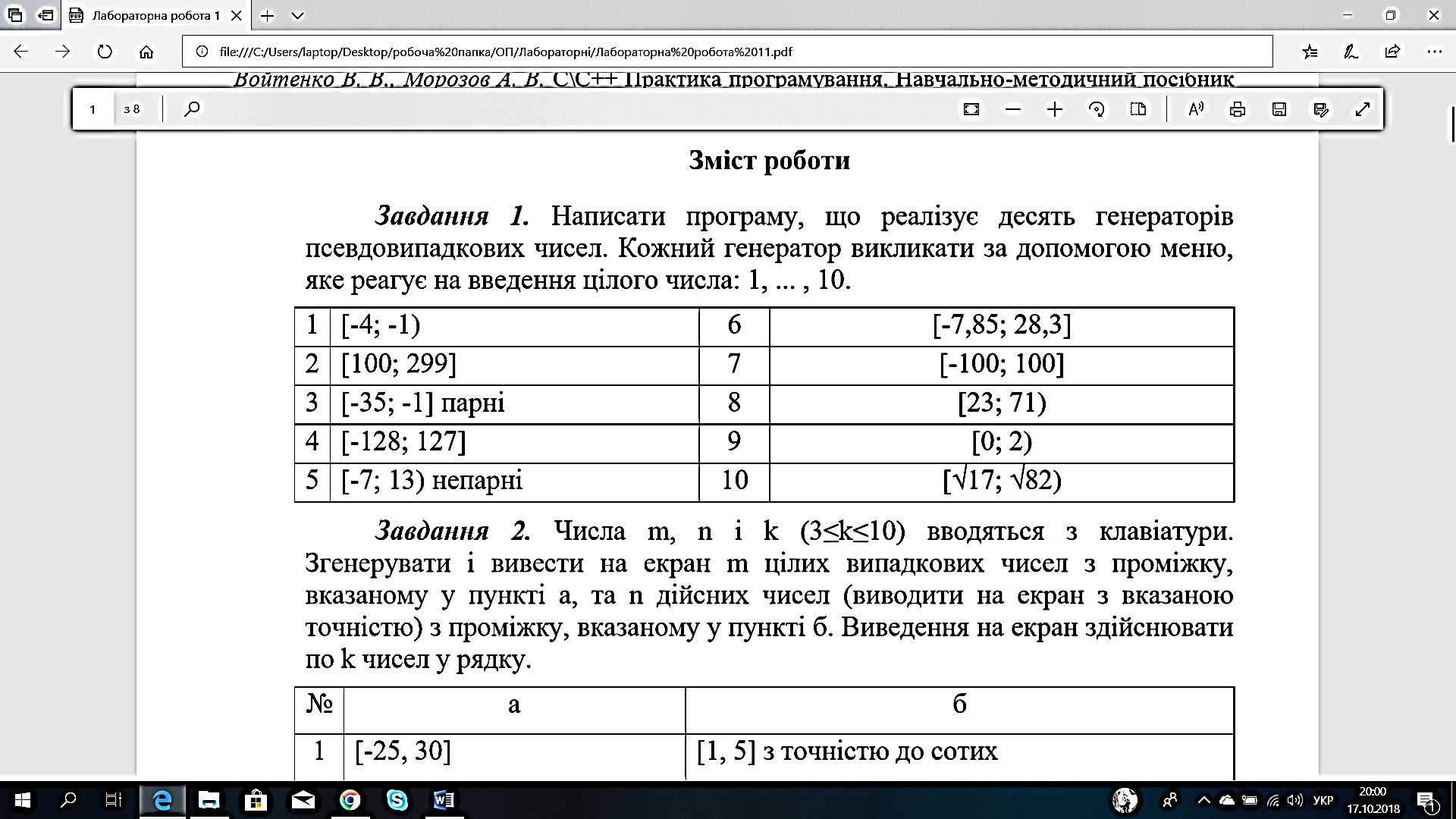
**ЛАБОРАТОРНА РОБОТА № 11**

**Одновимірні масиви**

***Мета:*** дослідити функції генерування послідовностей випадкових чисел, ознайомитись з можливостями створення одновимірних масивів і навчитись обробляти елементи масивів.

**Хід роботи:**

**Завдання 1**: Написати програму, що реалізує десять генераторів псевдовипадкових чисел. Кожний генератор викликати за допомогою меню, яке реагує на введення цілого числа: 1, ... , 10.



Умова до завдання № 1

#include "pch.h"

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <locale.h>

#include <math.h>

int main()

{

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int choise, var, res, i;

printf("Введите номер :");

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

printf("Введите количество повторений: ");

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

switch (choise)

{

case 1:

{ printf("{");

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

{

res = rand() % 3 - 4;

printf("%d, ", res);

}

printf("}");

break; }

case 2:

{ printf("{");

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

{

res = 100 + rand() % (299 - 100 + 1);

printf("%d, ", res);

}

printf("}");

break; }

case 3:

{ printf("{");

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

{

res = rand() % 35 - 36;

if (res % 2 == 0)

printf("%d, ", res);

}

printf("}");

break; }

case 4:

{ printf("{");

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

{

res = -128 + rand() % (127 + 128 + 1);

printf("%d, ", res);

}

printf("}");

break; }

case 5:

{

printf("{");

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

{

res = -7 + rand() % (13 + 7);

if (res % 2 != 0)

printf("%d, ", res);

}

printf("}");

break; }

case 6:

{

printf("{");

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

{

double sum = rand() % 34 - 7.85;

printf("%d, ", sum);

}

printf("}");

break;

}

case 7:

{

printf("{");

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

{

res = -100 + rand() % (100 + 100 + 1);

printf("%d, ", res);

}

printf("}");

break; }

case 8:

{

printf("{");

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

{

res = 23 + rand() % (71 - 23);

printf("%d, ", res);

}

printf("}");

break; }

case 9:

{

printf("{");

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

{

res = rand() % 2;

printf("%d, ", res);

}

printf("}");

break; }

case 10:

{

int a, b;

a = sqrt(17) \* 10000000;

b = sqrt(82) \* 10000000;

printf("{");

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

{

float sum;

sum = a + rand() % (b - a);

printf("%d, ", sum/10000000);

}

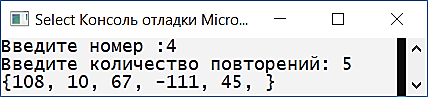
printf("}");

break; }

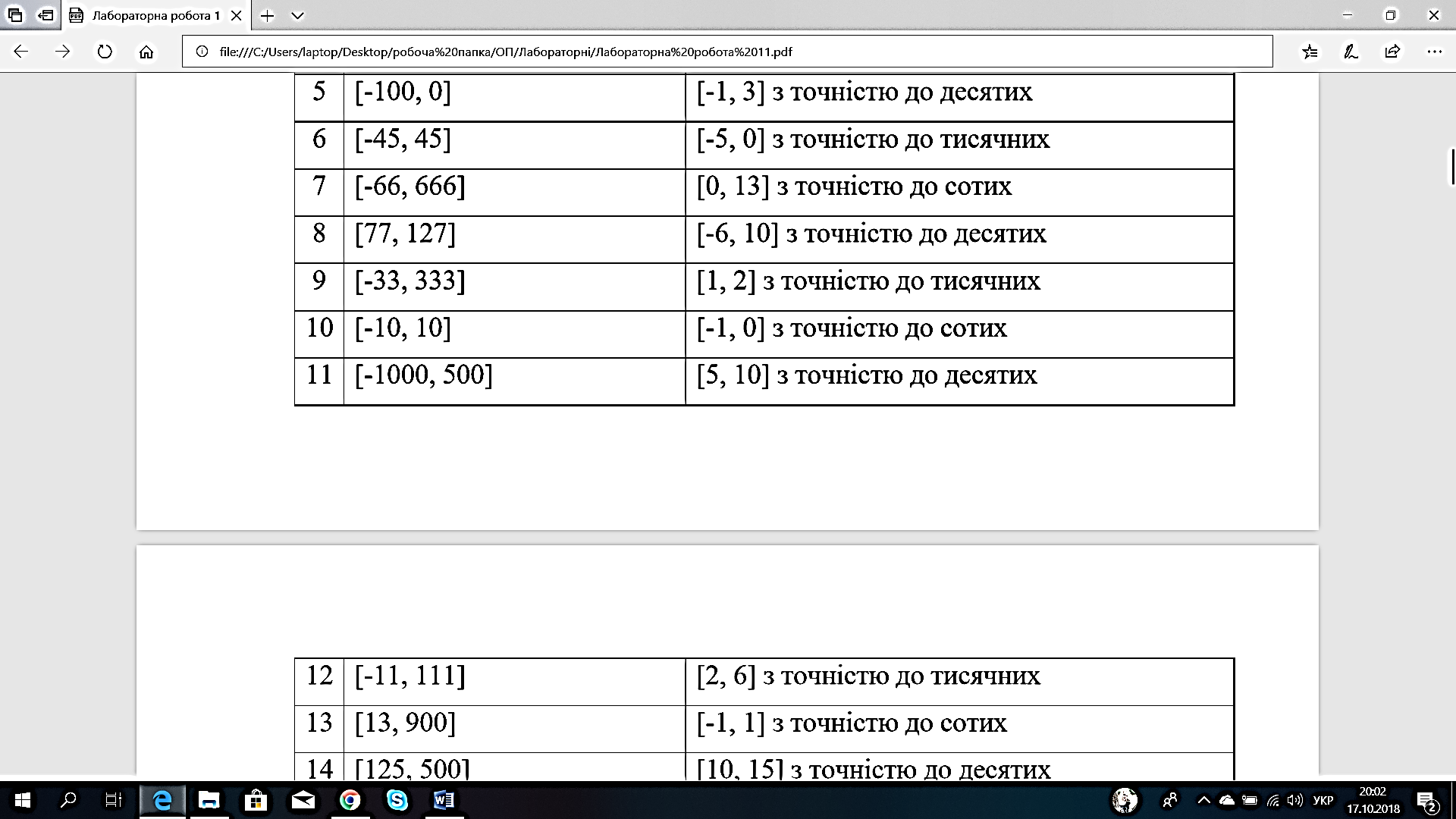
}

}

Результат виконання програми:



**Завдання 2:** Числа m, n і k (3≤k≤10) вводяться з клавіатури. Згенерувати і вивести на екран m цілих випадкових чисел з проміжку, вказаному у пункті а, та n дійсних чисел (виводити на екран з вказаною точністю) з проміжку, вказаному у пункті б. Виведення на екран здійснювати по k чисел у рядку.



Лістинг програми:

#include "pch.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

int main()

{

srand(time(0));

float temp;

int m, n, k;

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

printf("\nEnter n: "), scanf\_s("%d", &n);

printf("\nEnter k [3; 10]: "), scanf\_s("%d", &k);

if (k >= 3 && k <= 10) {

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

{

i % k == 0 ? printf("\n") : 0;

printf("%d \n", -10 + rand() % (10 + 10 + 1));

}

for (int i = 1; i <= n; i++)

{

i % k == 0 ? printf("\n") : 0;

temp = -1 + rand() % (0 + 1 + 1);

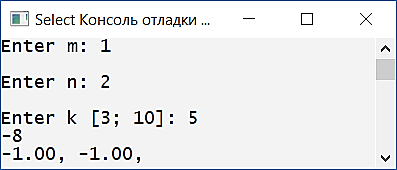
printf("%.2f, ", temp);

}

}

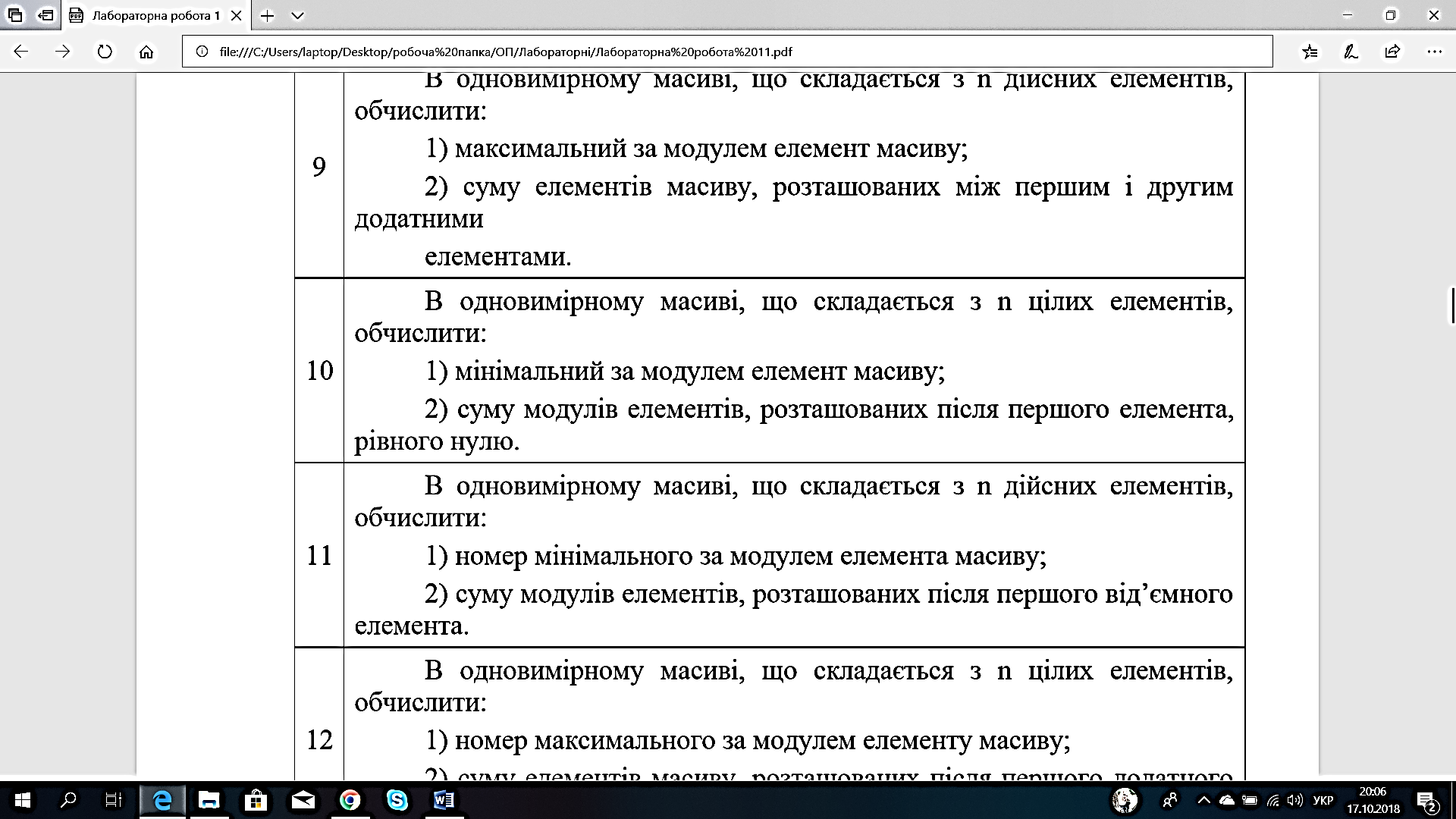
else printf("\n\nError");}

Результат виконання програми:



Результат виконання програми до завдання №2

**Завдання 3:** Розробити програму, дотримуючись таких вимог: −використовувати статичні масиви; −число n (кількість елементів масиву); −елементи масиву - випадкові числа, згенеровані на інтервалі [a, b], де a і b вводяться з клавіатури (a<b); −усі вхідні дані а також елементи масиву виводяться на екран.



Умова до завдання №3

Лістинг програми:

#include "pch.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

#include <math.h>

int main()

{

srand(time(0));

int arr[20], a, b, min = 1e7, sum = 0;

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

printf("\n Enter b: "), scanf\_s("%d", &b);

a > b ? printf("error") : 0;

for (int i = 0; i != 20; i++)

arr[i] = a + rand() % (b - a + 1), printf("\n%d", arr[i]);

for (int i = 0; i != 20; i++) {

arr[i] = abs(arr[i]);

min > arr[i] ? min = arr[i] : 0;

}

for (int i = 0; i != 20; i++)

{

if (arr[i] != 0)

{

arr[i] = abs(arr[i]);

sum += arr[i];

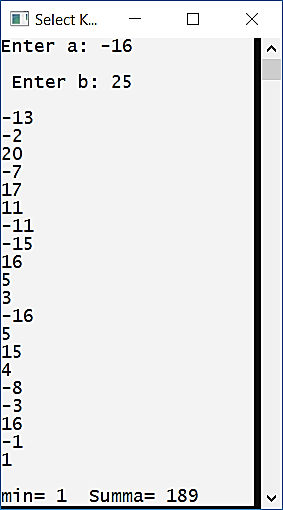
}

}

printf("\n\nmin= %d\tSumma= %d", min, sum);

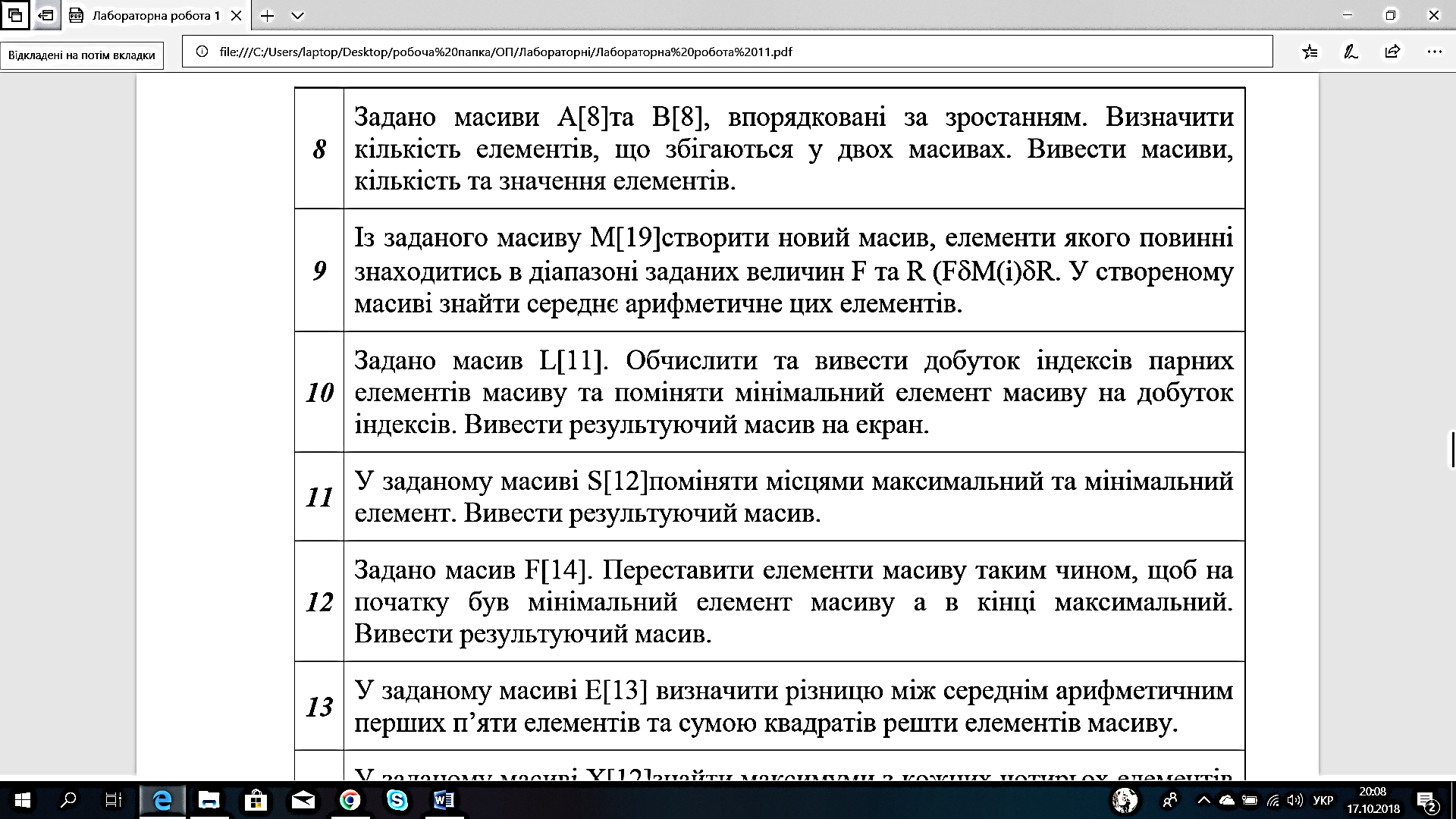
}

Результат виконання програми:



Результат виконання програми до завдання №3

**Завдання 4:** Написати програму:



Умова до завдання №4

#include "stdafx.h"

#include <iostream>

#include <stdlib.h>

#include <time.h>

#define N 11

int main()

{

srand(time\_t(NULL));

int L[N], i, dob = 1, min = 1e7, imin;

printf("{");

for(i = 0; i <N; i++)

L[i] = rand() % 11;

for (i = 0; i < N; i++)

{

if (L[i] % 2 == 0)

dob \*= i;

if (min > L[i])

{

min = L[i];

imin = i;

}

}

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

printf(" %d; ", L[i]);

}

printf("\n");

L[imin] = dob;

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

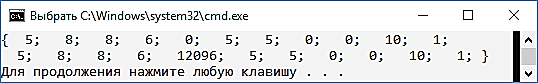
printf(" %d; ", L[i]);

}

printf("}\n");

}

Результат виконання програми:



Результат виконання програми до завдання №4

***Висновок:*** в ході виконання лабораторної роботи досліджено функції генерування послідовностей випадкових чисел, ознайомлено з можливостями створення одновимірних масивів і навчено обробляти елементи масивів.