

Для знаходження значення похідної в точці х використала схему Горнера:

1. Ідентифікатори:  
   х – аргумент;  
   Р – многочлен n - го степеня;  
   b[ ]– коефіцієнти похідної від многочлена;  
   а[ ] – коефіцієнти многочлена;  
   n – степінь многочлена;
2. Текст документованої програми:

//Програму обчислення значення многочлена n - го степеня за

//заданими значеннями коефіцієнтів многочлена та аргумента х, а також

//значення похідної вказаного многочлена у цій же точці х.

#include<iostream> // бібліотека і відповідний заголовний файл

#include <Windows.h> // підключення заголовного файлу

#include<math.h> // математична бібліотека

using namespace std; // директива для всіх ідентифікаторів в просторі імен

const int size1=20;

//Задання розмірності статичного масиву

void vvid(int m, double a[])

{

for (int i = 0; i < m; i++)

{

cout << "a\_" << i << " = ";

cin >> a[i];

}

}

//Цикл вводу коефіцієнтів многочлена

// m - кількість коефіцієнтів

// a[] - коеф. многочлена

void vyvid(int m, double a[])

{

if (m == 0)

{

cout << "(0)";

}

else

{

for (int i = 0; i < m - 1; i++)

cout << '(' << a[i] << ") \* x^" << m - i - 1 << " + ";

cout << '(' << a[m - 1] << ")";

}

}

//Цикл виводу коеф. многочлена на екран

double gorn(int m, double a[], double x)

{

long double res = a[0];

for (int k = 1; k < m; res = res \* x + a[k++]);

cout << res;

return res;

}

//Цикл обчислення значення многочлена за схемою Горнера

//m - кількість коефіцієнтів

//a[] - коефіцієнти

//x - аргумент

void form(int m, double a1[], double a2[])

{

for (int i = 0; i <= m; i++)

a2[i] = a1[i] \* (m - i);

}

//Цикл формування коефіцієнтів для похідної від многочлена

//m - кількість коефіцієнтів

//a1[] - коефіцієнти для многочлена

//a2[] - коефіцієнти для похідної

int main(){

SetConsoleOutputCP(1251); //Встановлення кодування Windows-1251 для виведення даних

long double P;

double x, a[size1 + 1], b[size1];

int n;

//х - аргумент

//Р - значення многочлена n - го степеня

//а[] - коефіцієнти многочлена

//b[] - коефіцієнти похідної від многочлена, b\_k=(n-k)\*a\_k, 0<=k<=n-1

//n - степінь многочлена

cout << "\nВведіть степінь многочлена (n - ціле додатне число)";

do

{

cout << "\nn = "; cin >> n;

} while (n < 0);

if (n == 0)

{

cout << "\nВведіть коефіцієнти многочлена a\_0 \* x^n + a\_1 \* x^(n-1) + ... + a\_(n - 1) \* x + a\_n:\n";

vvid(n + 1, a);

}

else

{

cout << "\nВведіть значення аргументу";

cout << "\nх = "; cin >> x;

cout << "\nВведіть коефіцієнти многочлена a\_0 \* x^n + a\_1 \* x^(n-1) + ... + a\_(n - 1) \* x + a\_n:\n";

vvid(n + 1, a);

}

//Ввід даних користувачем

cout << "\n............................................................................";

cout << "\n";

cout << "\nМногочлен n - го степеня:";

cout << "\nP(x) = ";

vyvid(n + 1, a);

//Вивід многочлена на екран

if (n == 0)

{

}

else

{

cout << "\nP(" << x << ") = ";

gorn(n + 1, a, x);

}

//Обчислення значення многочлена за заданими значеннями коефіцієнтів многочлена та аргумента х

cout << "\n";

cout << "\n........................................................................... ";

cout << "\n";

form(n, a, b);

cout << "\nПохідна від многочлена:";

cout << "\nP'(x) = ";

vyvid(n, b);

//Вивід похідної від многочлена на екран

if (n == 0)

{

}

else

{

cout << "\nP'(" << x << ") = ";

gorn(n, b, x);

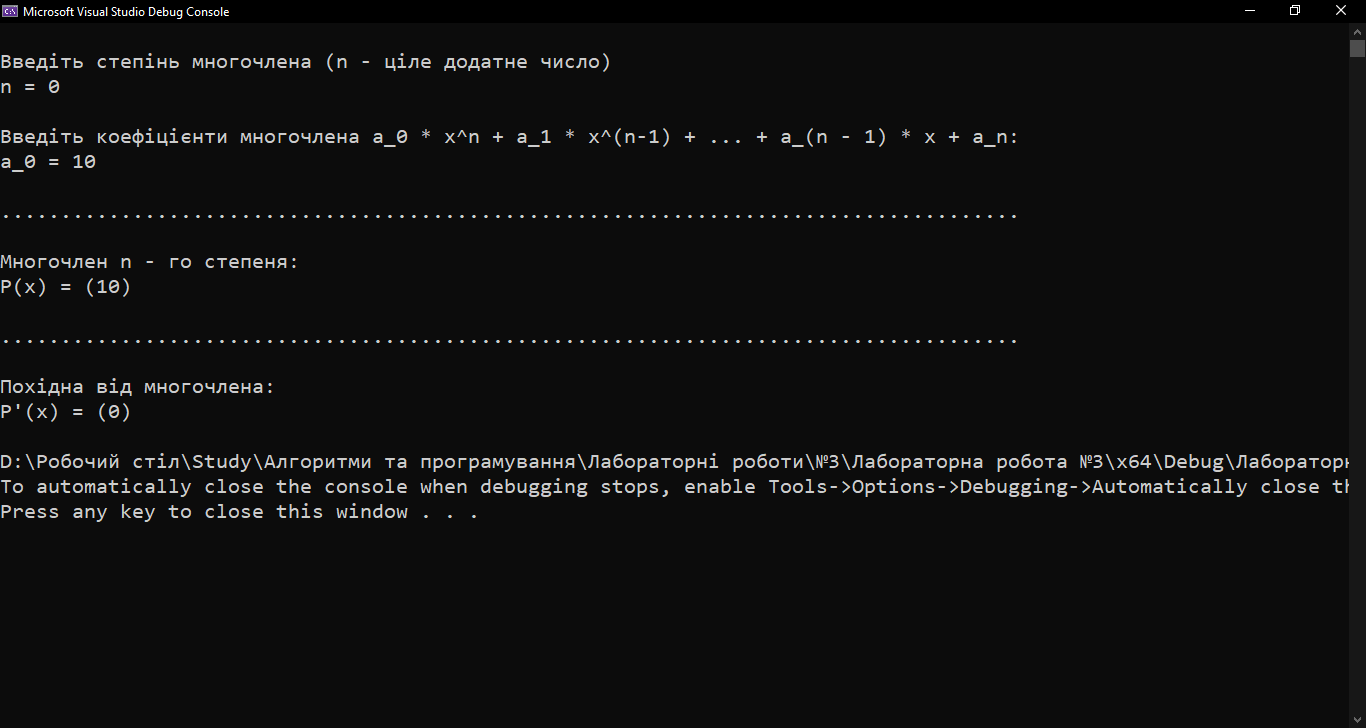
}

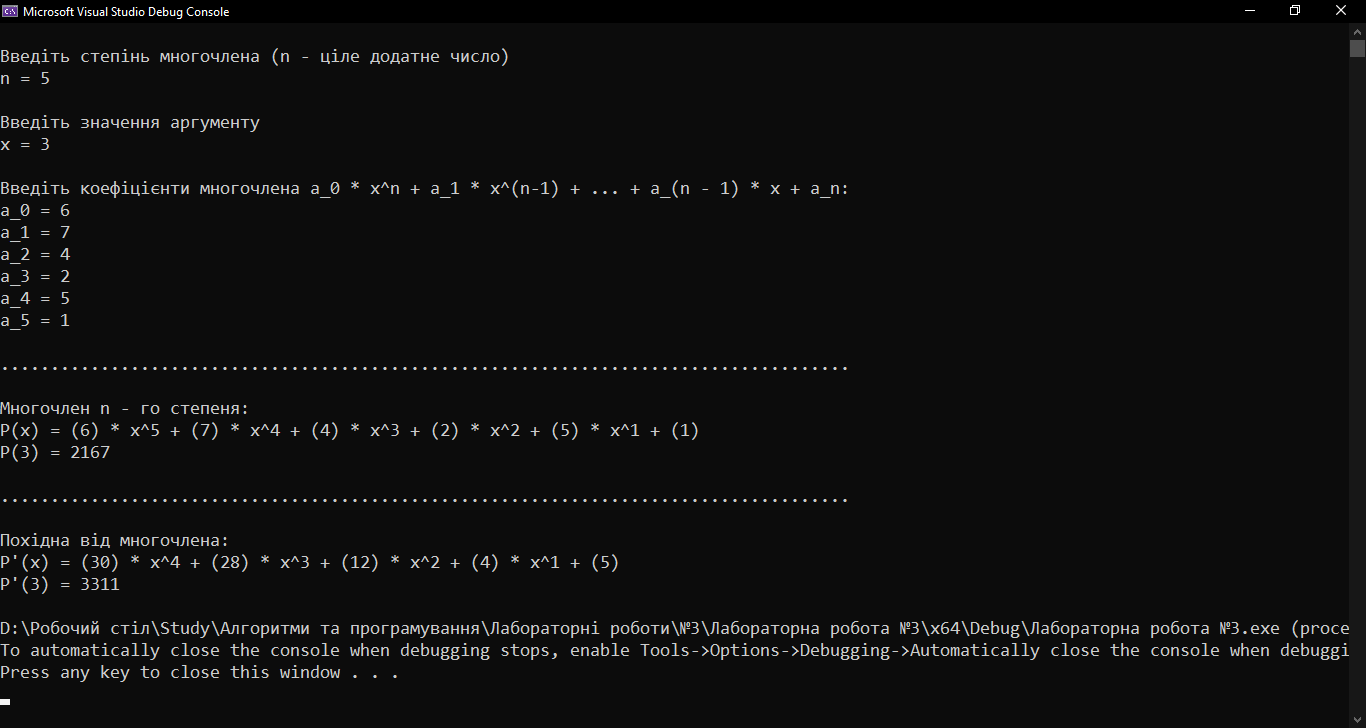
//Обчислення значення похідної в заданій точці х

cout << "\n";

return 0;

}

1. Результат виконання програми:  
   



1. Програма призначена для обчислення значення многочлена n-го степеня за заданими значеннями коефіцієнтів многочлена та аргументу х, а також значення похідної вказаного многочлена у цій же точці х.Після запуску коду програми на екран, користувачу необхідно ввести з клавіатури бажані значення n (степінь многочлена), x(аргумент)   
    (коефіцієнти многочлена). Слід зазначити, що існують певні вимоги до значення ідентифікатора n, з ними користувач може ознайомитися безпосередньо перед введенням даних. Після успішного введення даних, на екран виводиться многочлен , значення многочлена в точці x, похідна від многочлена і значення похідної в точці х.
2. Модифікації програми: удосконалила програму для многочленів  
   0-го степеня.

**Висновок:** виконання лабораторної роботи №3 допомогло засвоїти практичні навички у роботі з масивами даних. Також навчилася відлагоджувати цикл для обчислення многочленів в певній точці за допомогою схеми Горнера.