**ЛАБОРАТОРНА РОБОТА № 3**

**Арифметичні операції. Декремент. Інкремент**

***Мета:*** вивчити особливості використання оперціїї інкременту та дикременту та їх форми.

**Хід роботи:**

Завдання 1: Написати програму згідно вашого варіанту, яка виводить на екрані дані так, як представлено у таблиці. При цьому використовуйте можливість задати ширину поля, а також вирівнювання по лівому і правому краях.

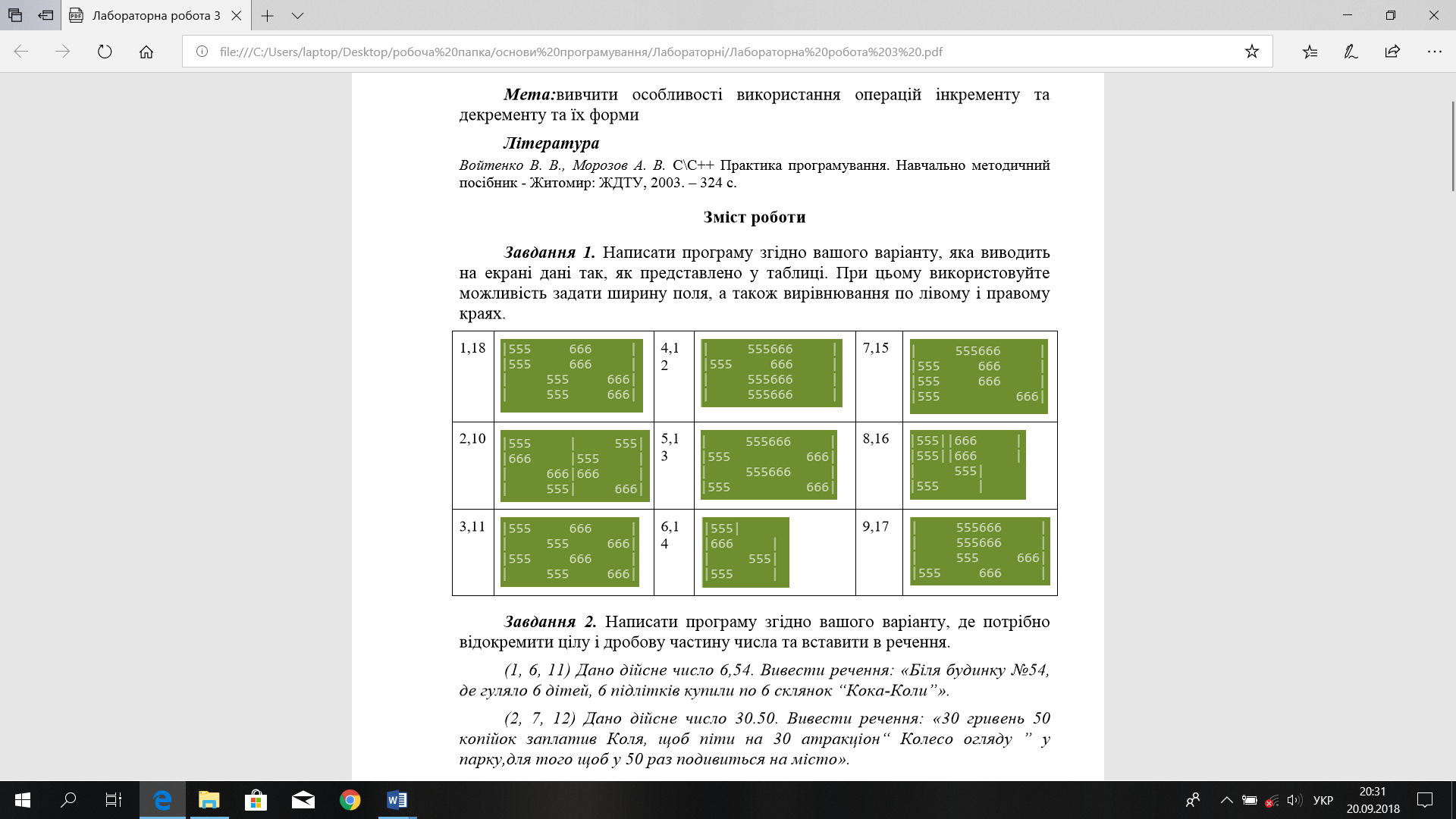


Рис 1. Умова до завдання № 1

Лістинг програми:

#include "pch.h"

#include <stdio.h>

int main()

{

// -6d пропуски вправо, 6d пропуск а потім число

printf("|%-6d|%6d|\n", 555, 555);

printf("|%-6d|%-6d|\n", 666, 555);

printf("|%6d|%-6d|\n", 666, 666);

printf("|%6d|%6d|\n", 555, 666);

return 0;

}

Результат виконання програми:

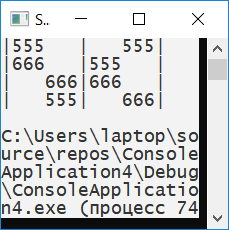


Рис. 2. Результат виконання програми

***Завдання 2.*** Написати програму згідно вашого варіанту, де потрібно відокремити цілу і дробову частину числа та вставити в речення.

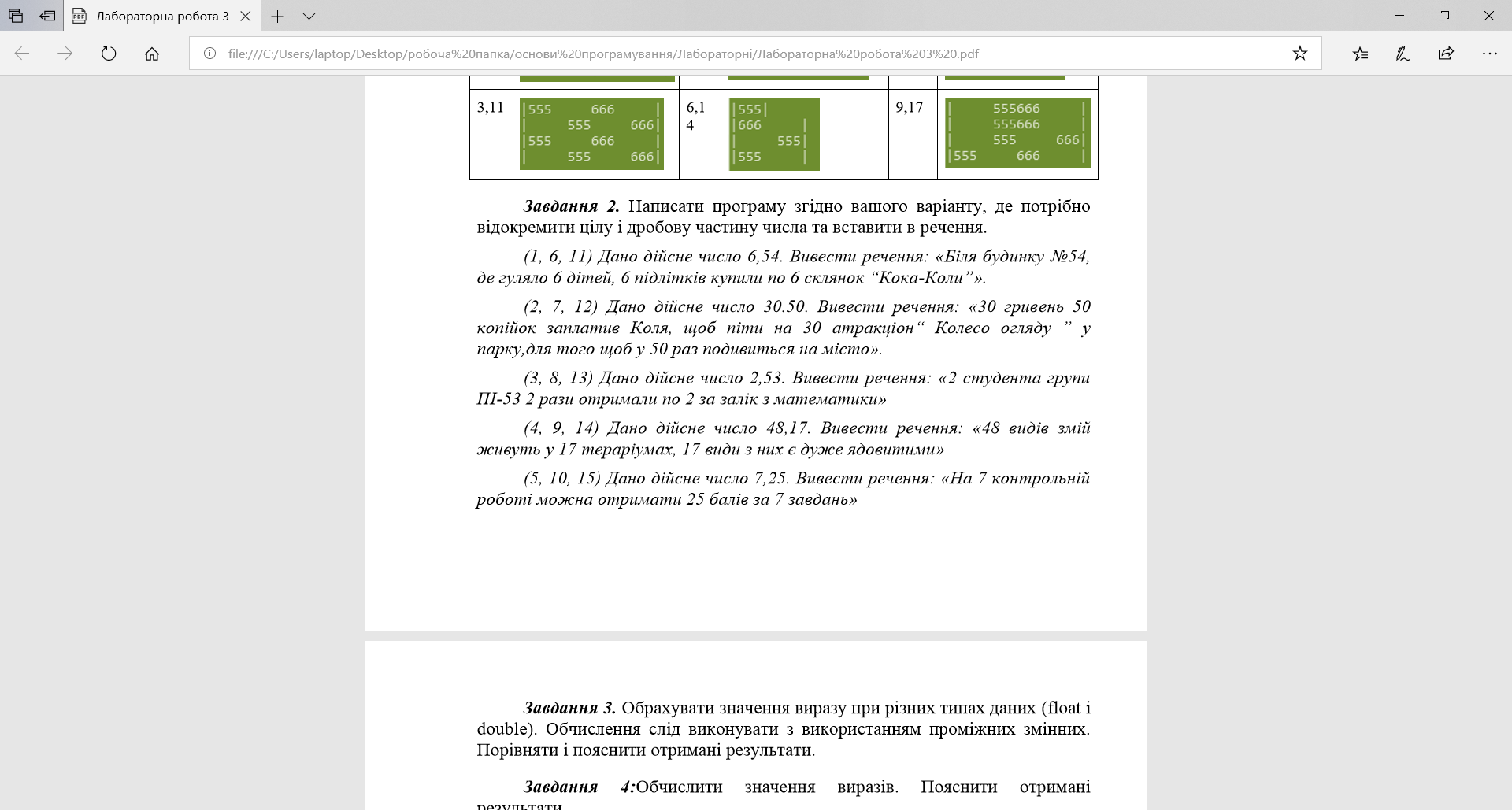


Рис 3. Умова до завдання для виділення цілої та дробової частини

Лістинг програми:

#include "pch.h"

#include <stdio.h>

#include <math.h>

#include "windows.h"

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

double a = 7.25, x, y; // ввожу 3 буквы (1 выраз, 2 остальные - целая часть и дробная)

y = modf(a, &x); // modf - розбиває число на целую и дробовую (y - дробовая, x - целая)

printf("\n\nНа %.0f контрольній роботі можна отримати %.0f балів за %.0f завдань\n\n\n", x, y \* 100, x);

return 0;

}

Результат виконання програми:

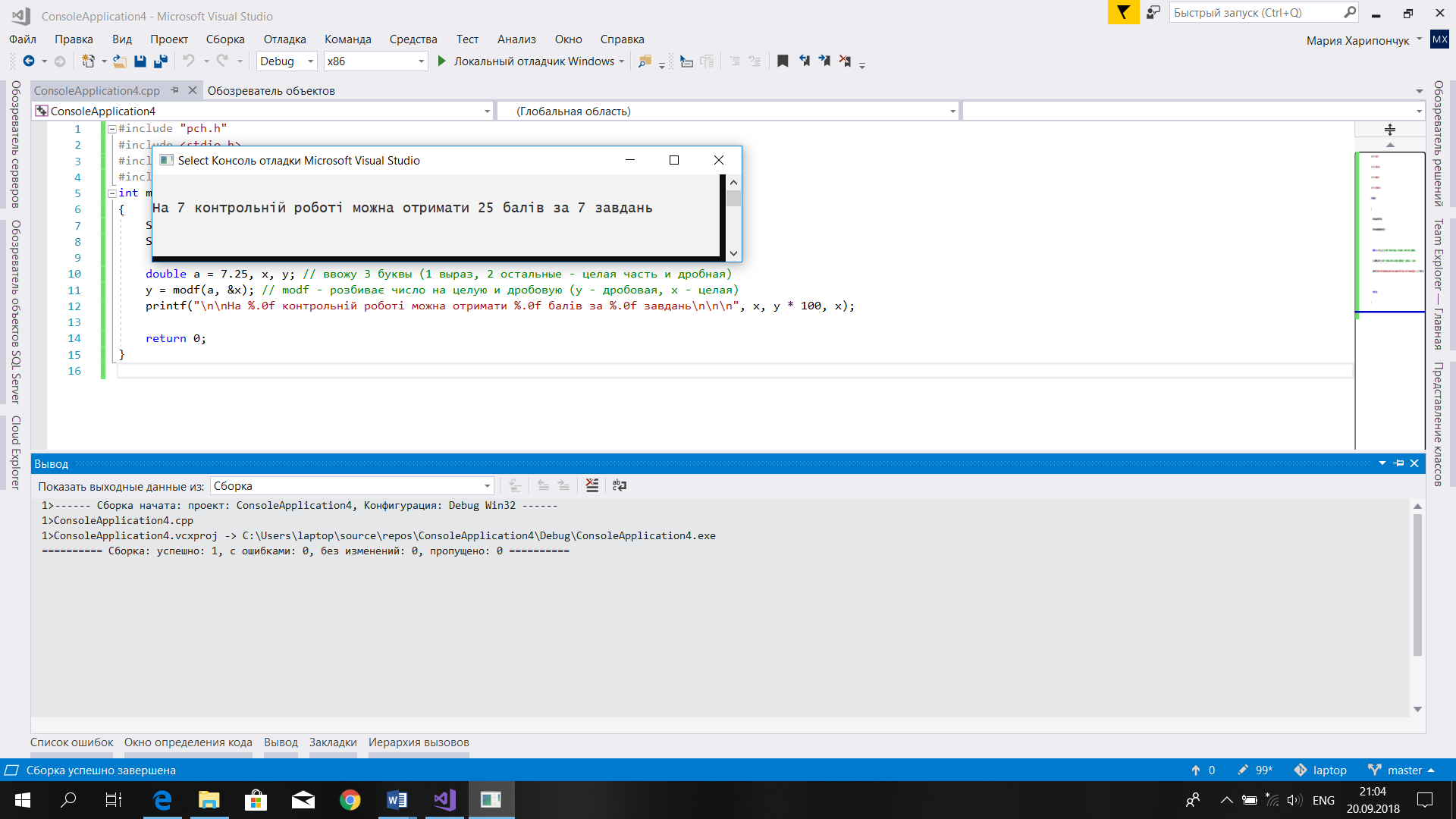


Рис. 4. Результат виконання програми для виділення цілої та дробової частини числа і вставлення їх в речення

***Завдання 3.*** Обрахувати значення виразу при різних типах даних (float і double). Обчислення слід виконувати з використанням проміжних змінних. Порівняти і пояснити отримані результати.

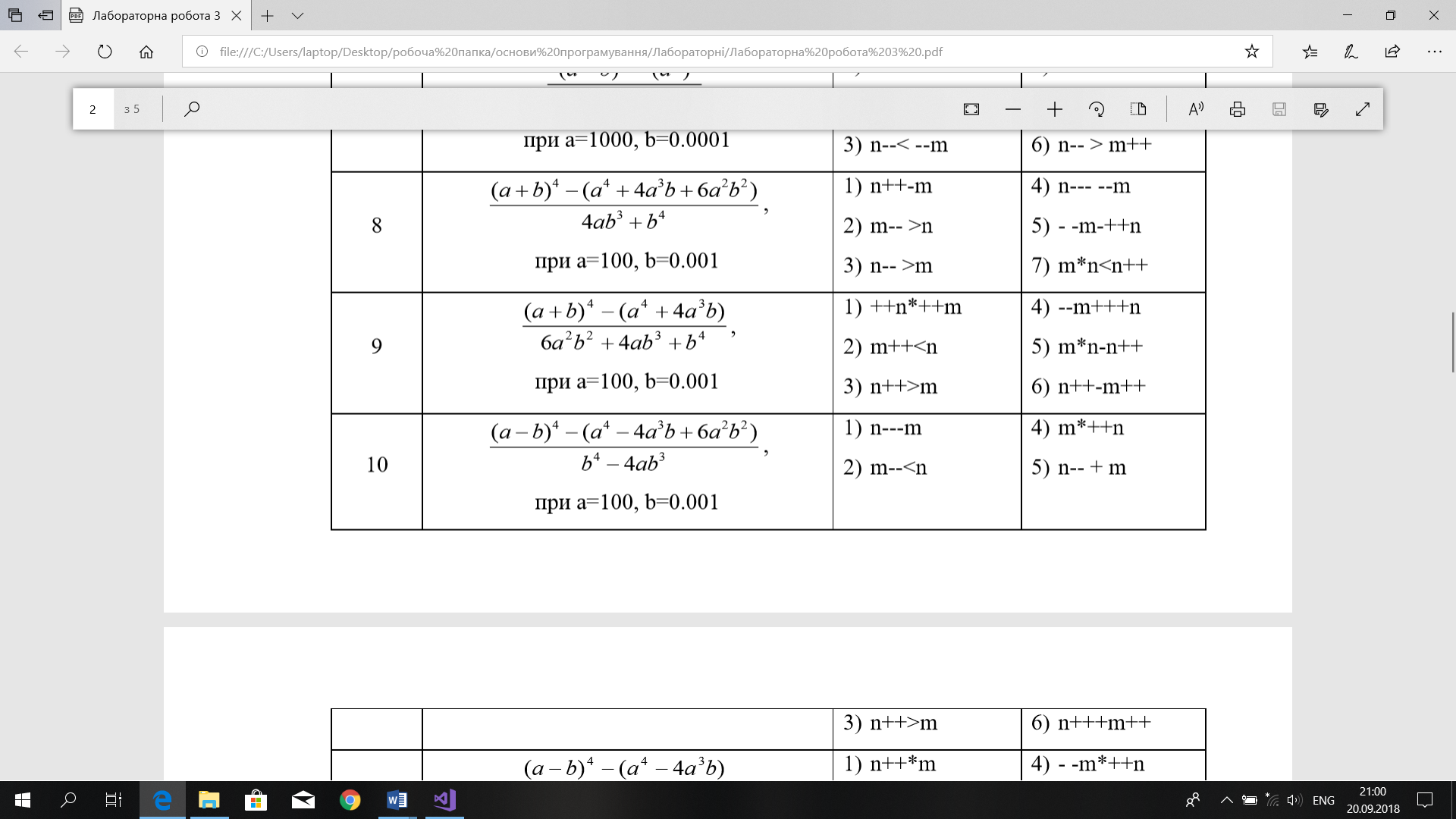


Рис 5. Умова до завдання №3

Лістинг програми:

#include "pch.h"

#include <stdio.h>

#include <math.h>

#include "windows.h"

int main()

{

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

float a, b, c, d, p1; // тут до 6 цифер

printf("\n Введите a1="); scanf\_s("%f", &a);

printf("\n Введите b1="); scanf\_s("%f", &b);

c = (pow(a - b, 4)) - (pow(a, 4) - pow(a, 3) \* b \* 4 + pow(a, 2)\*pow(b, 2) \* 6.0);

d = pow(b, 4) - pow(b, 3) \* a \* 4.0;

p1 = c / d;

printf("\nРезультат float : %f\n\n", p1);

double z, n, m, g, p2; // будет разница в числах здесь может доходить до 16 и варавниватся в маленьких числах

printf("\n Введите a2="); scanf\_s("%f", &z);

printf("\n Введите b2="); scanf\_s("%f", &n);

m = (pow(a - b, 4)) - (pow(a, 4) - pow(a, 3) \* b \* 4 + pow(a, 2)\*pow(b, 2) \* 6.0);

g = pow(b, 4) - pow(b, 3) \* a \* 4.0;

p2 = m / g;

printf("\nРезультат double: %f\n\n", p2);

return 0;

}

Результат виконання програми:

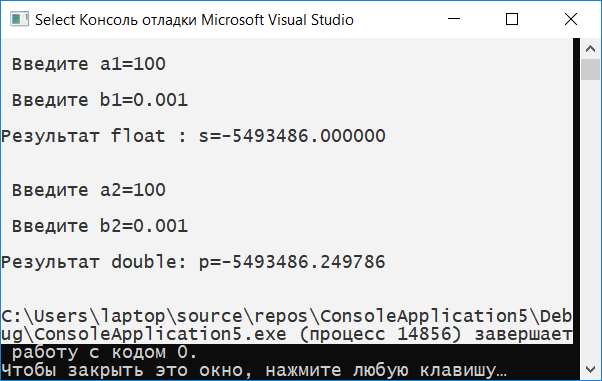


Рис 6. Результати обчислення приклада з різними типами данних

Данна программа показує те, що тип данних float має більшу точність, адже данні типу double були округлені до цілого числа.

***Завдання 4.*** Обчислити значення виразів. Пояснити отримані результати.

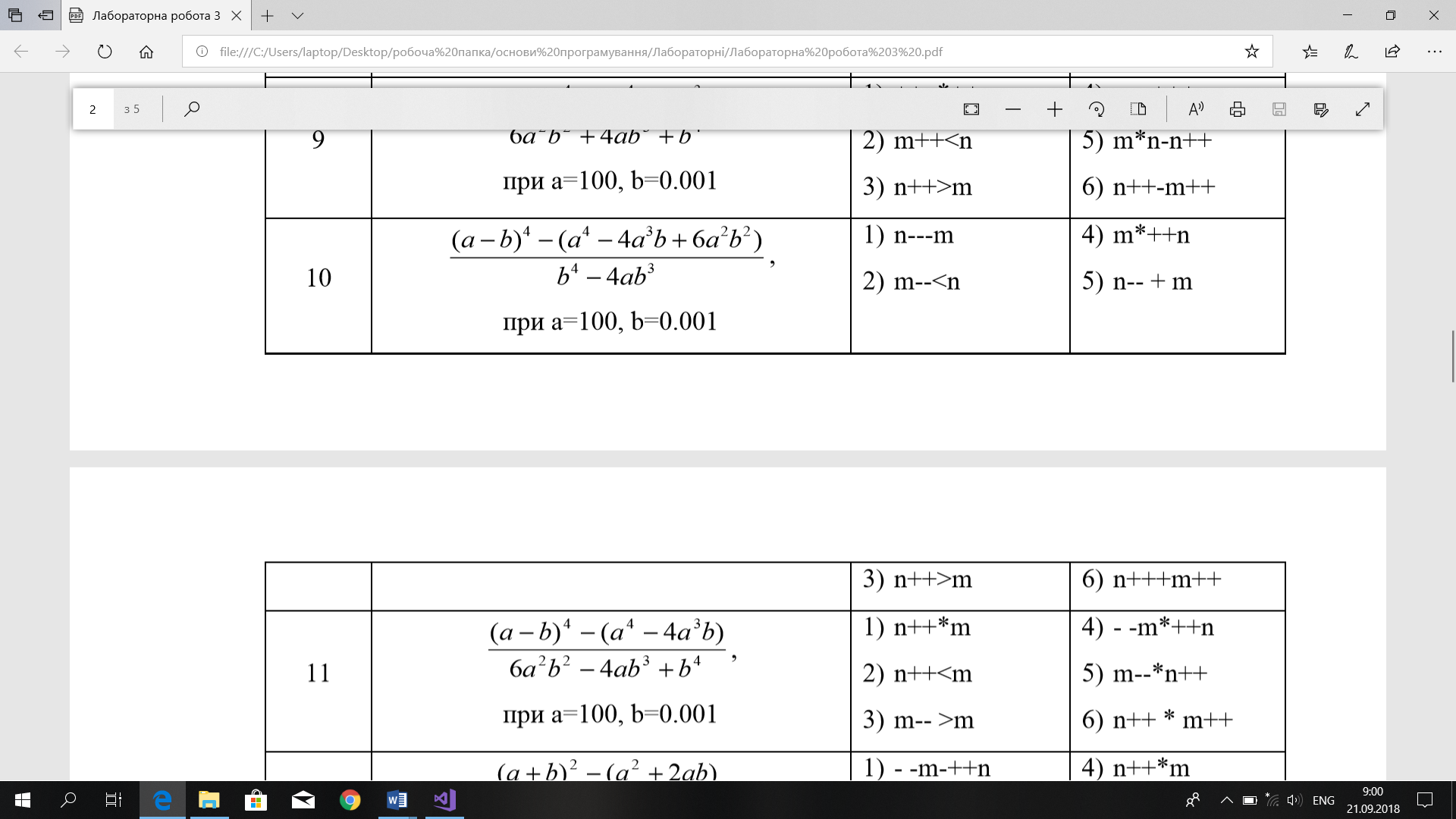
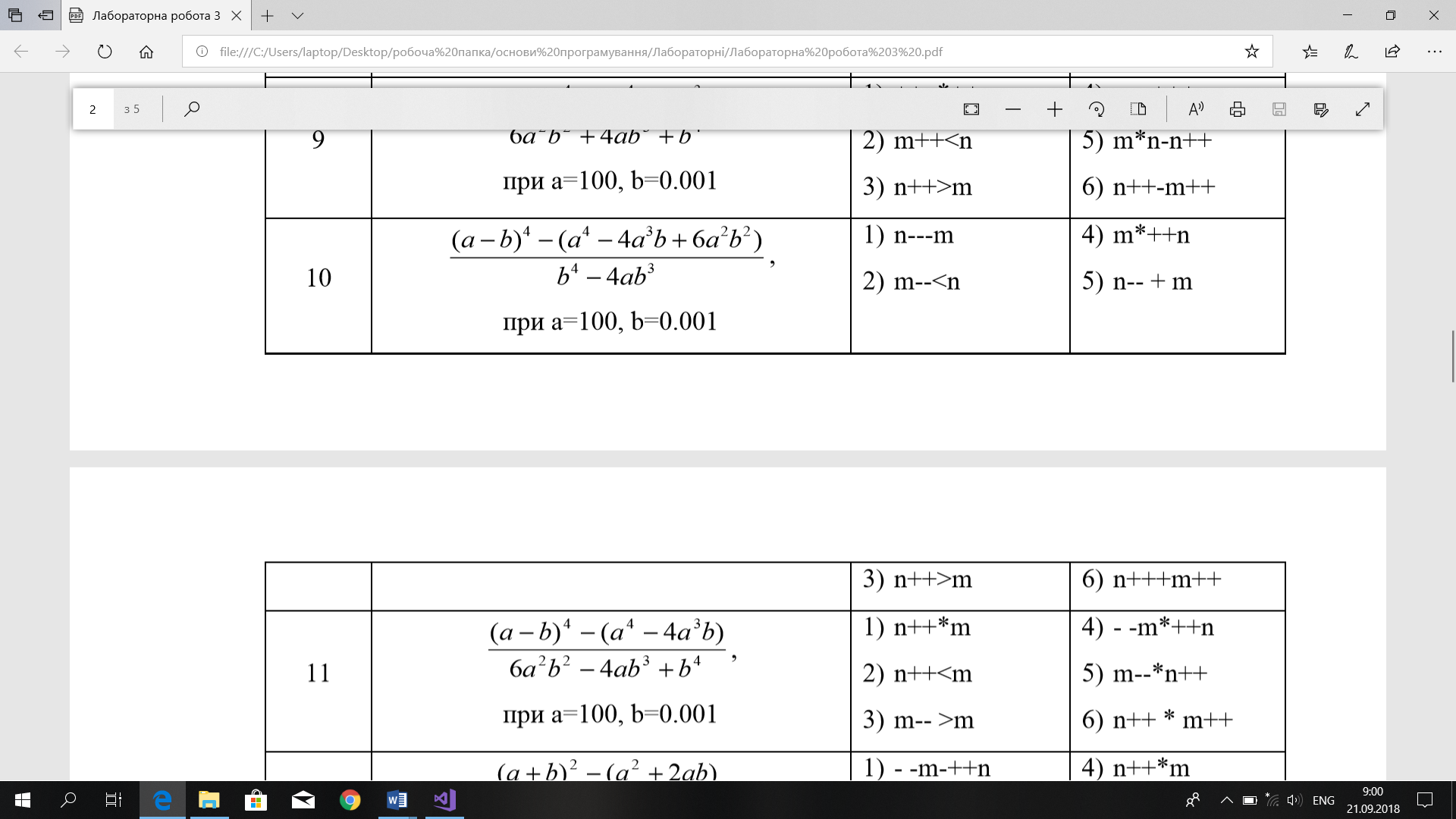


Рис 7. Умова до завдання №4

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <math.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int m = 5, n = 10, res;

//Пример 1.

res = n- --m; //1 формула.

printf("----------------------------------------------------------------\n");

printf("Пример 1.\n");

printf("----------------------------------------------------------------\n");

printf(" res = n- --m\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) m уменьшается на 1\n 2) n-m (10 - 4)\n 3) res = n-m (10 - 4) = 6\n");

printf("----------------------------------------------------------------\n");

//Пример 2.

m = 5, n = 10;

res = m-- < n; //2 формула.

printf("Пример 2.\n");

printf("----------------------------------------------------------------\n");

printf(" res = m-- < n\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) res = m < n (5 < 10) = В переменную попадает\n булевое значение 1 (true)\n 2) m уменьшается на 1\n");

printf("----------------------------------------------------------------\n");

//Пример 3.

m = 5, n = 10;

res = n++ > m; //3 формула.

printf("Пример 3.\n");

printf("----------------------------------------------------------------\n");

printf(" res = n++ > m\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) res = n > m (10 > 5) = В переменную попадает\n булевое значение 1 (true)\n 2) n увеличивается на 1\n");

printf("----------------------------------------------------------------\n");

//Пример 4.

m = 5, n = 10;

res = m \* ++n; //4 формула.

printf("Пример 4.\n");

printf("----------------------------------------------------------------\n");

printf(" res = m \* ++n\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) n увеличивается на 1\n 2) res = m \* n (5 \* 11) = 55\n");

printf("----------------------------------------------------------------\n");

//Пример 5.

m = 5, n = 10;

res = n-- + m; //5 формула.

printf("Пример 5.\n");

printf("----------------------------------------------------------------\n");

printf(" res = n-- + m\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) res = n + m (10 + 5) = 15\n 2) n уменьшается на 1\n");

printf("----------------------------------------------------------------\n");

//Пример 6.

m = 5, n = 10;

res = n++ + m++; //6 формула.

printf("Пример 6.\n");

printf("----------------------------------------------------------------\n");

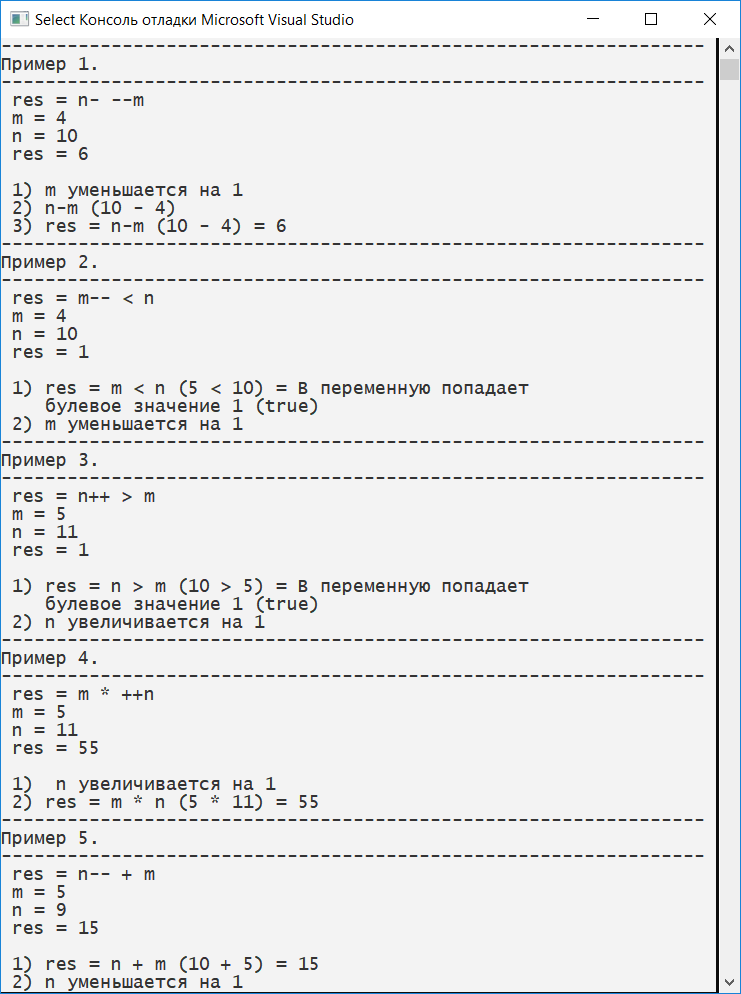
printf(" res = n++ + m++\n m = %d\n n = %d\n res = %d\n\n", m, n, res);

printf(" 1) res = n + m (10 + 5) = 15\n 2) n увеличивается на 1\n 3) m увеличивается на 1\n");

printf("----------------------------------------------------------------\n");

}

Результат виконання програми:



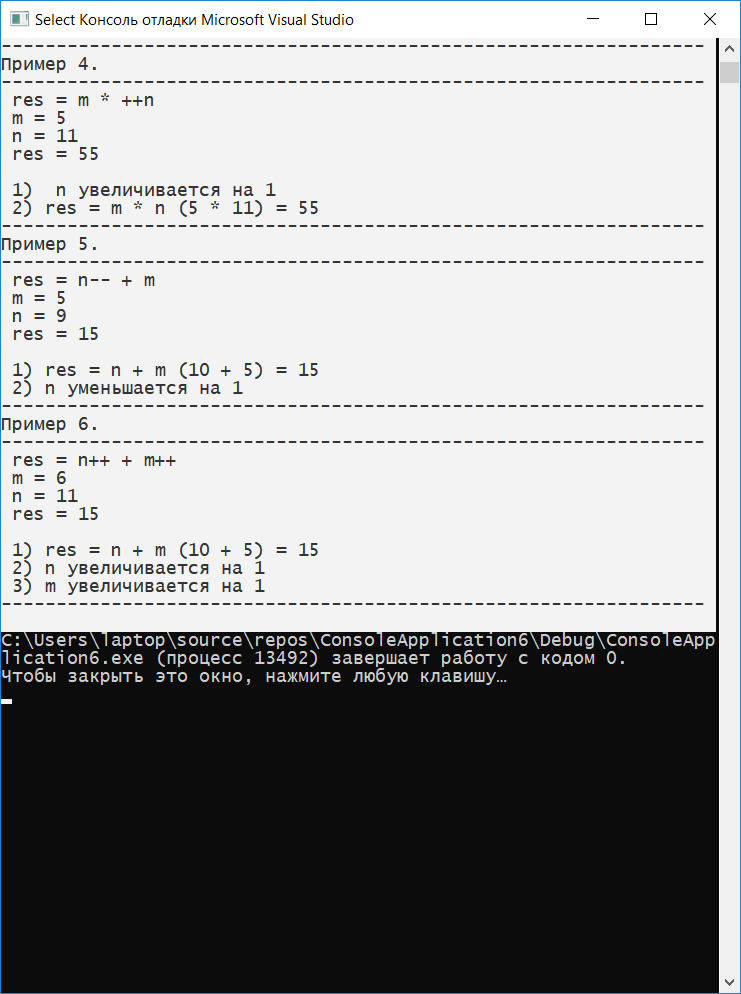
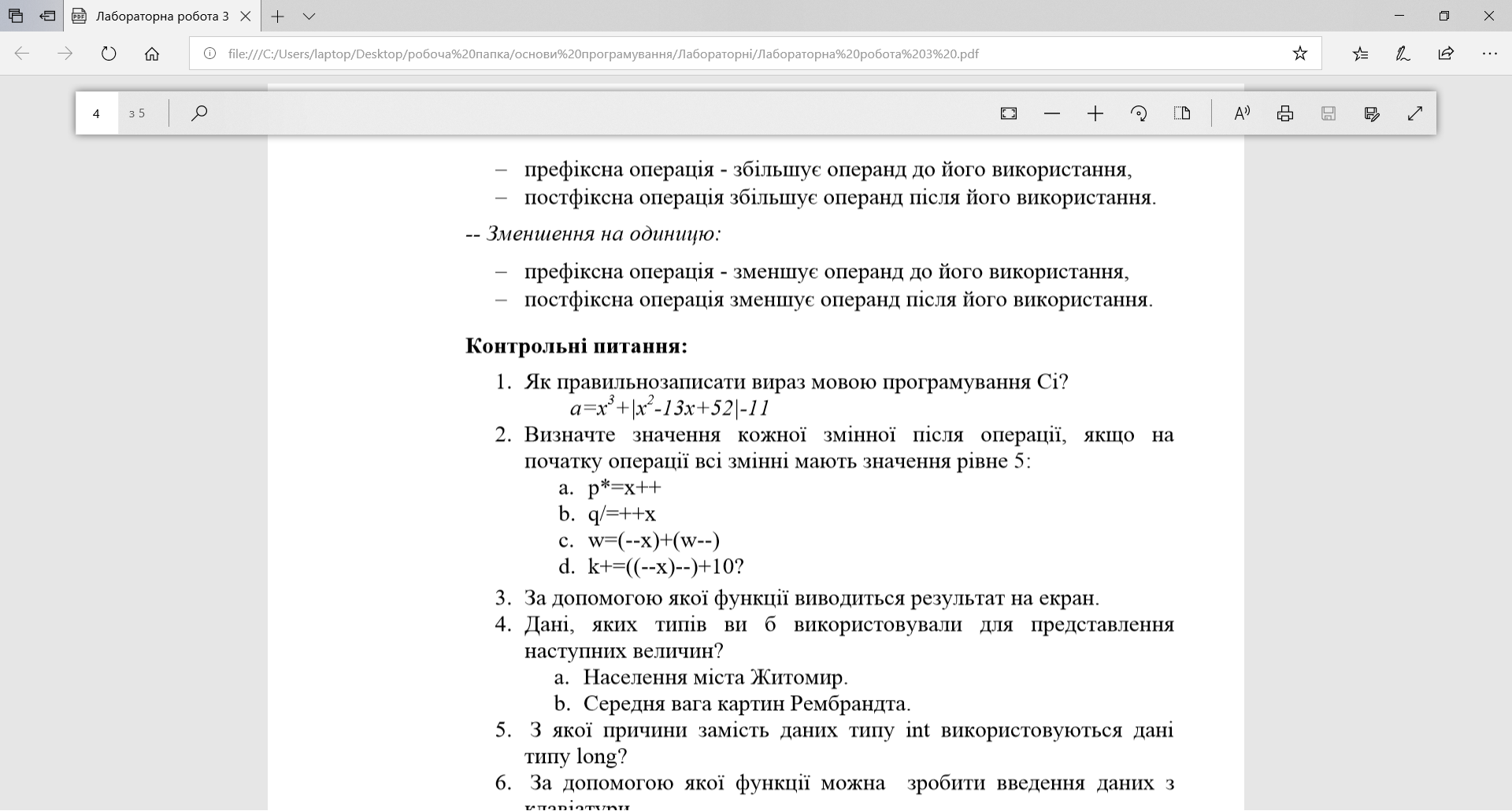


Рис 8. Результат виконання програми:

**Контрольні питання:**

**1.** Як правильно записати вираз мовою програмування Сі?



a=pow(x,3)+fabs(pow(x,2)-13\*x+52)-11

**2.** Визначте значення кожної змінної після операції, якщо на початку операції всі змінні мають значення рівне 5:

a. p\*=x++

b. q/=++x

c. w=(--x)+(w--)

d. k+=((--x)--)+10?

**3**. За допомогою якої функції виводиться результат на екран.

Результат функції виводиться на екран за допомогою функції printf( ) - форматне виведення на екран:

int printf(char \*format, <список виводу >);

**4.** Дані, яких типів ви б використовували для представлення наступних величин?

a. Населення міста Житомир.

b. Середня вага картин Рембрандта.

a. Для представлення величин кількості населення міста використовуються цілочисельні типи величин, тобто числові типи даних.

b. основні типи даних (арифметичні).

**5.** З якої причини замість даних типу int використовуються дані типу long?

При використанні модифікаторів short і long дозволяється опускати ім'я типу int.

**6.** За допомогою якої функції можна зробити введення даних з клавіатури.

Введення даних з клавіатури можна зробити за допомогою функції scanf( ) - форматне введення з клавіатури:

int scanf(char \*format, <список введення >);

**7.** Визначите, що являють собою наступні керуючі послідовності:

a. Символ переведення на наступний рядок записується як рядкова константа, тобто "\n"

b.  відступ —константа «\t».

**8.** Визначите, якого типу наступні константи:

a. ‘\ b’ - возврат назад – символьна константа

b. 1066 – ціла константа

c. 99.44 – константа з плаваючою точкою – змінний тип

**9.** Назвіть пріоритети виконання арифметичних операцій.

Пріоритети виконання арифметичних операцій від найвищого до найнижчого:

1. \*, /, % мультиплікаційні

2. +, – адитивні

3. >, >=, <, <= відношення

4. ==, !=

5. =, \*=, /=, %=,+=, –=

***Висновки:*** в ході виконання лабораторної роботи було ознайомлено з середовищем MS Visual Studio. Досліджено та отримано практичні навики щодо створення найпростішої програми. Вивчено особливості використання операцій інкременту та декременту та їх форми.