Московский технический университет связи и информатики

Кафедра «Кибернетика и информационная безопасность»

Лабораторная работа по дисциплине «Теория и методы программирования» № 1

Тема: «*РАЗРАБОТКА АЛГОРИТМОВ ПСЕВДОСЛУЧАЙНЫХ ПОСЛЕДОВАТЕЛЬНОСТЕЙ»*

Выполнил:

Блёсткин Михаил

Группа БИБ2203

Вариант 6

Москва 2023 г.

**Цель:**

Изучить методику тестирования генераторов случайных чисел (ГСЧ) и генераторов псевдослучайных чисел (ГПСЧ).

**Постановка задачи:**

1. Случайное выбранное число сохраняется в переменную. Тип переменной long int.
2. Получить с помощью генератора случайных чисел 200 чисел в интервале от 0 до 1. Оценить по ним равномерность распределения. Разбить интервал (0,1) на интервалов равной длины [0..0,1), [0,1..0,2), …,[0,9..1] и построить горизонтальную гистограмму (столбчатую диаграмму), показывающую, сколько чисел попало в каждый интервал.

Пример гистограммы:

0.0..0.1 \*\*\*\*\*\*\*

0.1..0.2 \*\*\*\*\*\*\*\*\*

…

0.9..1.0 \*\*\*\*\*\*\*

2.1 Получить с помощью генератора случайных чисел:

а) 25 действительных чисел, лежащих в диапазоне от –50 до 50;

б) 30 целых чисел, лежащих в диапазоне от –20 до 20;

в) 20 неотрицательных действительных чисел, не превосходящих 40;

г) 35 неотрицательных целых чисел, не превосходящих 1000;

д) 27 натуральных чисел, не превосходящих 20;

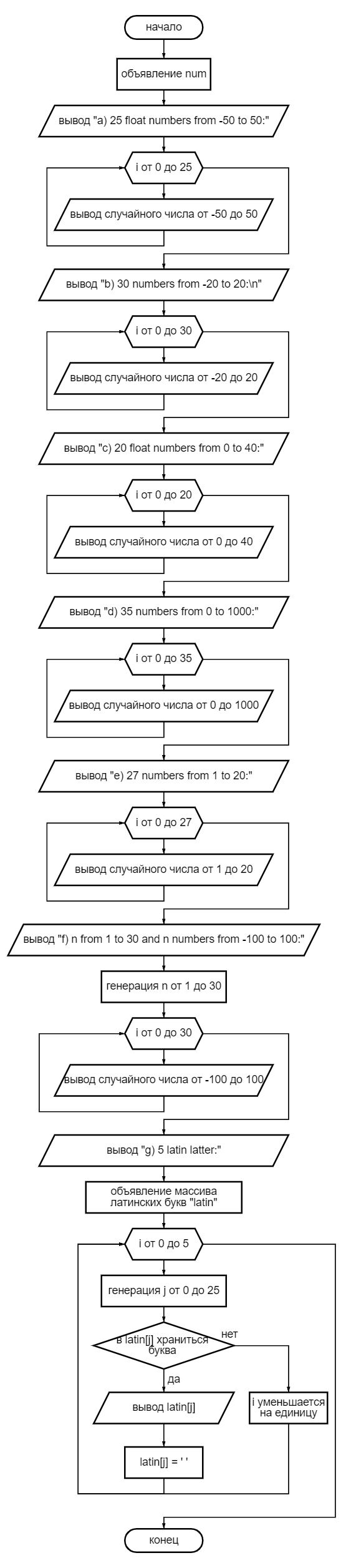
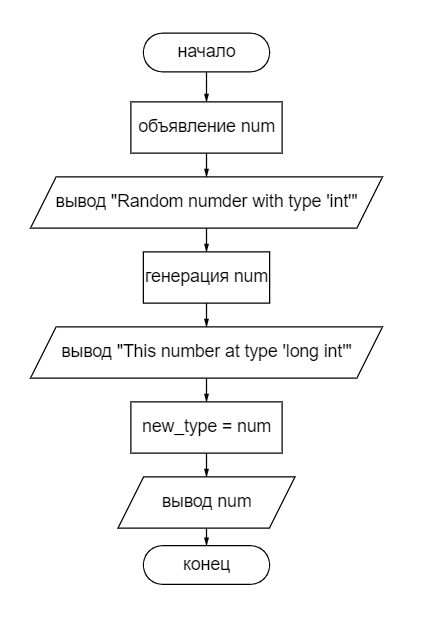
е) натуральное *n*, не превосходящее 30, и *n* действительных чисел, лежащих в диапазоне от –100 до 100;

ж) 5 неповторяющихся латинских букв.

1. Составить программу для игры в “кости”. Играющий называет любое число в диапазоне от 2 до 12 и ставку, которую он делает в этот ход. Программа с помощью генератора случайных чисел дважды выбирает числа от 1 до 6 (“бросает кубик”, на гранях которого цифры от 1 до 6). Если сумма выпавших очков меньше либо равна 7 и играющий назвал число меньше либо равное 7, он выигрывает ставку. Если сумма выпавших цифр больше 7 и играющий сделал ставку на число больше 7, он также выигрывает ставку. Если игрок угадал сумму цифр, он получает в 4 раза больше очков, чем сделанная ставка. Ставка проиграна, если ни одна из описанных ситуаций не имеет места. В начальный момент у игрока и компьютера по 100 очков. Игра идет до тех пор, пока у кого-либо из играющих останется 0 очков.

**Алгоритм решения задачи:**

Задание 1 Задание 2.1

Задание 1:

* Генерируется целое случайное число через функцию rand().
* Сгенерированное число записывается в переменную с типом long int.
* Результат выводится на экран.

Задание 2:

* Объявляется функция random(), которая генерирует действительное число в интервале [0,1].
* В основной функции создаётся массив целых чисел intervals с размером 10 и заполняется нулями.
* Выполняется цикл 200 раз, на каждом этапе:
  + Генерируется число через функцию random(),
  + Если сгенерированное число попадает в интервал от [i, i+1], где i принимает целое значение от 0 до 9, то i-й элемент массива intervals увеличивает своё значение на единицу.
  + Сгенерированное число выводится на экран.
* Через цикл выполняется вывод гистограммы, на каждом шаге на экран слитно выводится количество знаков «\*», соответствующие размеру элемента массива.

Задание 2.1:

* В цикле, который длится n шагов, в зависимости от задания, происходит генерация чисел в заданном интервале по функциям rand() и random(), и последующий вывод сгенерированного числа на экран.
* Аналогично выполняется 6 пунктов задания.
* В последнем седьмом задании создаётся массив латинских заглавных букв.
* В цикле размером 5 генерируется целое число в интервале от 0 до 25.
* На экран выводится буква латинского алфавита, соответствующая сгенерированному номеру массива.
* Буква в массиве заменяется на нейтральный символ и больше не может быть выведена на экран.

Задание 3:

* Выполняется цикл с условием выхода: пока количество очков игрока и компьютера больше нуля. Количество очков определяется переменными целого типа, изначально равными по 100.
* Пользователь вводит ставку и число.
* С помощью функции rand() генерируется два числа от 1 до 6.
* Если сумма сгенерированных чисел больше 7 и игрок ввёл число больше 7, то к переменной его очков прибавляется сумма, равная ставке, а из переменной компьютера это число вычитается.
* Если сумма сгенерированных чисел не больше 7 и игрок ввёл число не больше 7, то к переменной его очков прибавляется сумма, равная ставке, а из переменной компьютера это число вычитается.
* Если сумма сгенерированных чисел равна числу, которое ввёл пользователь, то к переменной его очков прибавляется сумма, равная тройной ставке, а из переменной компьютера это число вычитается.
* Если ни одно из условий не выполняется, то к переменной очков компьютера прибавляется сумма, равная ставке, а из переменной игрока это число вычитается.
* После выхода из цикла на экран выводится сообщение о победе игрока или компьютера в зависимости от того, у какой переменной положительное значение соответственно.

**Листинг программы**

Задача 1

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

void main()

{

time\_t t;

srand((unsigned)time(&t));

printf("Random number with type of 'int':\n");

int num = rand();

printf("%d\nThis number at 'long int' type:\n", num);

long int new\_type = num;

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

}

Задача 2

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

#include <random>

#include <chrono>

double random()

{

return (double)rand() / RAND\_MAX;

}

void main()

{

time\_t t;

srand((unsigned)time(&t));

int intervals[10];

for (int i = 0; i < 10; i++)

intervals[i] = 0;

printf("200 random numbers from 0 to 1:\n\n");

for (int i = 0; i < 200; i++)

{

double num = random();

if (num <= 0.1) intervals[0]++;

if (num > 0.1 && num <= 0.2) intervals[1]++;

if (num > 0.2 && num <= 0.3) intervals[2]++;

if (num > 0.3 && num <= 0.4) intervals[3]++;

if (num > 0.4 && num <= 0.5) intervals[4]++;

if (num > 0.5 && num <= 0.6) intervals[5]++;

if (num > 0.6 && num <= 0.7) intervals[6]++;

if (num > 0.7 && num <= 0.8) intervals[7]++;

if (num > 0.8 && num <= 0.9) intervals[8]++;

if (num > 0.9 && num <= 1) intervals[9]++;

printf("\t%f\n", num);

}

printf("\n\nThe diagram:\n");

for (int i = 0; i < 10; i++)

{

printf("0.%d..%d.%d ", i, (i + 1) / 10, (i + 1) % 10);

for (int j = 0; j < intervals[i]; j++)

printf("\*");

printf("\n");

}

}

Задача 2.1

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

double random()

{

return (double)rand() / RAND\_MAX;

}

void main()

{

time\_t t;

srand((unsigned)time(&t));

printf("\na) 25 float numbers from -50 to 50:\n");

for (int i = 0; i < 25; i++)

printf("\t%d: %f\n", i + 1, random() \* 100 - 50);

printf("\nb) 30 numbers from -20 to 20:\n");

for (int i = 0; i < 30; i++)

printf("\t%d: %d\n", i + 1, (int)(random() \* 40) - 20);

printf("\nc) 20 float numbers from 0 to 40:\n");

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

printf("\t%d: %f\n", i + 1, random() \* 40);

printf("\nd) 35 numbers from 0 to 1000:\n");

for (int i = 0; i < 35; i++)

printf("\t%d: %d\n", i + 1, (int)(random() \* 1000));

printf("\ne) 27 numbers from 1 to 20:\n");

for (int i = 0; i < 27; i++)

printf("\t%d: %d\n", i + 1, (int)(random() \* 20) + 1);

printf("\nf) n from 1 to 30 and n numbers from -100 to 100:\n");

int n = (int)(random() \* 30) + 1);

printf("\tn = %d\n", n);

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

printf("\t%d: %d\n", i + 1, (int)(random() \* 200) - 100);

printf("\ng) 5 latin latter:\n");

char latin[26] = "ABCDEFGHIJKLMNOPQRSTUVXYZ";

for (int i = 0; i < 5; i++)

{

int j = rand() % 25;

if (latin[j] != ' ')

{

printf("\t%d: %c\n", i + 1, latin[j]);

latin[j] = ' ';

}

else

i--;

}

}

Задача 3

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdlib.h>

#include <stdio.h>

#include <time.h>

void main()

{

time\_t t;

srand((unsigned)time(&t));

int playerScore = 100, computerScore = 100;

int stavka, num, n1, n2;

while (playerScore > 0 && computerScore > 0)

{

printf("Player: %d\nComputer: %d\n", playerScore, computerScore);

do {

printf("Input your number:\n");

scanf("%d", &num);

} while (num < 2 || num > 12);

do {

printf("Input cash:\n");

scanf("%d", &stavka);

} while (stavka > playerScore);

n1 = rand() % 7 + 1;

n2 = rand() % 7 + 1;

printf("Two random number:\n%d %d\n", n1, n2);

if (n1 + n2 == num)

{

printf("Player won!\n");

playerScore += 3 \* stavka;

computerScore -= 3 \* stavka;

}

else if (n1 + n2 <= 7 && num <= 7)

{

printf("Player won!\n");

playerScore += stavka;

computerScore -= stavka;

}

else if (n1 + n2 > 7 && num > 7)

{

printf("Player won!\n");

playerScore += stavka;

computerScore -= stavka;

}

else

{

printf("Player lose!\n");

playerScore -= stavka;

computerScore += stavka;

}

printf("\n");

}

if (playerScore <= 0)

printf("Game over. Computer win this party\n");

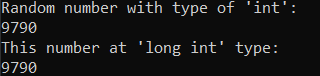
else

printf("Game over. Player win this party\n");

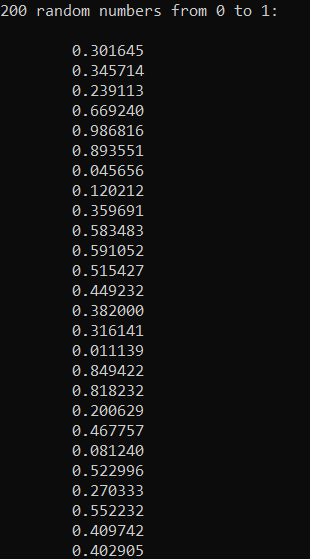
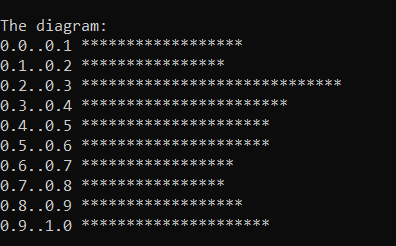
}

**Результат работы программы**

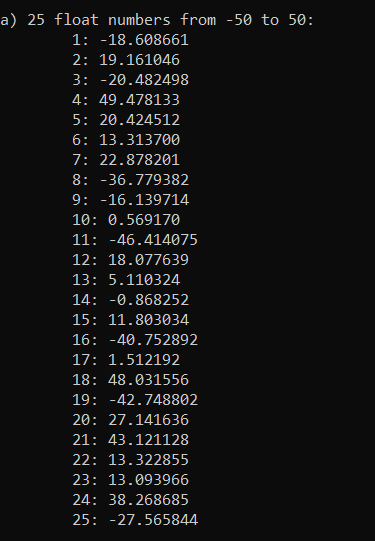
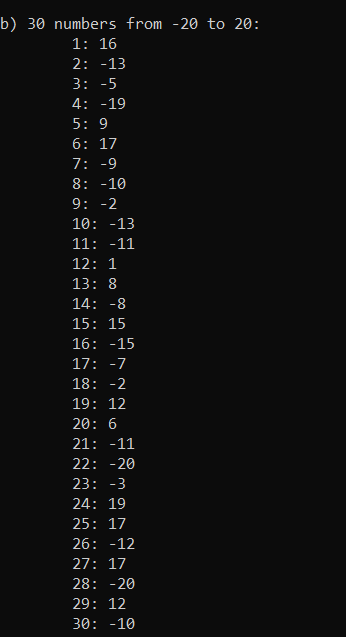
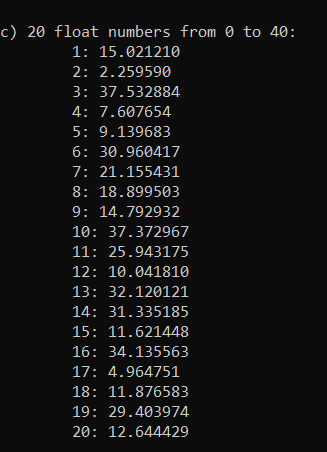
Задача 1

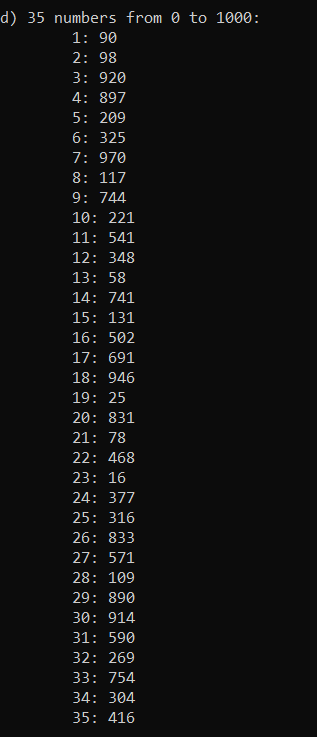


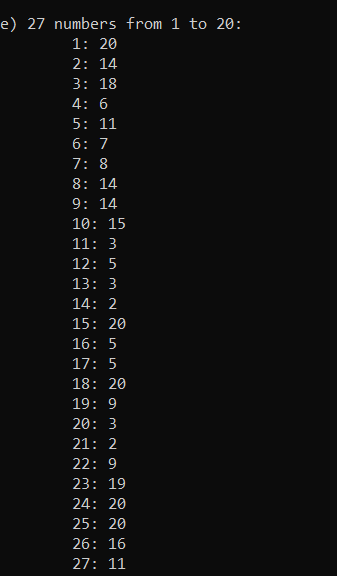
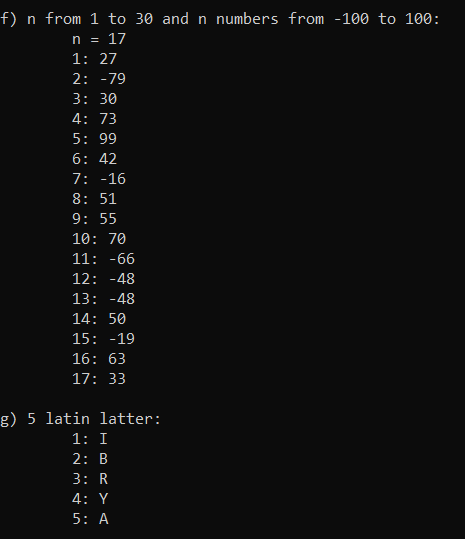
Задача 2

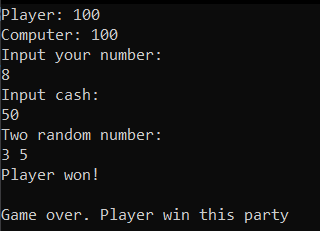
Задача 2.1







Задача 3



**Вывод**

Мной было решено 4 задачи с использованием генератора случайных чисел, решение каждой задачи было реализовано в среде Visual Studio 2019 на языке программирования C++. Написанная программа протестирована и проверена на отладочных данных.

По проделанной работе мной был подготовлен отчёт и сдан преподавателю на проверку.

Я изучил методику тестирования генераторов случайных чисел (ГСЧ) и генераторов псевдослучайных чисел (ГПСЧ).