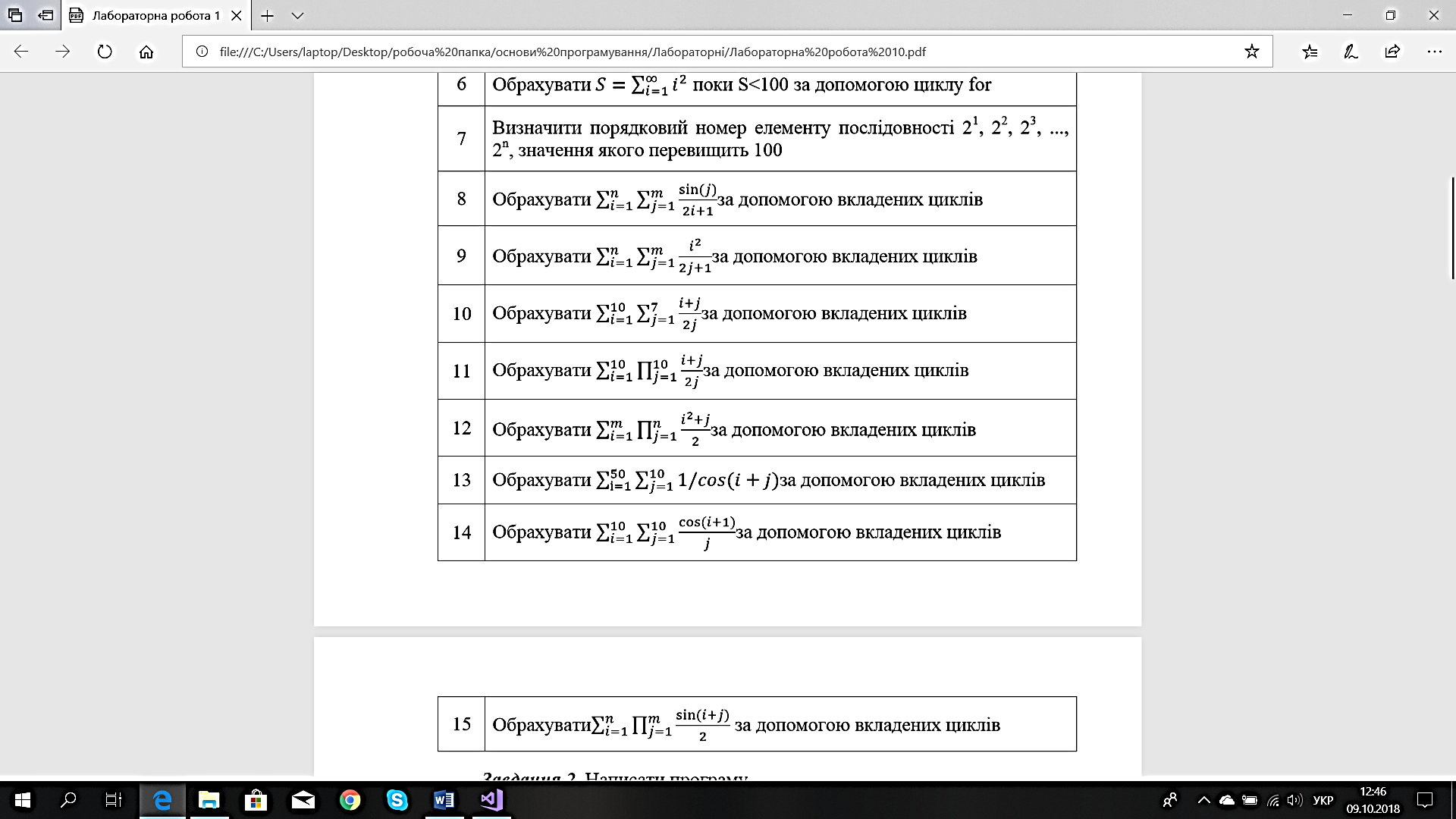
**ЛАБОРАТОРНА РОБОТА № 10**

**Вкладені цикли**

***Мета:*** набуття навичок роботи з різними типами операторів циклу.

**Хід роботи:**

**Завдання 1**: Написати програму.



Умова до завдання № 1

#include "pch.h"

#include <iostream>

#include <locale.h>

#include <math.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int i, j, sum = 0;

for (i = 1; i < 10; i++)

for (j = 1; j < 7; j++)

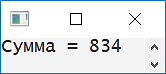
sum += (i + j) / 2 \* j;

printf("Сумма = %d\n", sum);

return 0;

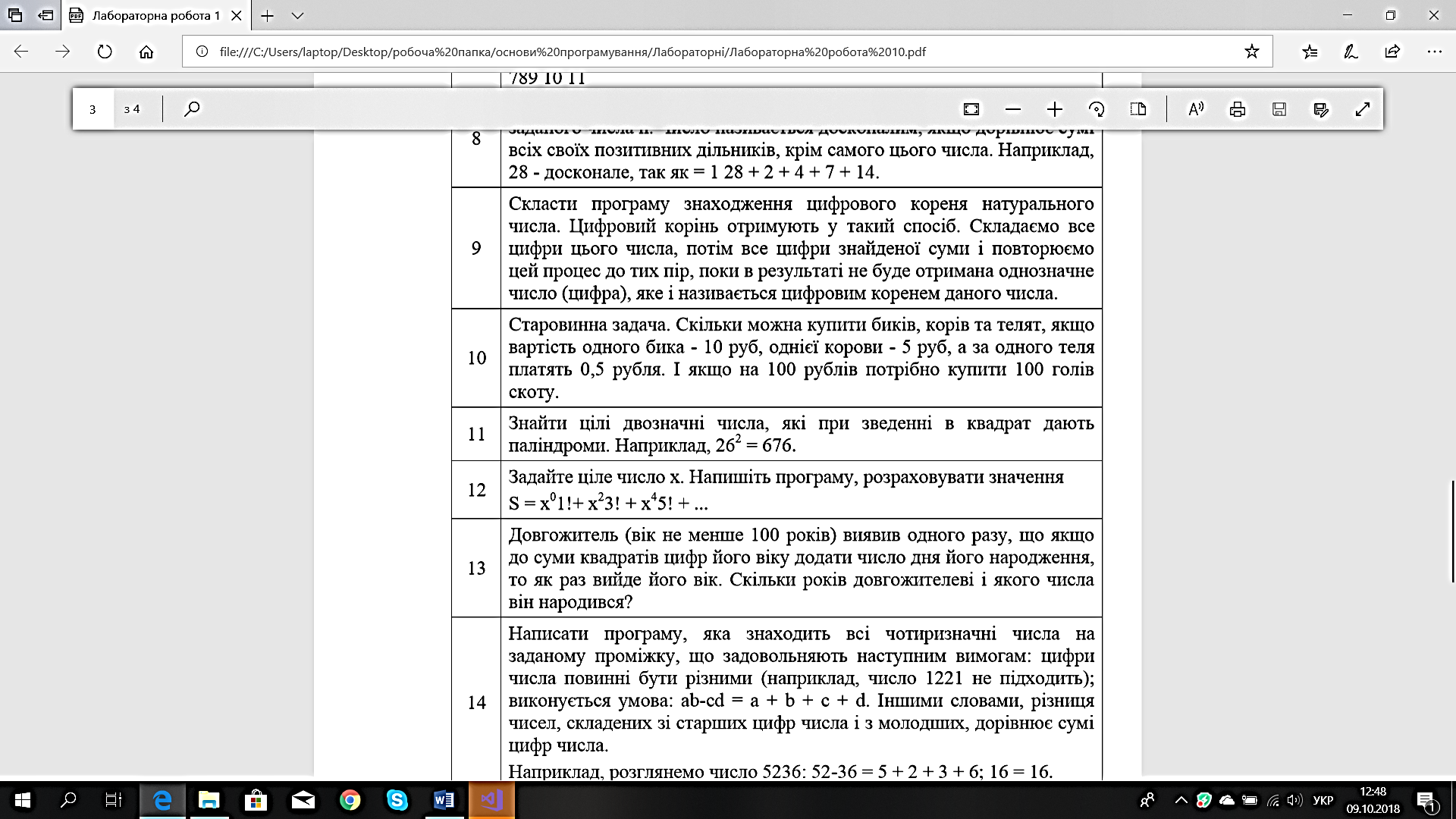
}

Результат виконання програми:



Результат програми для рішення квадратного рівняння.

**Завдання 2:** Написати програму.



Умова до завдання №2

Лістинг програми:

#include "stdafx.h"

#include <stdio.h>

#include <locale.h>

#include <math.h>

int main()

{

setlocale(LC\_ALL, "Rus"); //byk=10, cow=5, telya=0.5

double b, c, t;

for (b = 0; b <= 1; b++) {

for (c = 0; c <= 9; c++) {

for (t = 0; t <= 90; t++) {

if (b + c + t == 100)

printf("Биков =%f, коров =%f, телят =%f\n", b, c, t);

}

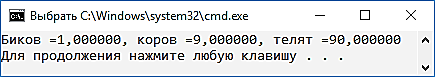
}

}

return 0;

}

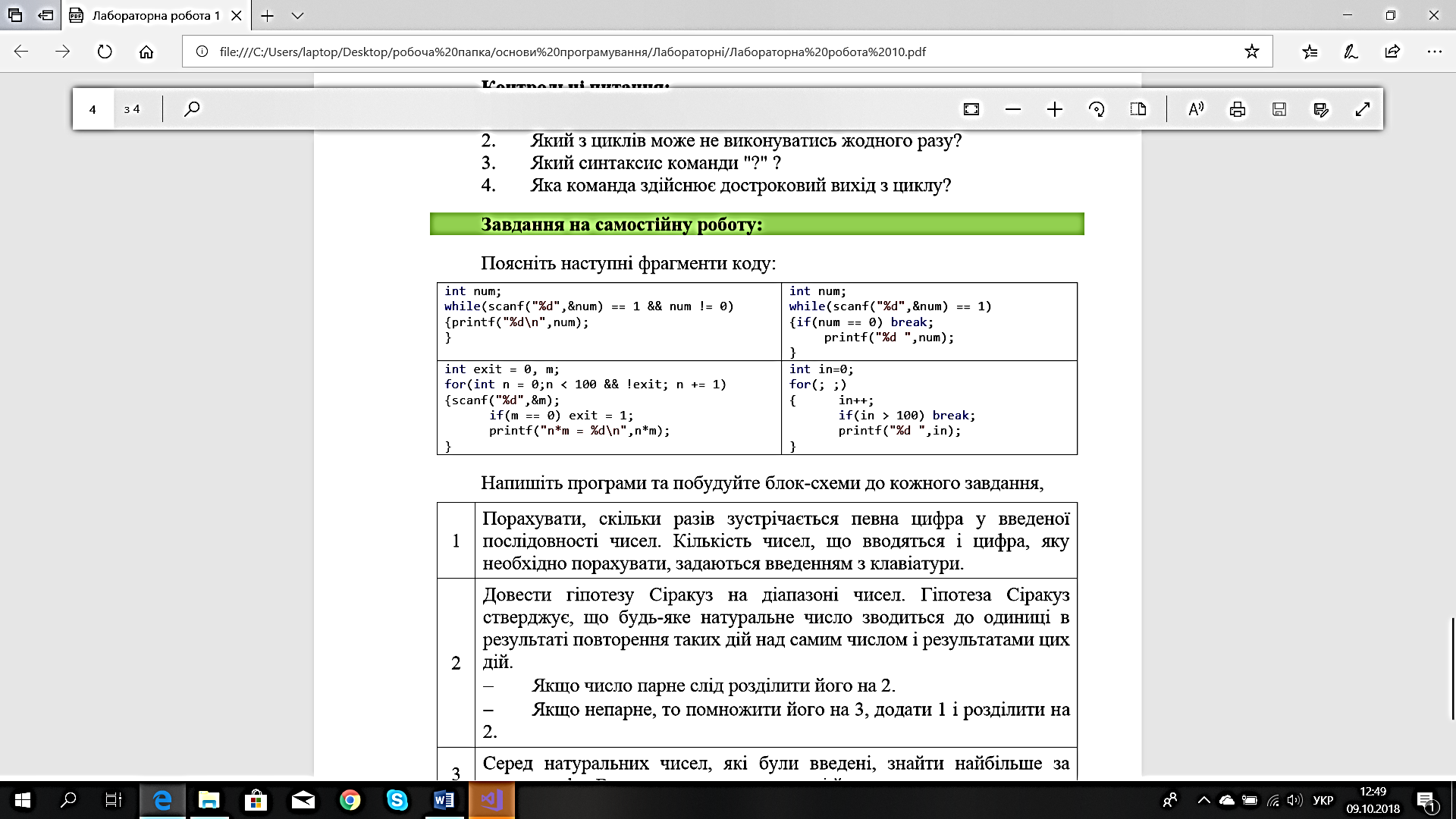
Результат виконання програми:



Результат виконання програми до завдання №2

**Завдання на самостійну роботу:**

1. Поясніть наступні фрагменти коду:



Умова до завдання на самостійну роботу (1)

Лістинг программи:

#include "stdafx.h"

#include <stdio.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int num; //вводим переменную: num

while (scanf\_s("%d", &num) == 1 && num != 0)

{

printf("%d\n", num);

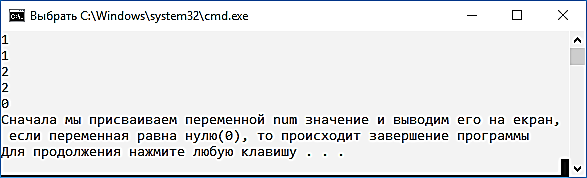
}

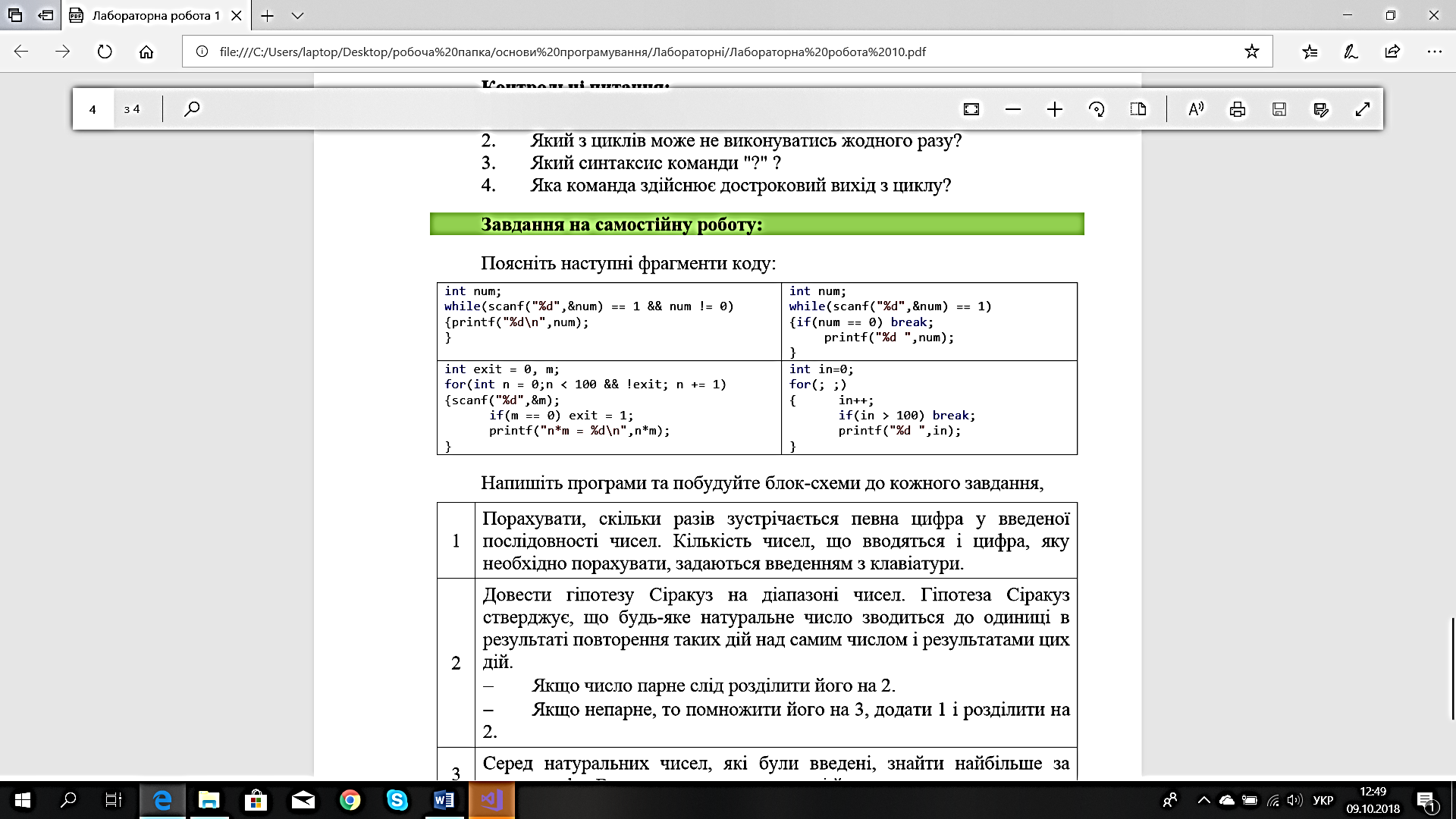
printf("Сначала мы присваиваем переменной num значение и выводим его на екран,\n если переменная равна нулю(0), то происходит завершение программы\n");

return 0;

}

Результат виконання програми:





Умова до завдання на самостійну роботу (2)

Лістинг программи:

#include "stdafx.h"

#include <stdio.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int num;

while (scanf\_s("%d", &num) == 1) {

if (num == 0) break;

printf("%d \n", num);

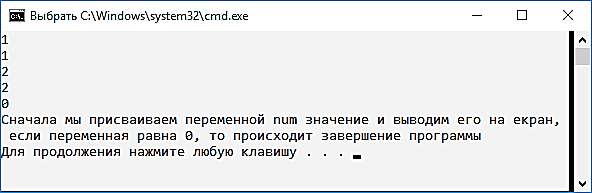
}

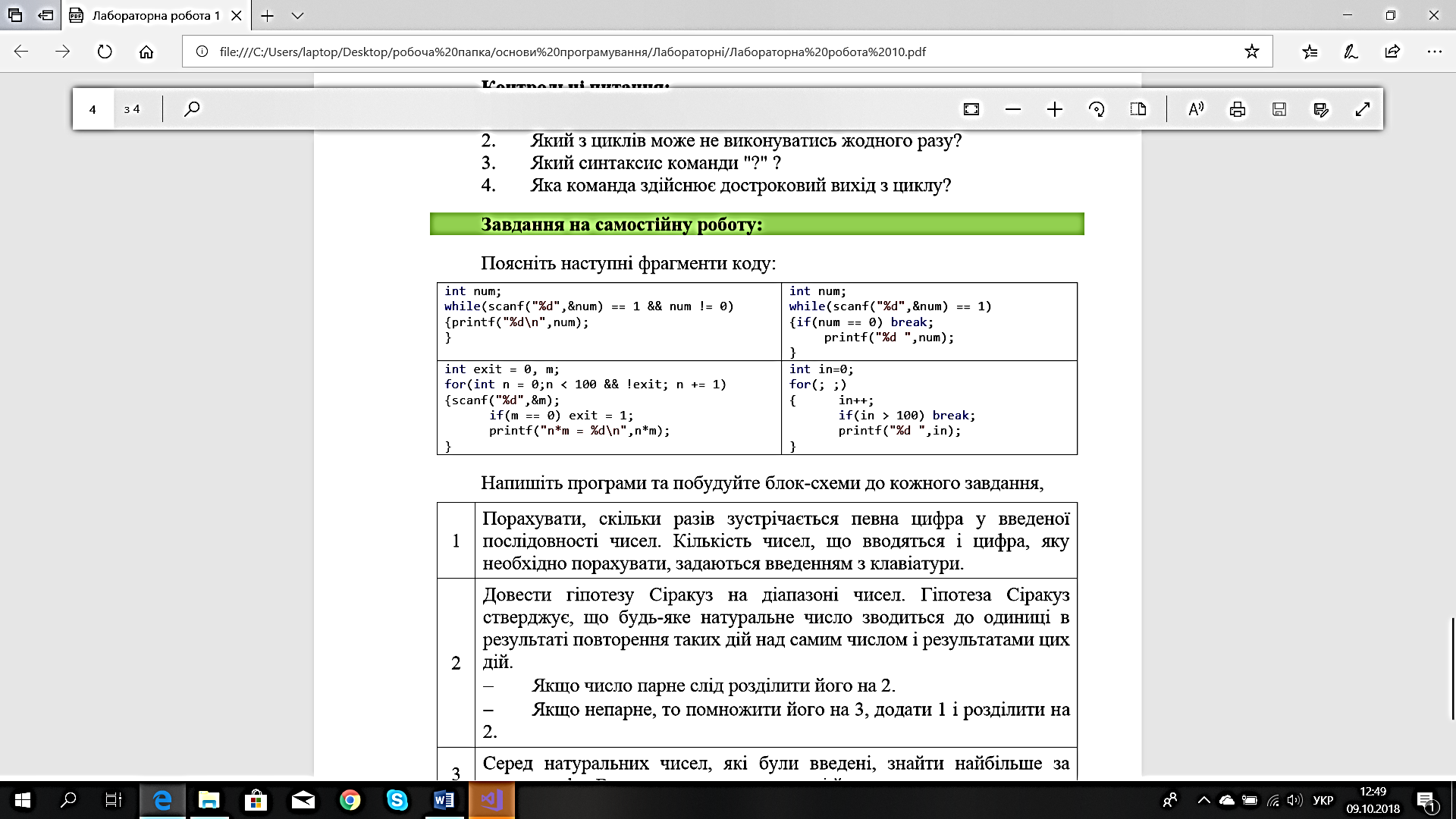
printf("Сначала мы присваиваем переменной num значение и выводим его на екран,\n если переменная равна 0, то происходит завершение программы\n");

return 0;

}

Результат виконання програми:





Умова до завдання на самостійну роботу (3)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <math.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int exit = 0, m; //вводим переменные: exit, m

for (int n = 0; n < 100 && !exit; n += 1)

/\*вводиться переменная n которой присвоено значение 0,

проверяеться условие что n < 100 и не равно exit,

n каждый раз увеличиваеться на 1\*/

{ scanf\_s("%d", &m);

if (m == 0) exit = 1;

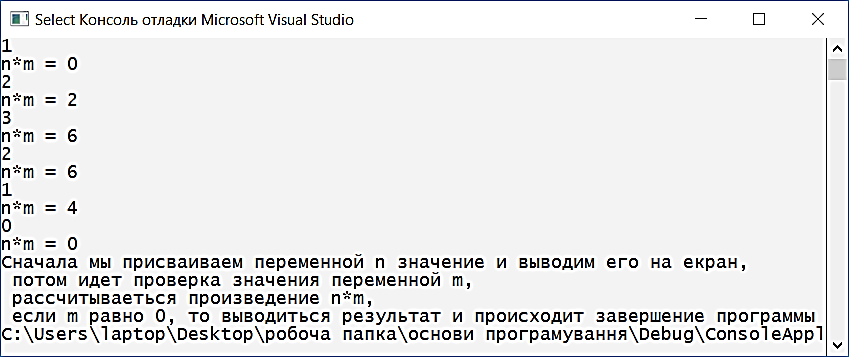
printf("n\*m = %d\n", n\*m);

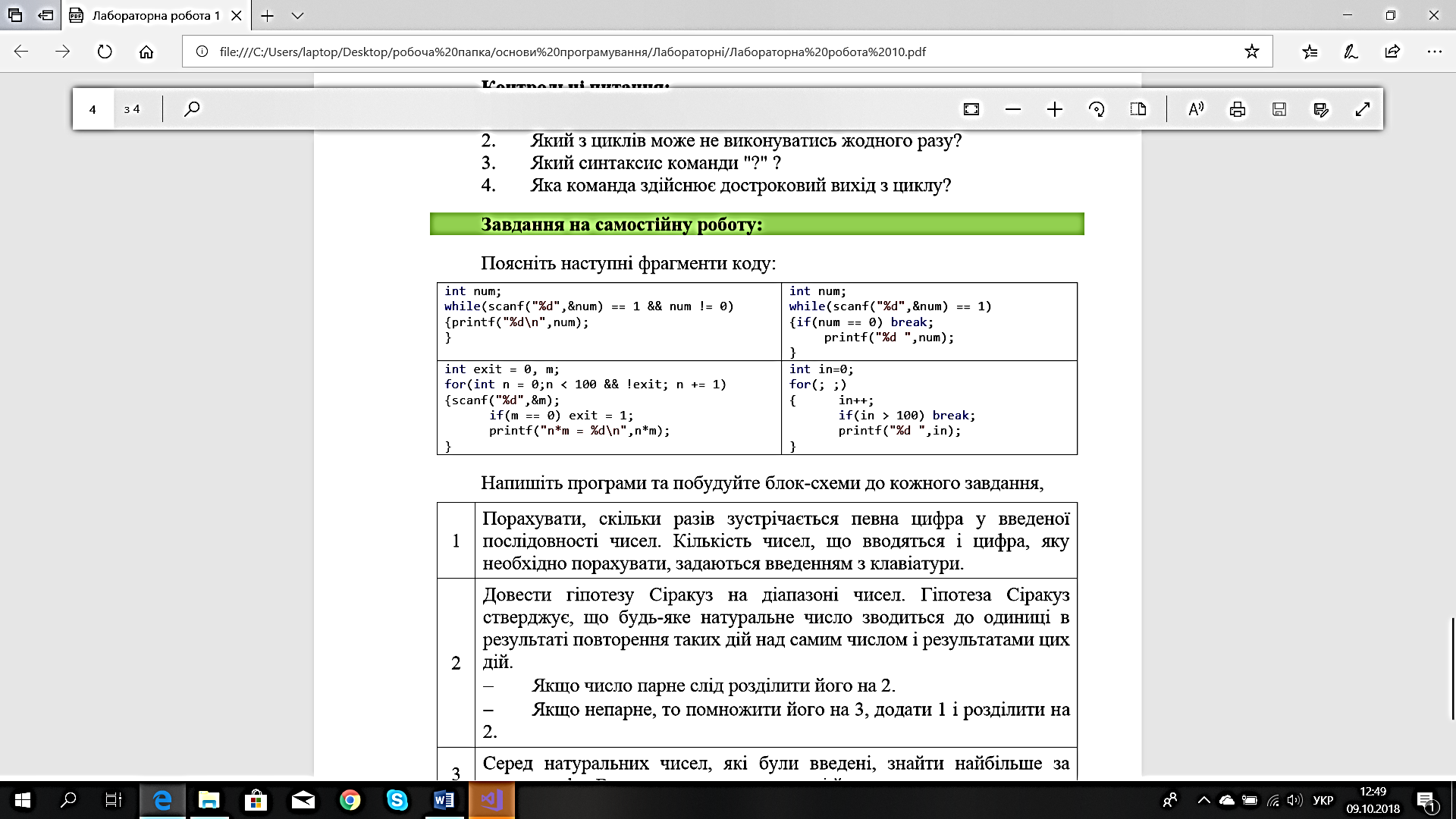
}

printf("Сначала мы присваиваем переменной n значение и выводим его на екран,\n потом идет проверка значения переменной m,\n рассчитываеться произведение n\*m,\n если m равно 0, то выводиться результат и происходит завершение программы ");

}

Результат виконання програми:





Умова до завдання на самостійну роботу (4)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <math.h>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int in = 0; //вводим переменную in и присваиваем значение 0

for (; ;)

{ in++;

if (in > 100) break;

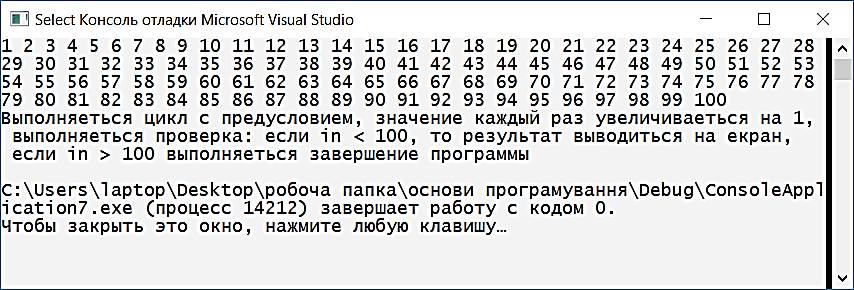
printf("%d ", in);

}

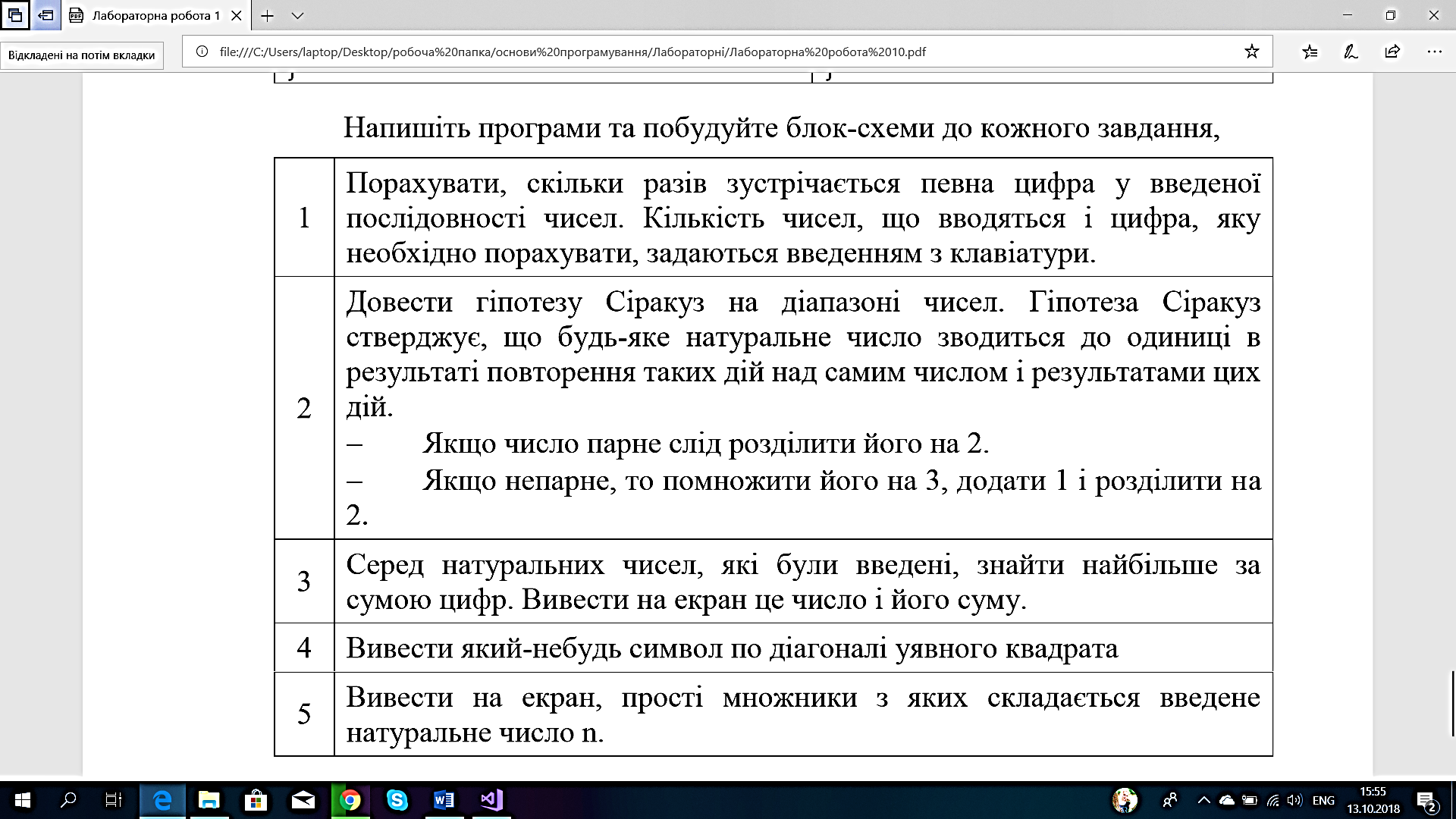
printf("\nВыполняеться цикл с предусловием, значение каждый раз увеличиваеться на 1,\n выполняеться проверка: если in < 100, то результат выводиться на екран,\n если in > 100 выполняеться завершение программы\n");

}

Результат виконання програми:



1. Напишіть програми та побудуйте блок-схеми до кожного завдання.



Умова до завдання на самостійну роботу (1)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int number, count, i, numeral, firstNum, secondNum, counter = 0;

printf("Выберите какое число будем искать (0-9):\n");

scanf\_s("%d", &numeral);

printf("Сколько числ будете вводить:\n");

scanf\_s("%d", &count);

printf("Введите двухзначные числа:\n");

for (i = 0; i < count; i++) {

scanf\_s("%d", &number);

firstNum = number / 10;

secondNum = number - (firstNum \* 10);

if (numeral == firstNum) {

counter++;

}

if (numeral == secondNum) {

counter++;

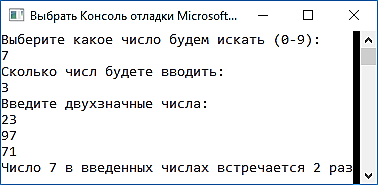
}

}

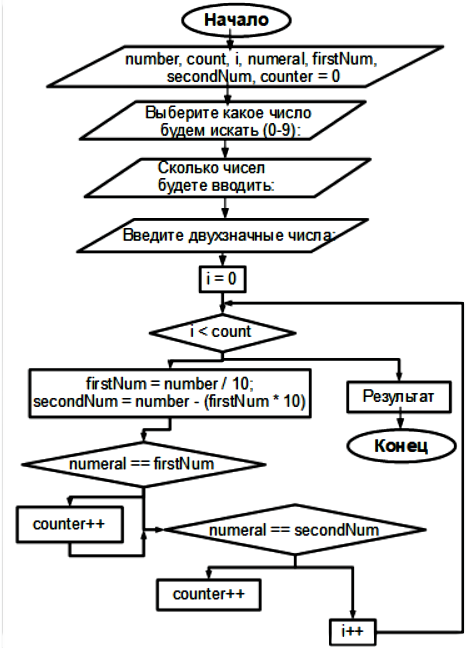
printf("Число %d в введенных числах встречается %d раз", numeral, counter);

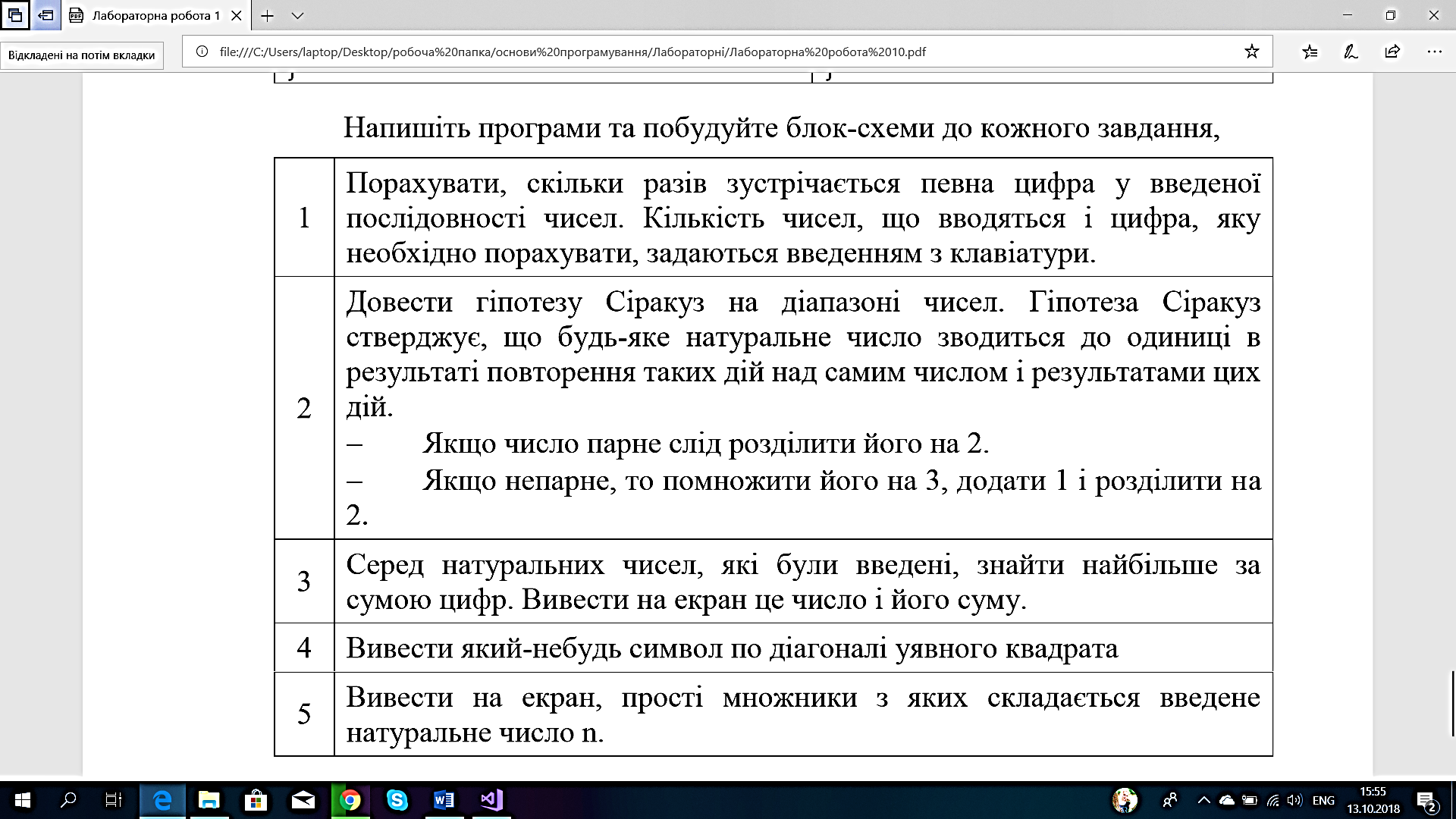
}

Результат виконання программи:



Блок-схема





Умова до завдання на самостійну роботу (2)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <Windows.h>

int main()

{ SetConsoleCP(1251);

SetConsoleOutputCP(1251);

long num, countDown, totalDividers = 0, k = 0;

long allResults[100]{ 0 };

printf(""), scanf\_s("%d", &num);

countDown = num;

while (countDown > 0)

{

totalDividers = 0;

for (long i = 1; i < countDown; i++) {

if ((countDown % i) == 0)

totalDividers += i;

}

if (totalDividers == countDown) {

k++;

allResults[k] = countDown;

}

countDown--;

}

printf("\tДоскональні числа:\n");

while (k > 0)

{

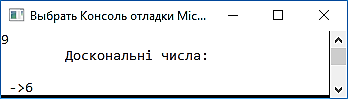
printf("\n ->%d", allResults[k]);

k--;

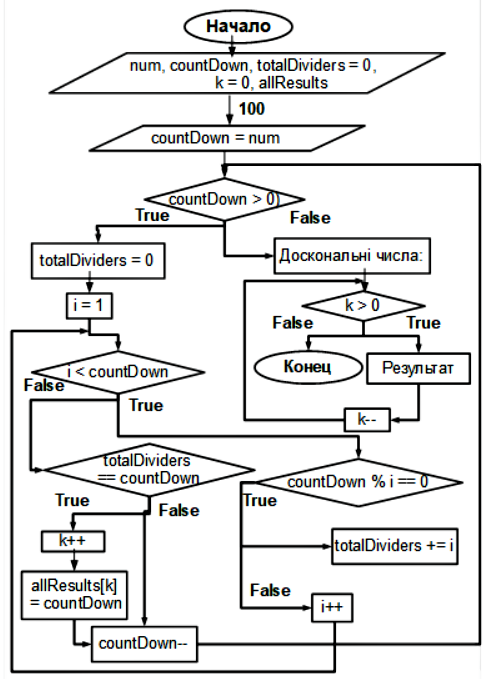
}

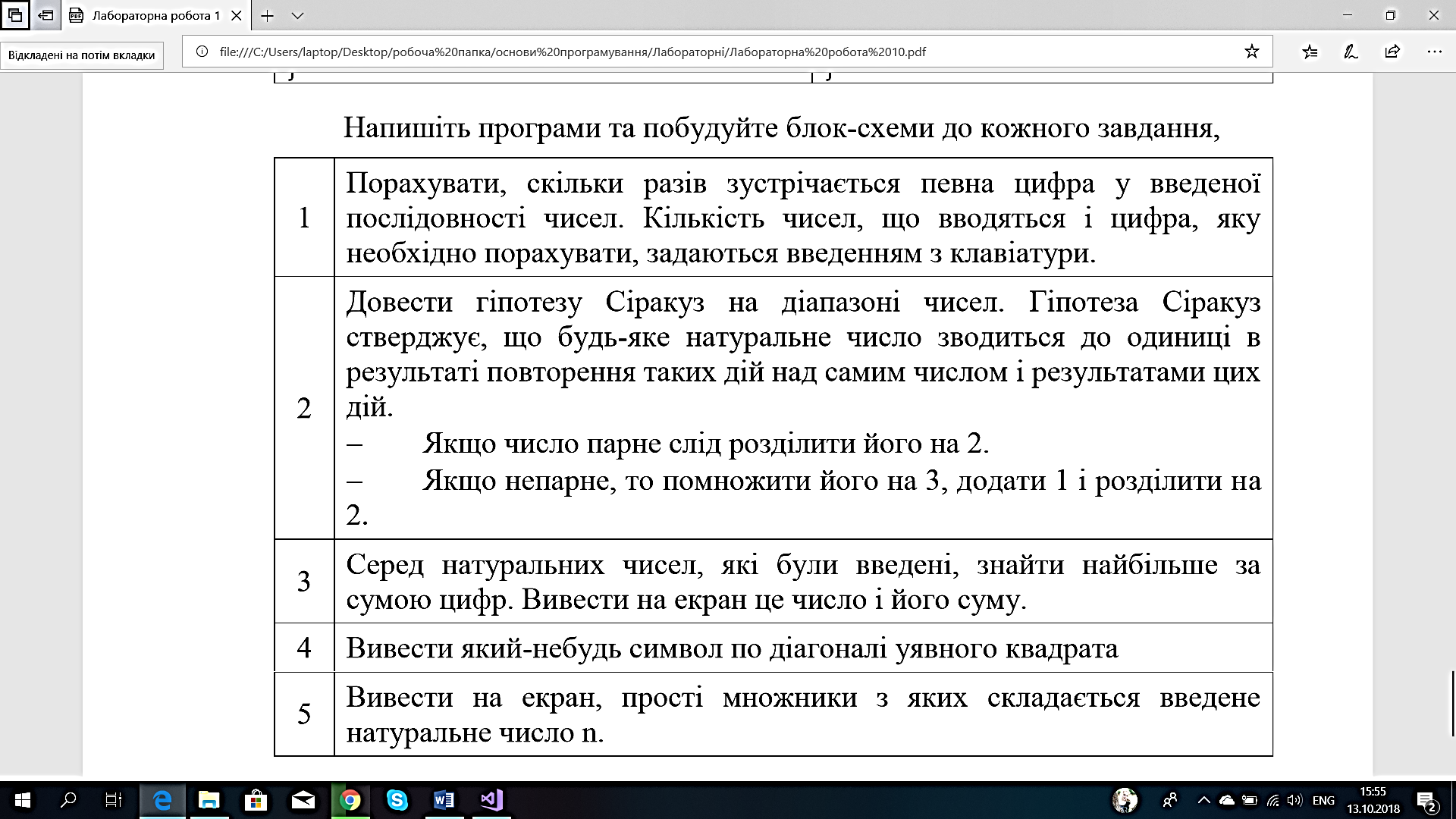
}

Результат виконання программи:



Блок-схема





Умова до завдання на самостійну роботу (3)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

int number, firstNum, secondNum, numSum = 0, numSumTheBiggest = 0;

printf("Для завершения ввода числ, введите 0\n\n");

do {

printf("Введите 2-ух значное число:\n");

scanf\_s("%d", &number);

if (number < 10 && number > 99) {

printf("Вы должны были ввести 2-ух значное число!");

continue;

}

firstNum = number / 10;

secondNum = number - (firstNum \* 10);

numSum = firstNum + secondNum;

if (numSum > numSumTheBiggest) {

numSumTheBiggest = numSum;

}

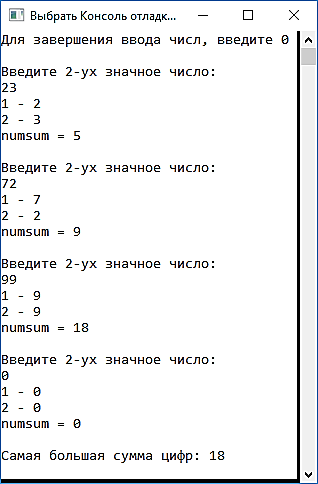
printf("1 - %d\n2 - %d\nnumsum = %d\n\n", firstNum, secondNum, numSum);

} while (number != 0);

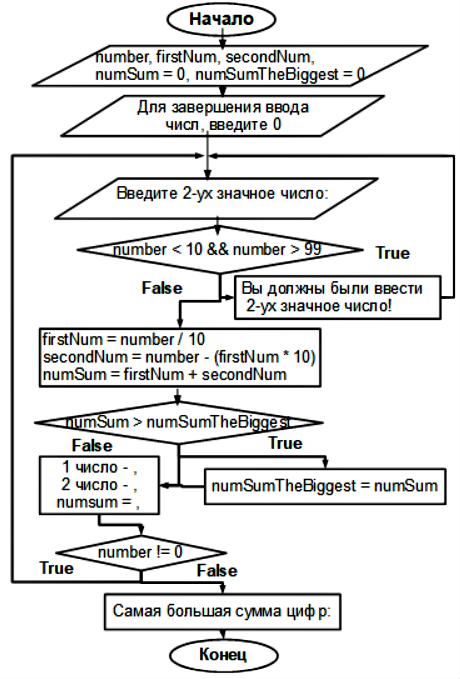
printf("Самая большая сумма цифр: %d\n", numSumTheBiggest);

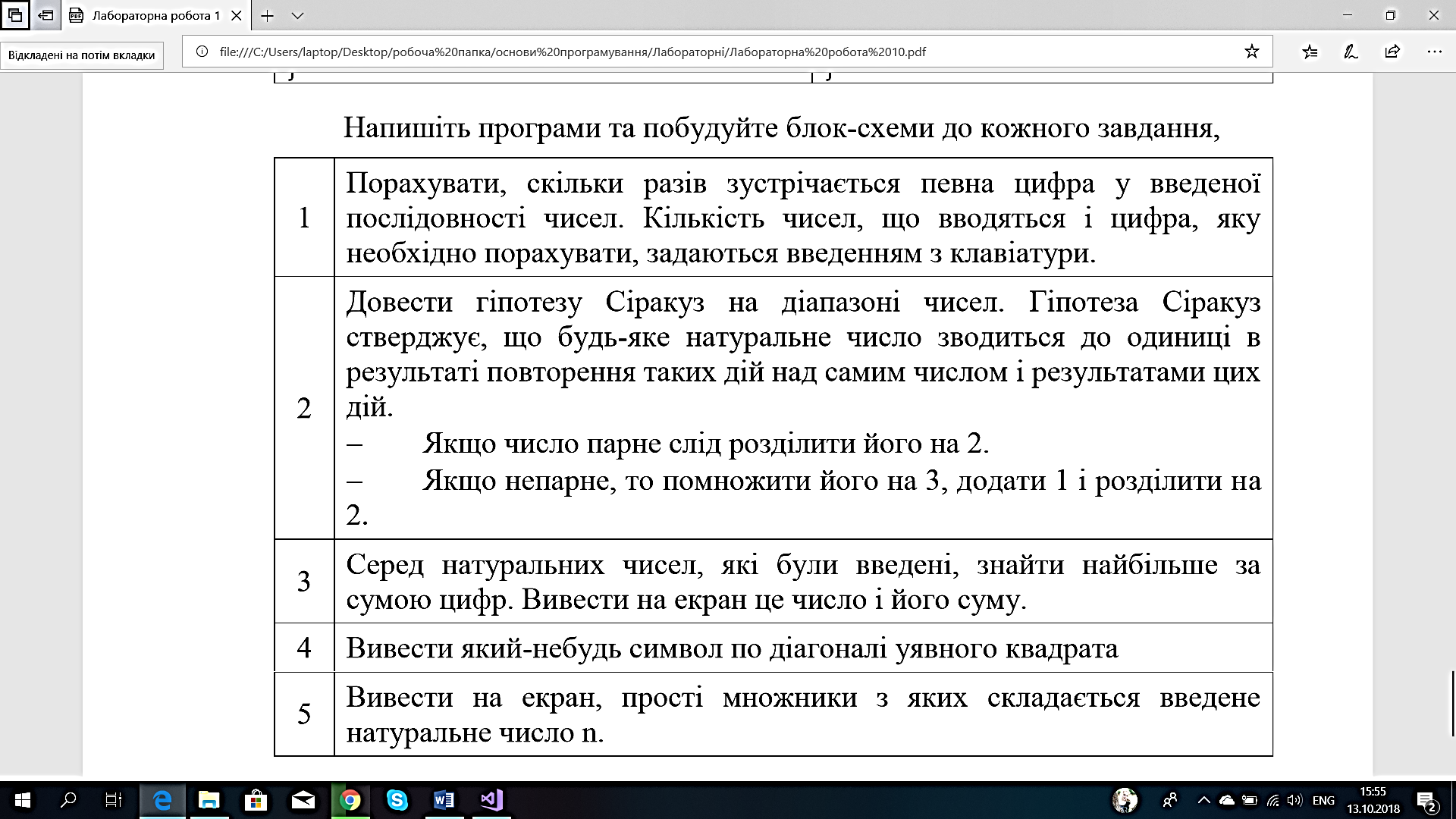
}

Результат виконання программи:



Блок-схема





Умова до завдання на самостійну роботу (4)

Лістинг программи:

#include "pch.h"

#include <iostream>

int main()

{

int repeat;

printf("Enter repeat quantity: "), scanf\_s("%d", &repeat);

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

{ for (int k = 0; k <= i; k++)

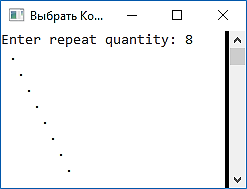
printf(" ");

printf(".\n");

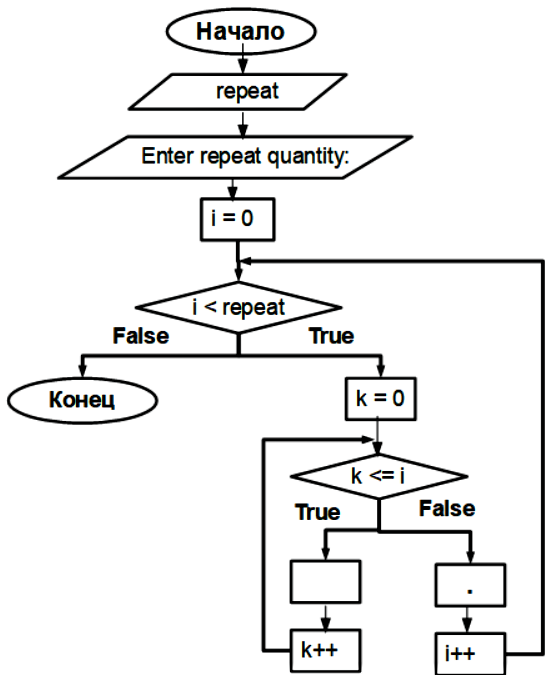
}

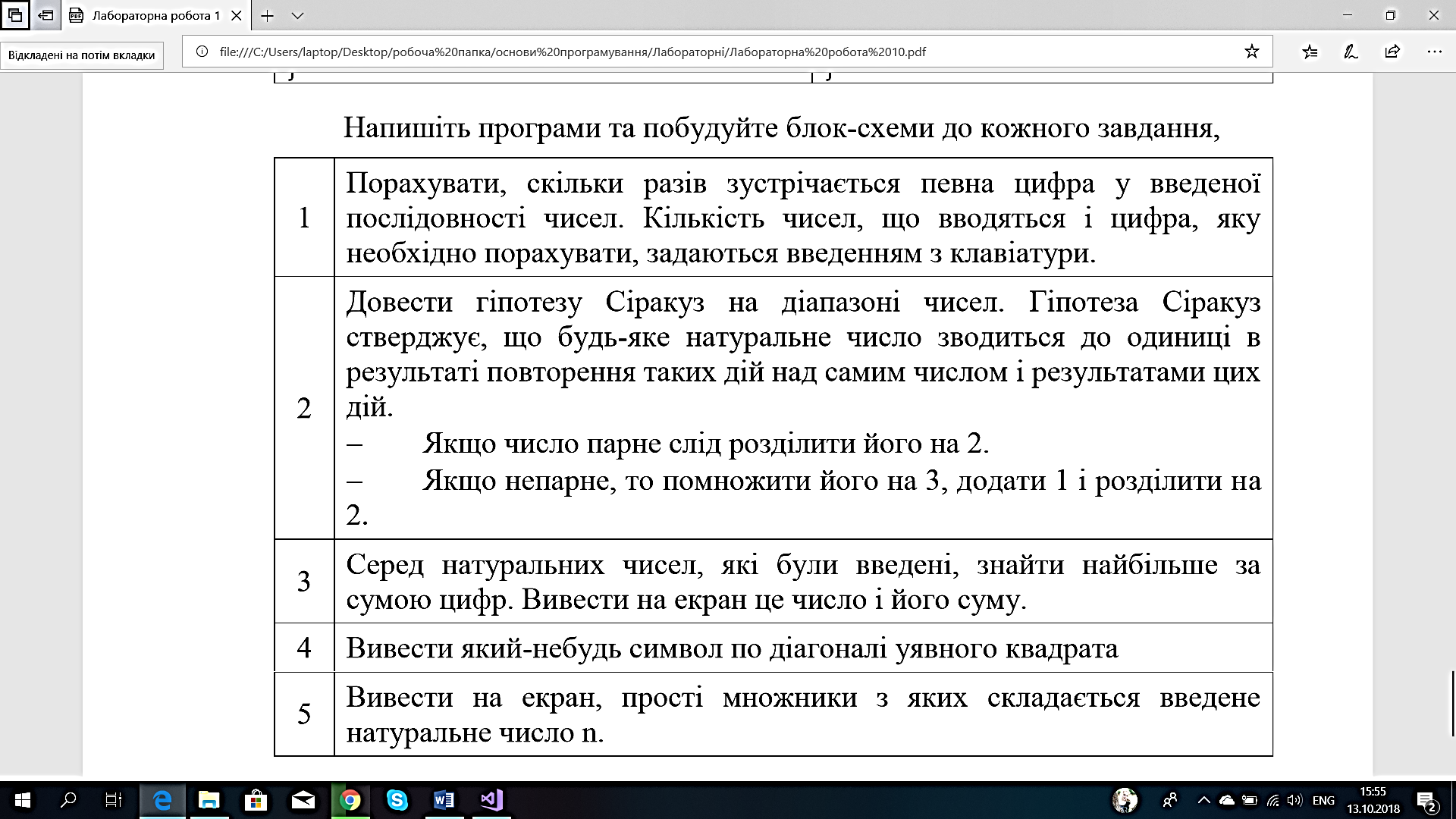
}

Результат виконання программи:



Блок-схема





Умова до завдання на самостійну роботу (5)

Лістинг программи:

#include "pch.h"

#include <iostream>

#include <locale.h>

int main()

{

setlocale(LC\_ALL, "Rus");

float number;

int i;

printf("Введите число\n");

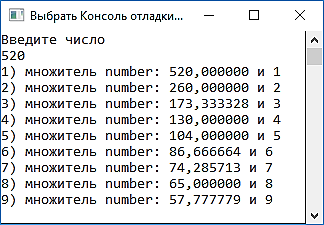
scanf\_s("%f", &number);

for (i = 1; i < 10; i++)

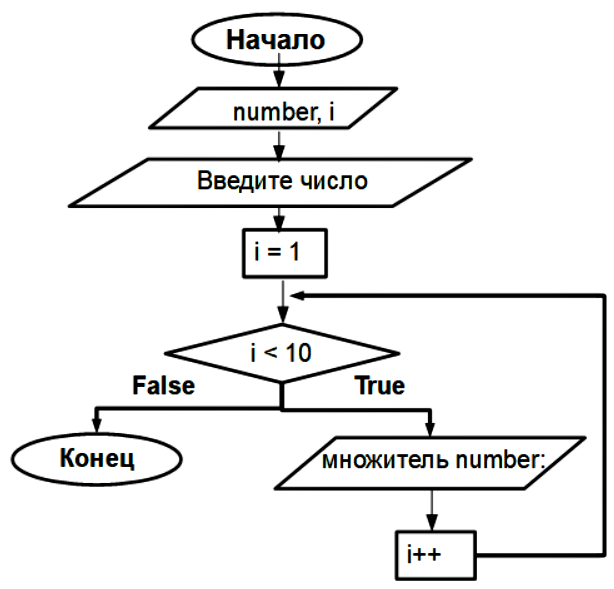
printf("%d) множитель number: %f и %d\n", i, number / i, i);

}

Результат виконання программи:



Блок-схема



**Контрольні питання:**

1. *Що таке цикли, які бувають цикли?*

**Цикл** – багаторазове повторення виконання операторів. Цикл повинен мати умову продовження (завершення) циклу. Інакше відбудеться зациключання (зависання програми).

В мові С є три оператори циклів:

1) оператор while

2) оператор for

3) оператор do … while

цикли з передумовою

цикл з постумовою

1. *Який з циклів може не виконуватись жодного разу?*

Цикли з передумовою можуть не виконуватись жодного разу.

Ту частину, яку потрібно виконувати, називають тілом циклу. Працює ця

конструкція таким чином. Спочатку обчислюється значення "логічного виразу". Якщо воно дорівнює TRUE (істина), виконується тіло циклу після чого виконується повернення на початок циклу і все повторюється знову починаючи з обчислення "логічного виразу". Цикл виконується доти, поки значенням "логічного виразу" не стане FALSE (хибність). Після цього продовжують виконуватись оператори, що записані після циклу (відразу за його тілом). Якщо при початковій перевірці значенням "логічного виразу" відразу стало FALSE, то тіло циклу не виконується жодного разу.

1. *Який синтаксис команди «?»?*

Виконує ту ж функцію, що й оператор if, якщо умова істинна.

1. *Яка команда здійснює достроковий вихід з циклу?*

Достроковий вихід з циклу здійснює оператор break. Тобто здійснює вихід з циклу незалежно від істинності умови виходу.

***Висновок:*** в ході виконання лабораторної роботи вивчено особливості використання операторів циклу.