Министерство науки и высшего образования Российской Федерации

Федеральное государственное бюджетное образовательное учреждение высшего образования «Российский химико-технологический университет имени Д.И. Менделеева»

Факультет цифровых технологий и химического инжиниринга

Кафедра информационных компьютерных технологий

**ОТЧЕТ ПО КУРСОВОЙ РАБОТЕ**

**ПО ТЕХНОЛОГИЯМ ПРОГРАММИРОВАНИЯ**

**«Игра «Кости»»**

**Ведущий преподаватель** Васецкий А.М.

**Студент группы КС-23** Мятлик П.А.

**Москва**

**2023**

**Цель**

Целью данной работы является разработка игры «Кости» на языке C#.

**Задача**

Реализовать алгоритм игры «Кости». Играющий называет любое число в диапазоне от 2 до 12 и ставку, которую он делает в этот ход. Программа с помощью генератора псевдослучайных чисел дважды выбирает числа от 1 до 6. Если сумма выпавших цифр меньше 7 и играющий задумал число меньшее 7, он выигрывает сделанную ставку. Если сумма выпавших цифр больше 7 и играющий задумал число большее 7, он также выигрывает сделанную ставку. Если играющий угадал сумму цифр, он получает в четыре раза больше очков, чем сделанная ставка. Ставка проиграна, если не имеет место ни одна из описанных ситуаций. В начальный момент у играющего 100 очков. Если количество очков становится меньше 0, то игрок проигрывает. Если больше заданного в начале игры максимального значения, то выигрывает.

Построить гистограмму используемых игроком и компьютером чисел с форматированием оси категорий (цвет, тип линии).

Протоколировать ход игры. Стирать старый файл или дополнять его по выбору пользователя.

Запись в бинарный файл всех пользовательских настроек и считывание их оттуда.

**Внешний вид программы**

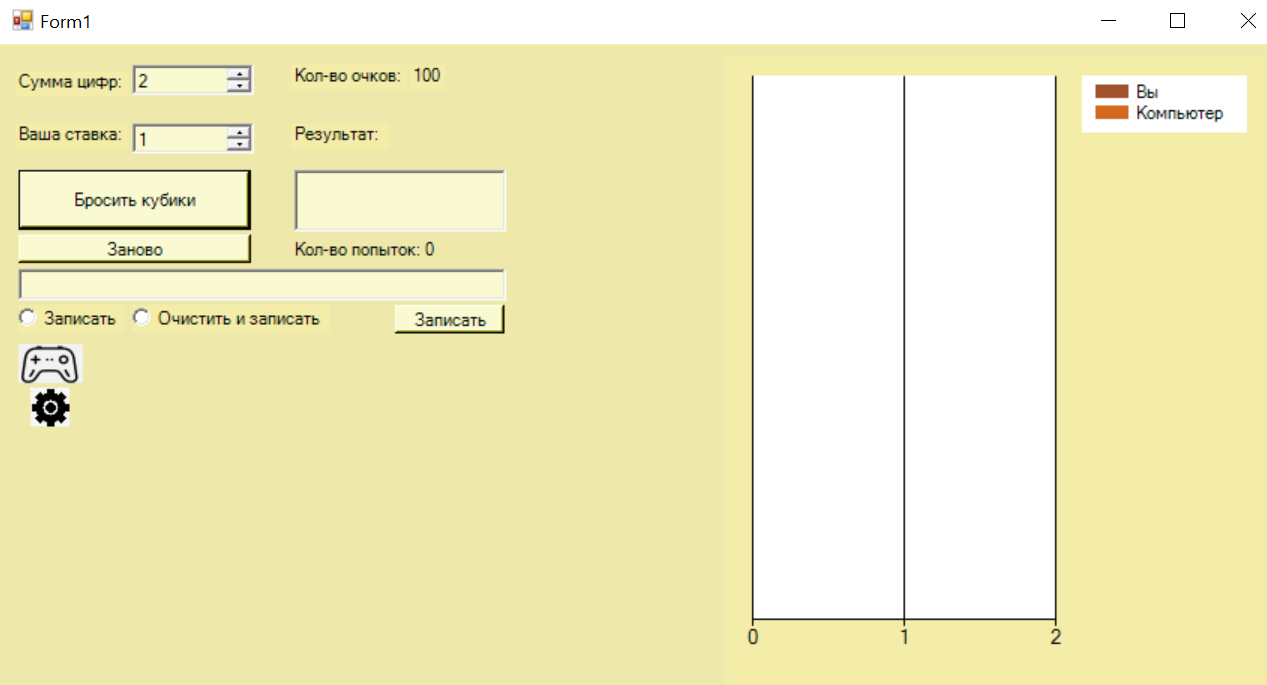


Рисунок 1. Интерфейс программы после запуска

Помимо основного алгоритма игра подразумевает вывод результатов на гистограмму для статистики, запись промежуточных результатов в файл с его предварительной очисткой или без (по выбору пользователя) и настройки игры с возможностью их сохранения в бинарный файл и возвращения настроек к сохранённому.

Настройки можно вывести на экран нажав на иконку «Шестерёнка». Сами настройки включают в себя выбор цветов фона игры и гистограммы пользователем.

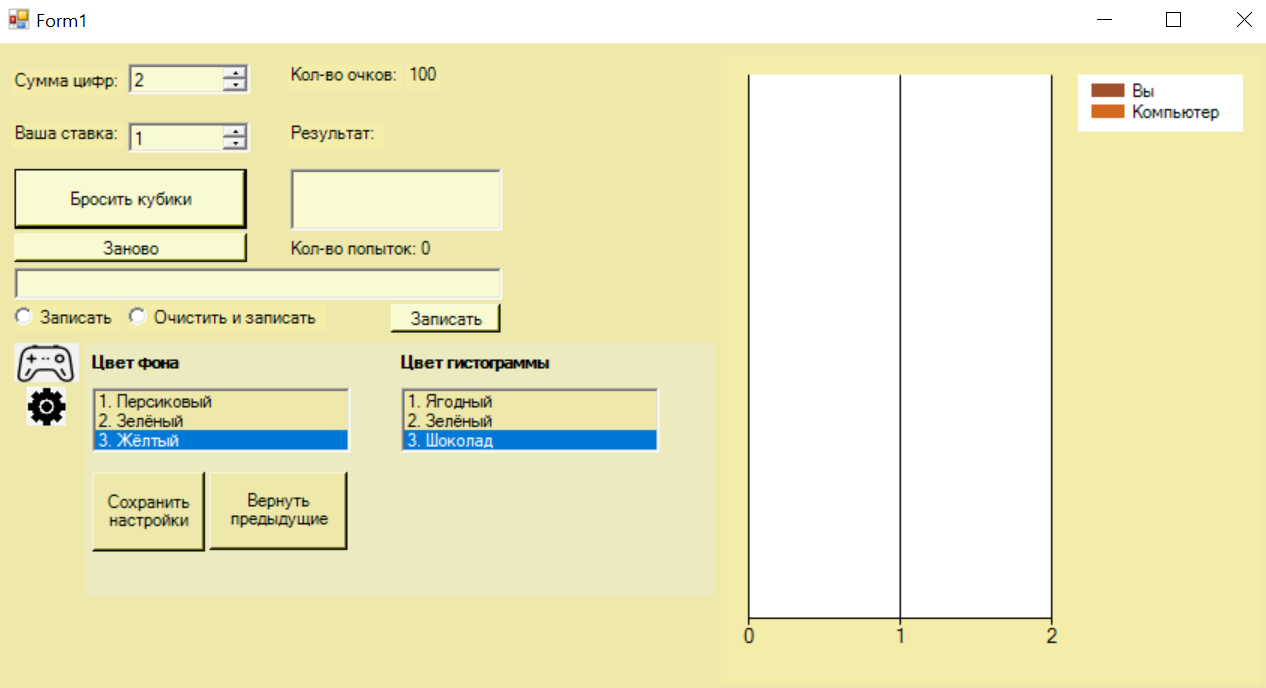
****

Рисунок 2. Настройки

Так же при наведении на иконку выше можно ознакомиться с правилами игры.

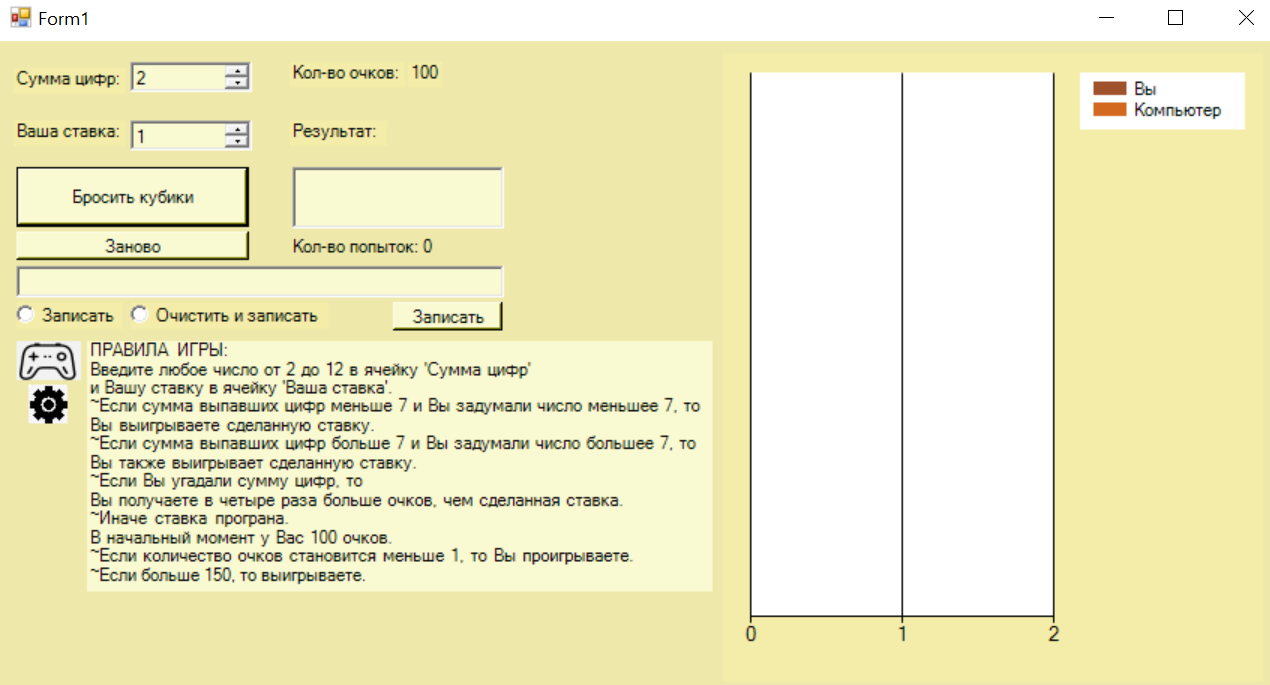
****

Рисунок 3. Правила игры

После того как пользователь ввёл желаемые сумму цифр и ставку следует нажать на кнопку «Бросить кубики». Правее отобразиться результат, заключающий в себе информацию о победе или поражении, и выпавшие очки на каждом из кубиков. Данные отобразятся на гистограмме, увеличится количество попыток.

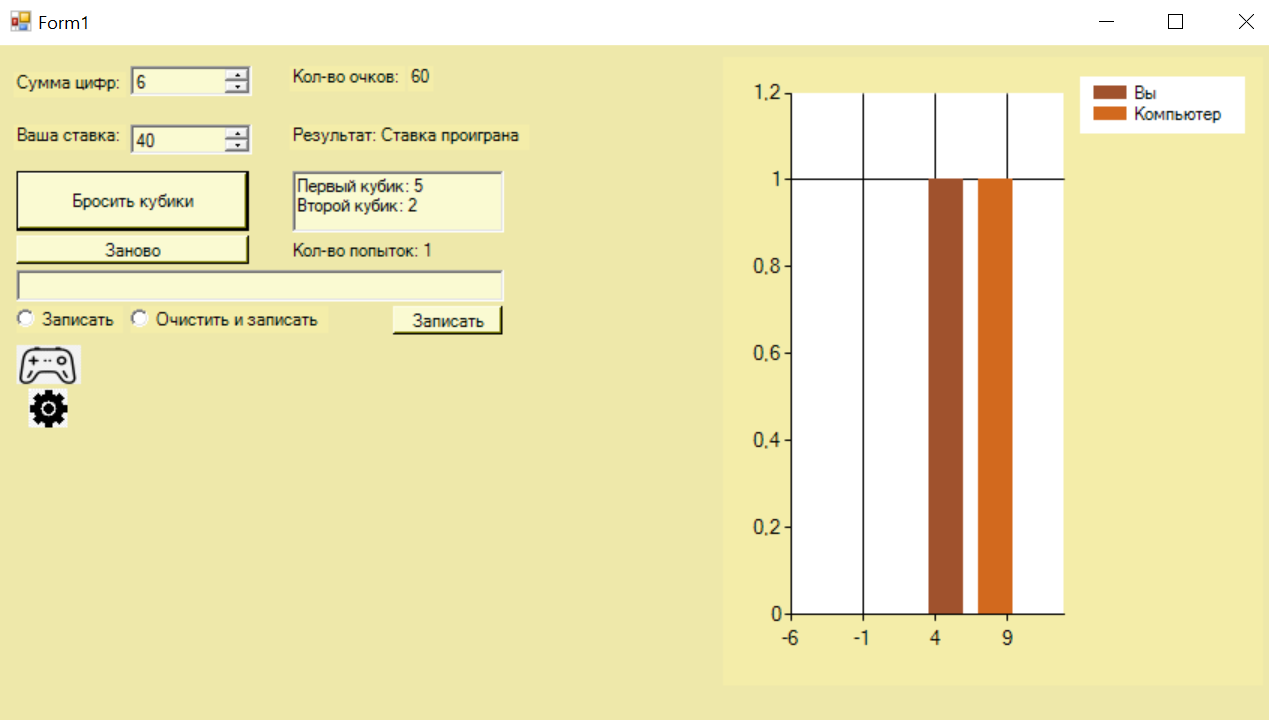


Рисунок 4. Результат после броска кубиков

При желании начать игру заново пользователь может нажать на кнопку «Заново», что позволит обнулить все полученные результаты.

Для записи результатов в файл игрок должен ввести в нужное поле полный путь к файлу и выбрать один из двух пунктов:

1. «Записать» (для записи результатов без предварительной очистки файла)
2. «Очистить и записать» (для записи результатов с предварительной очистки файла)

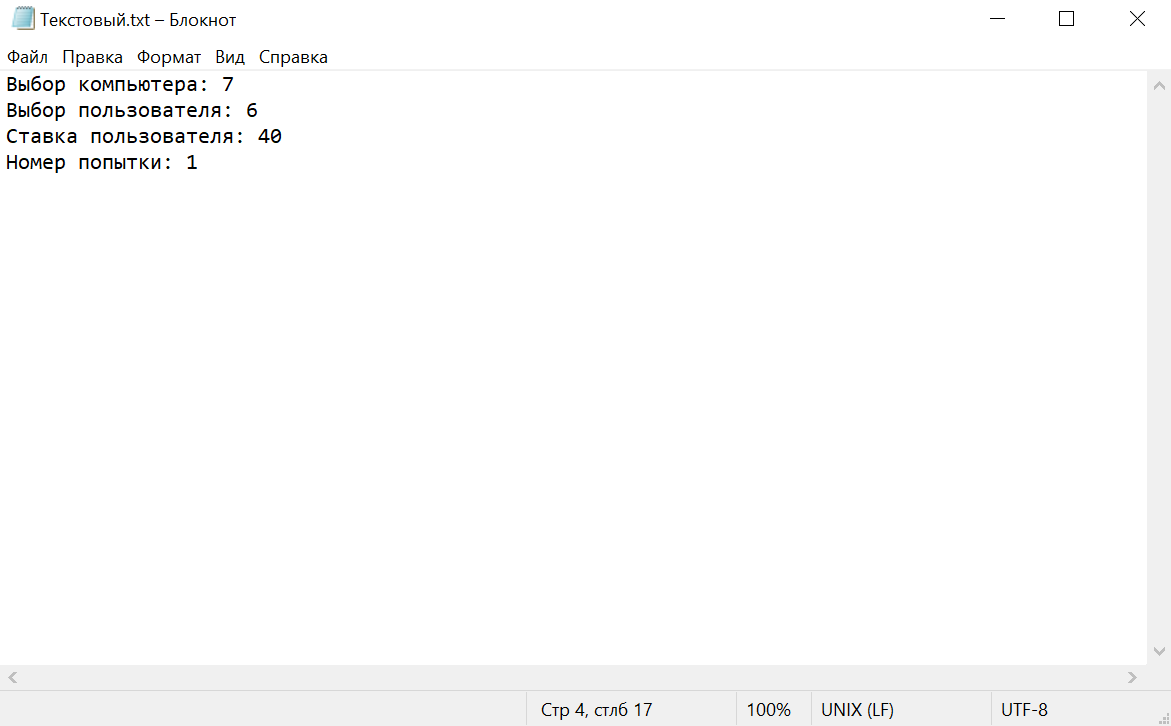


Рисунок 5. Результат записи значений в файл

**Код программы**

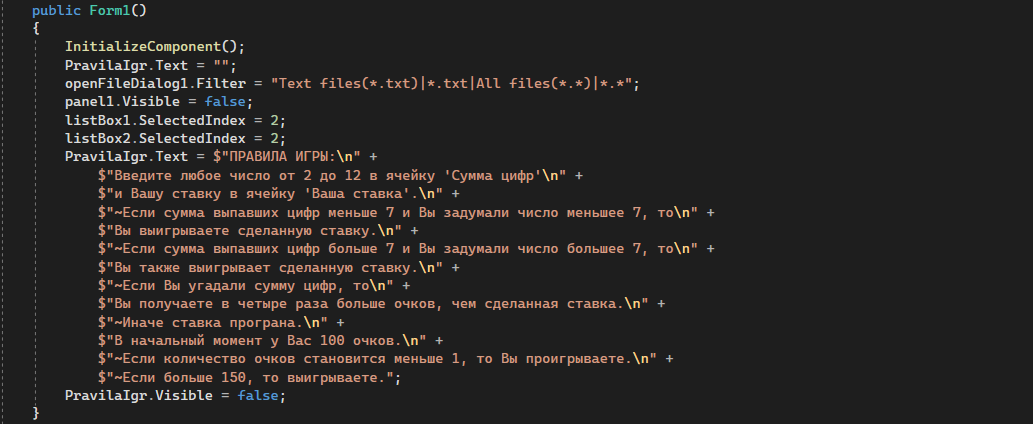


Рисунок 6. Инициализация компонентов перед запуском

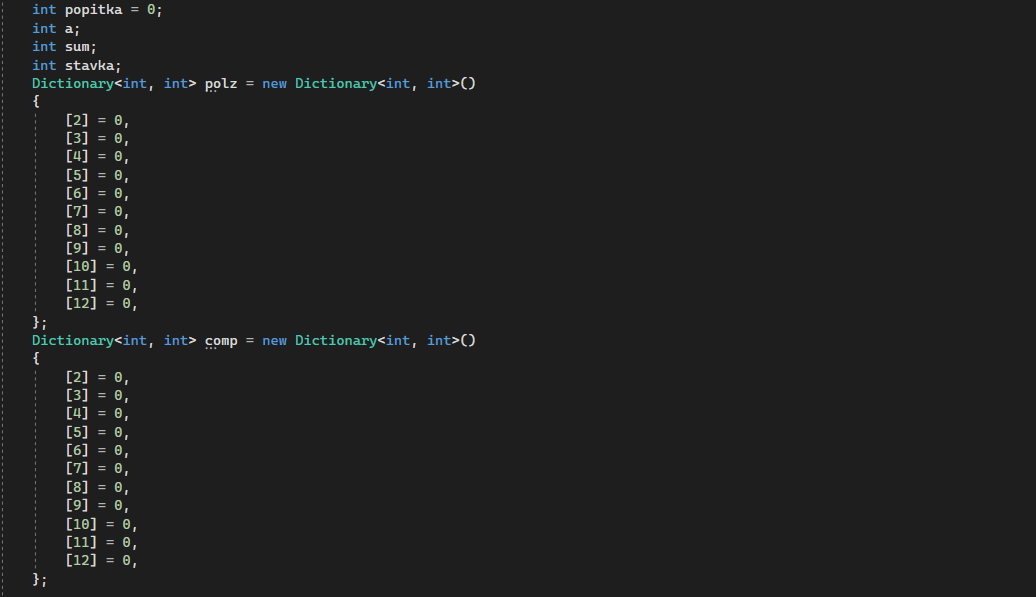


Рисунок 7. Инициализация некоторых переменных и обозначение точек для гистограммы

Ставка пользователя не должна превышать того количества очков, которое есть на счету.

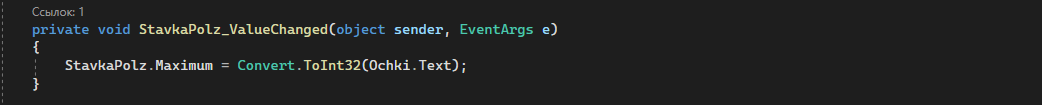


Рисунок 8. Функция для обозначения максимума ставки пользователя

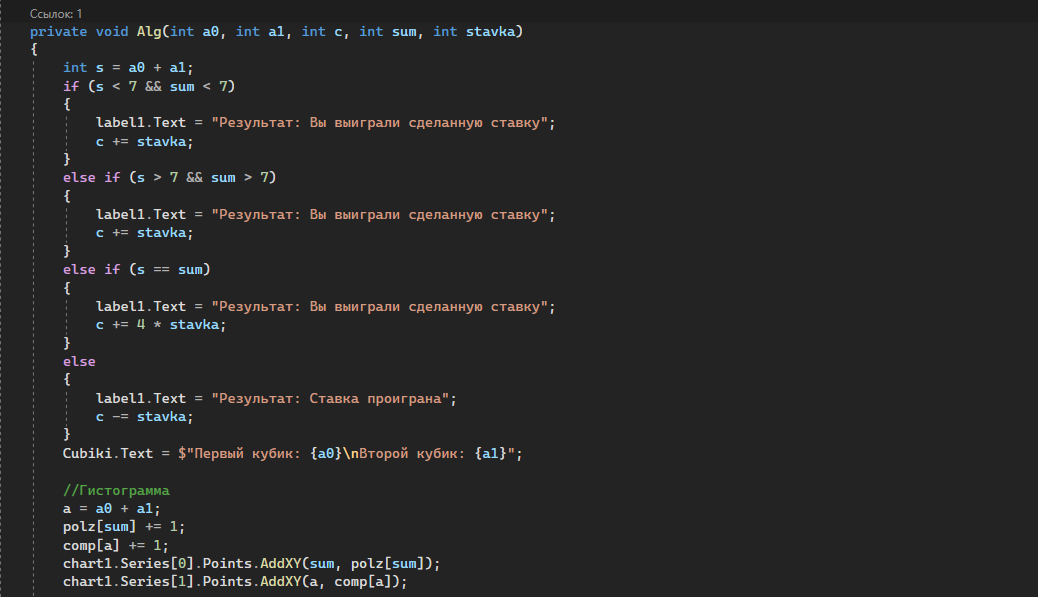


Рисунок 9.1 Основной алгоритм кода

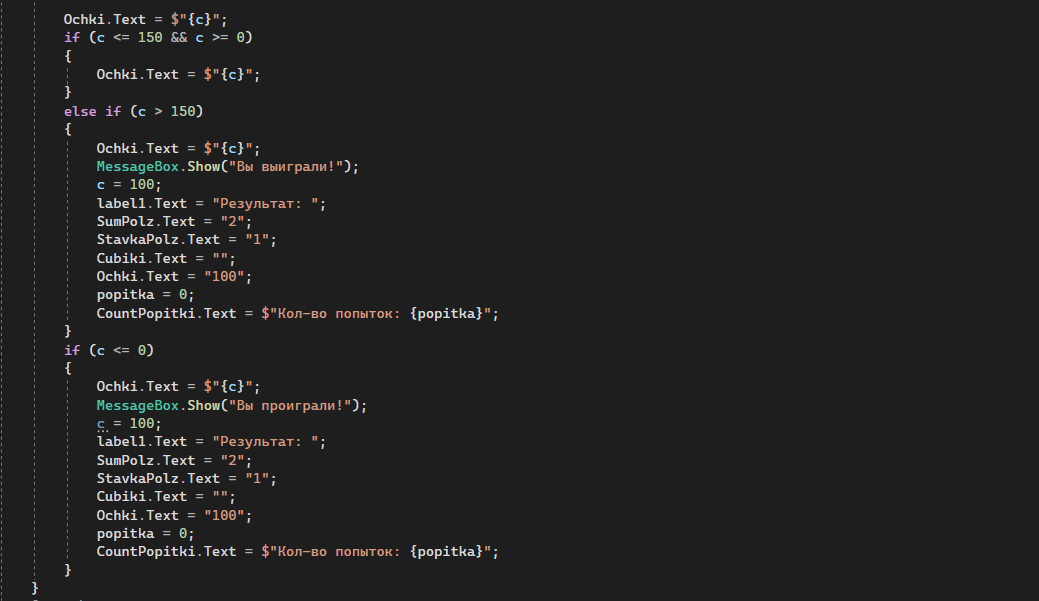


Рисунок 9.2 Основной алгоритм кода

Включение и выключение видимости правил игры осуществляется с помощью наведения и отведения курсора от картинки «Правила\_игры»:

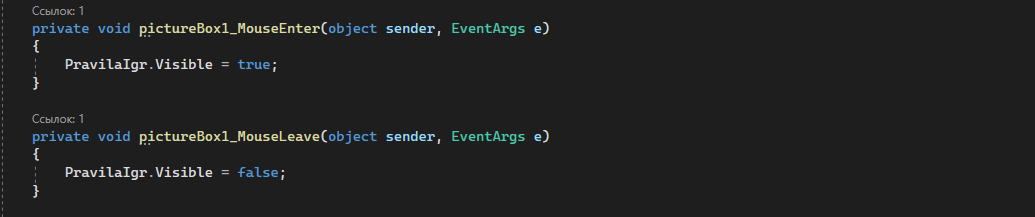
****

Рисунок 10. Правила игры

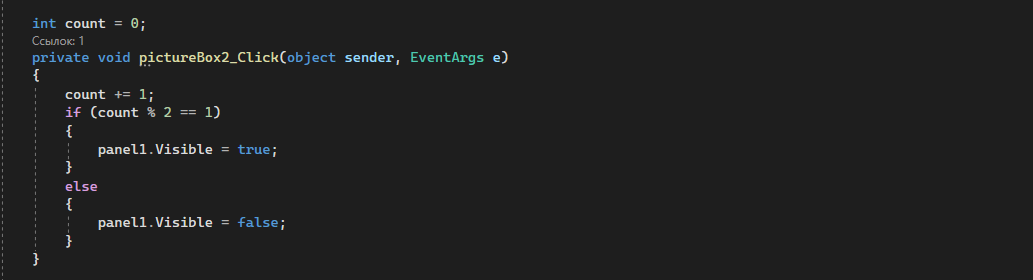
****

Рисунок 11. Настройки

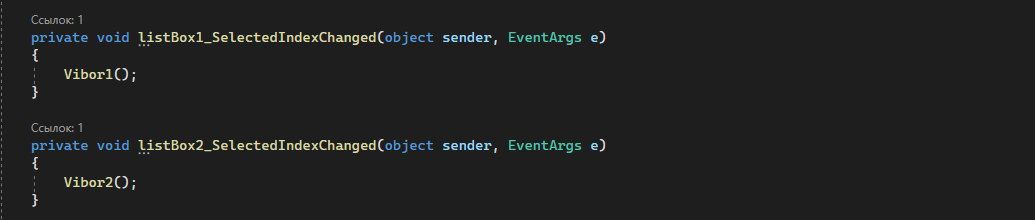


Рисунок 12.1 Изменение настроек

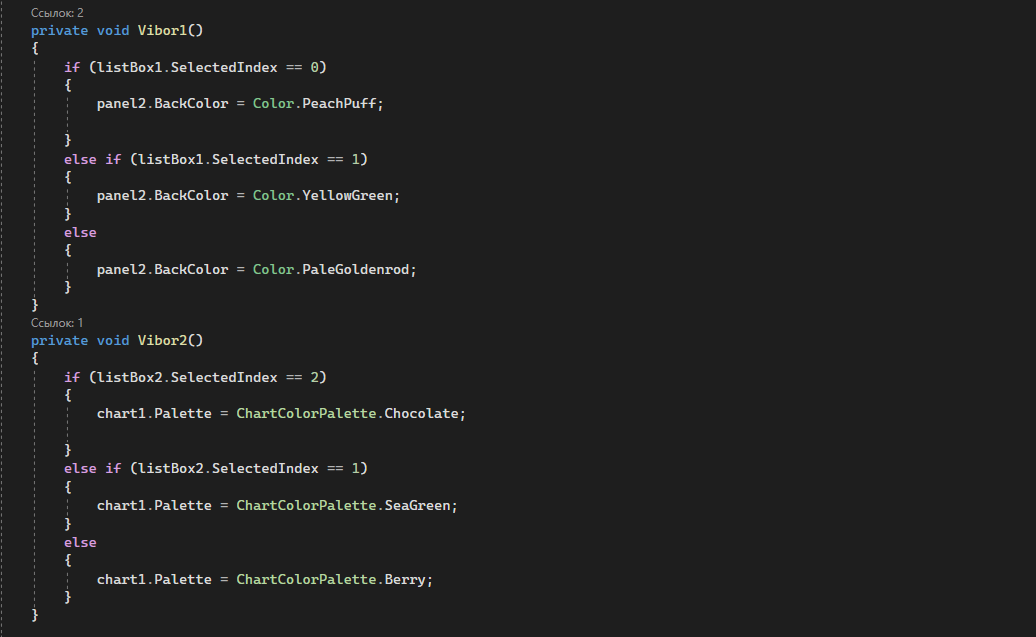
****

Рисунок 12.2 Изменение настроек

****

Рисунок 13.1 Работа с файлом

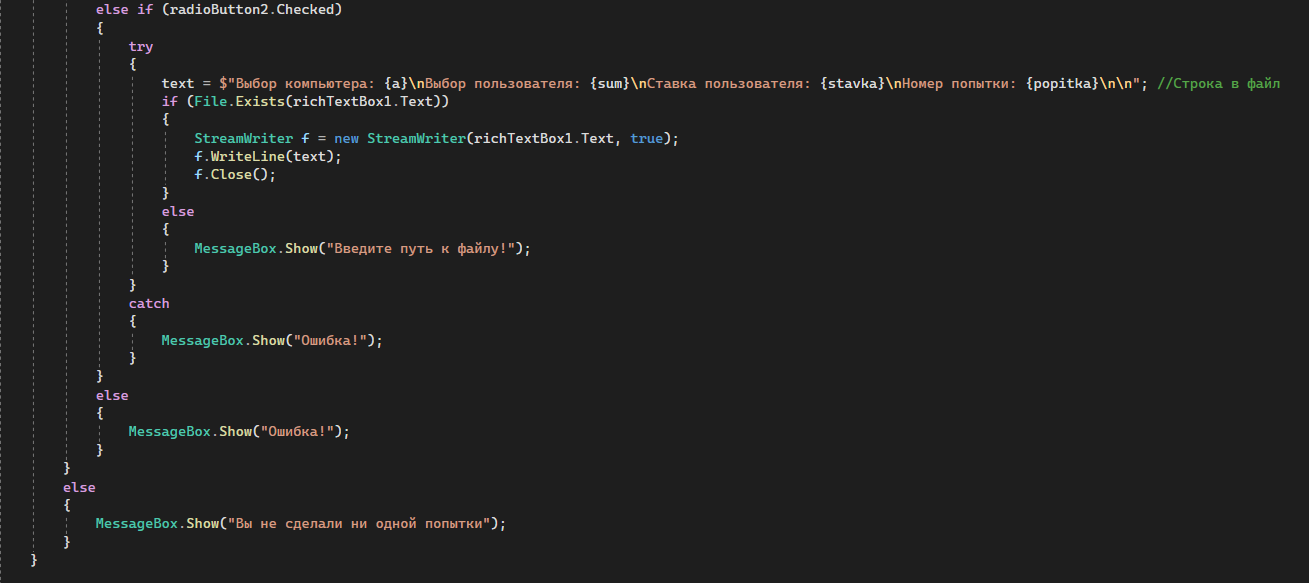
****

Рисунок 13.2 Работа с файлом



Рисунок 14. Алгоритм работы кнопки "Бросить"

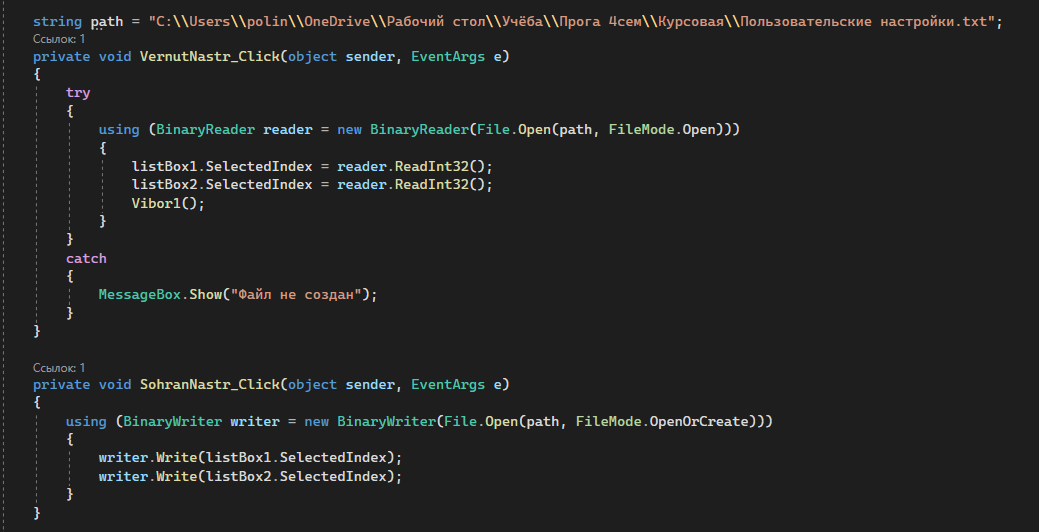


Рисунок 15. Алгоритм работы кнопок "Сохранить настройки" и "Вернуть предыдущие"

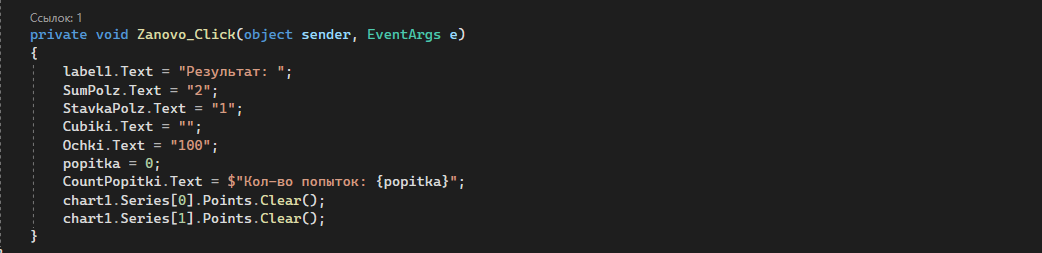


Рисунок 16. Алгоритм работы кнопки "Заново"

**Тестирование**

Тестирование 1

Входные данные: Сумма цифр – 8, ставка – 10, запись результатов в файл с предварительной очисткой, вывод правил игры.

Ожидаемый результат: Изменение количества очков, просмотр результата и количества очков на каждом из кубиков, запись результатов в файл с предварительной очисткой.

Полученный результат: совпадает с ожидаемым.

Тест пройден успешно.

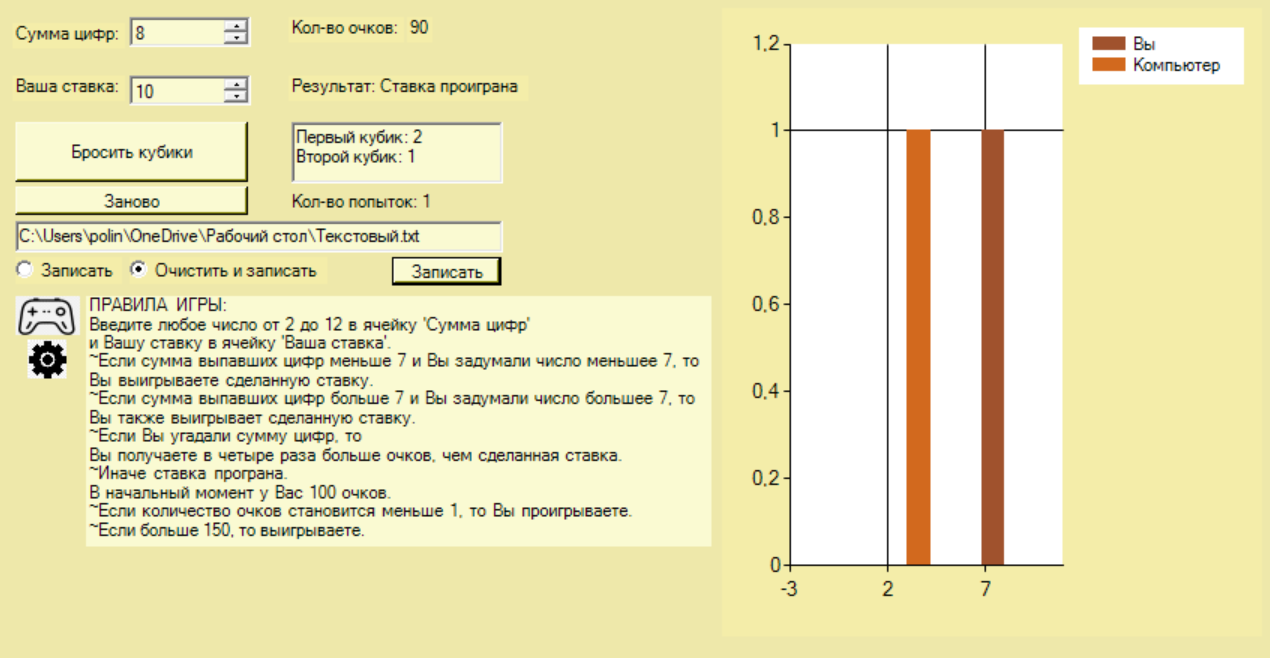


Рисунок 17.1 Результат тестирования 1

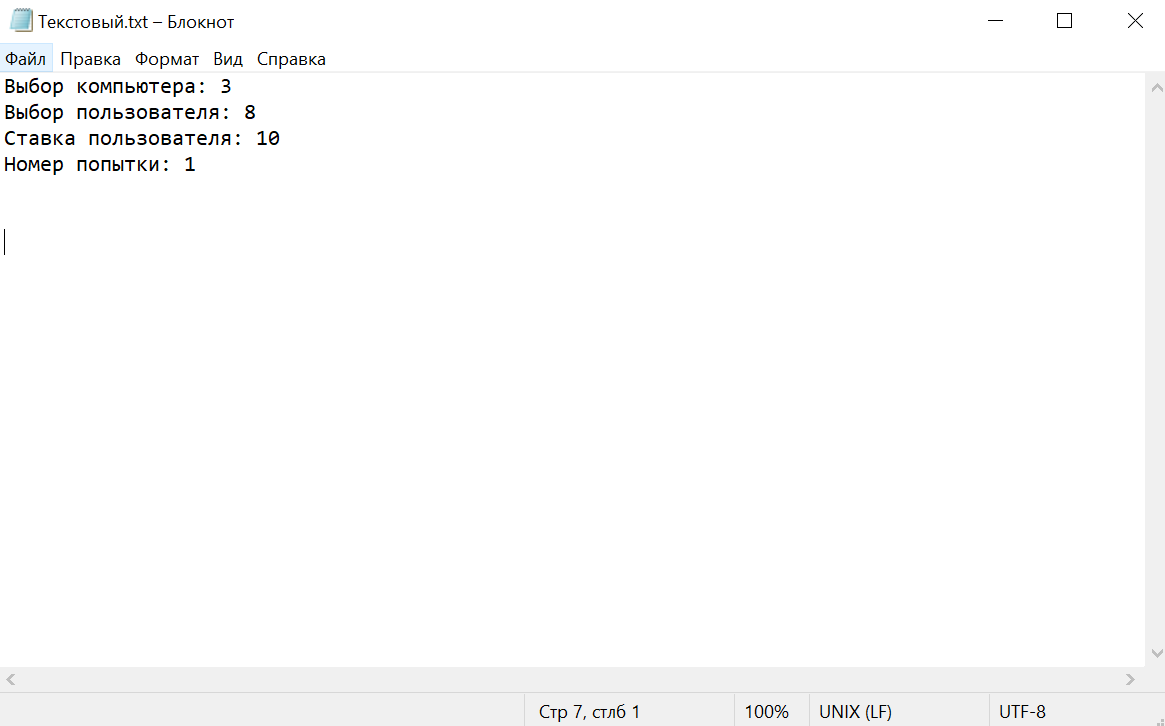


Рисунок 17.2 Результат тестирования 1

Тестирование 2

Входные данные: Сумма цифр – 12, ставка – 90, изменение цвета фона и гистограммы.

Ожидаемый результат: Изменение количества очков, просмотр результата и количества очков на каждом из кубиков, изменение цвета фона и гистограммы.

Полученный результат: совпадает с ожидаемым.

Тест пройден успешно.

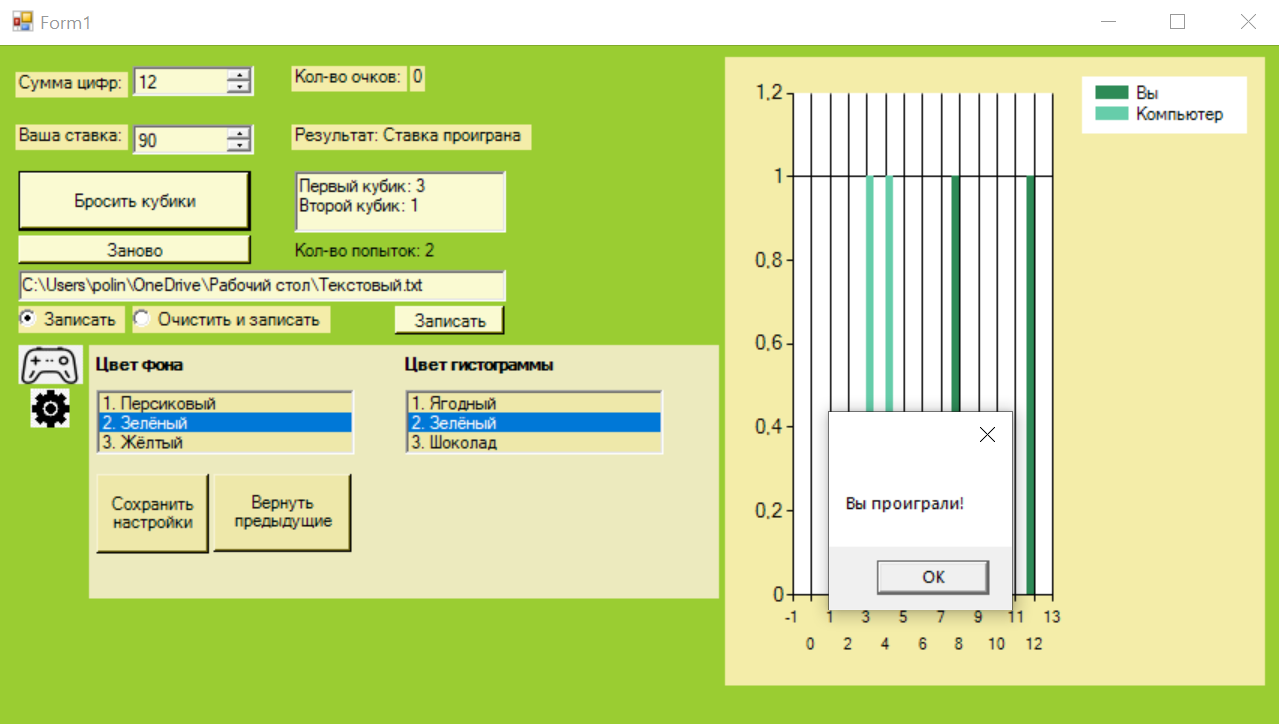
****

Рисунок 18. Результат тестирования 2

Тестирование 3

Входные данные: Сумма цифр – 2, ставка – 100, запись результатов в файл без предварительной очистки, изменение цвета фона и гистограммы, возвращение к сохранённым настройкам.

Ожидаемый результат: Изменение количества очков, просмотр результата и количества очков на каждом из кубиков, запись результатов в файл без предварительной очистки, изменение цвета фона и гистограммы, возвращение к сохранённым настройкам.

Полученный результат: совпадает с ожидаемым.

Тест пройден успешно.

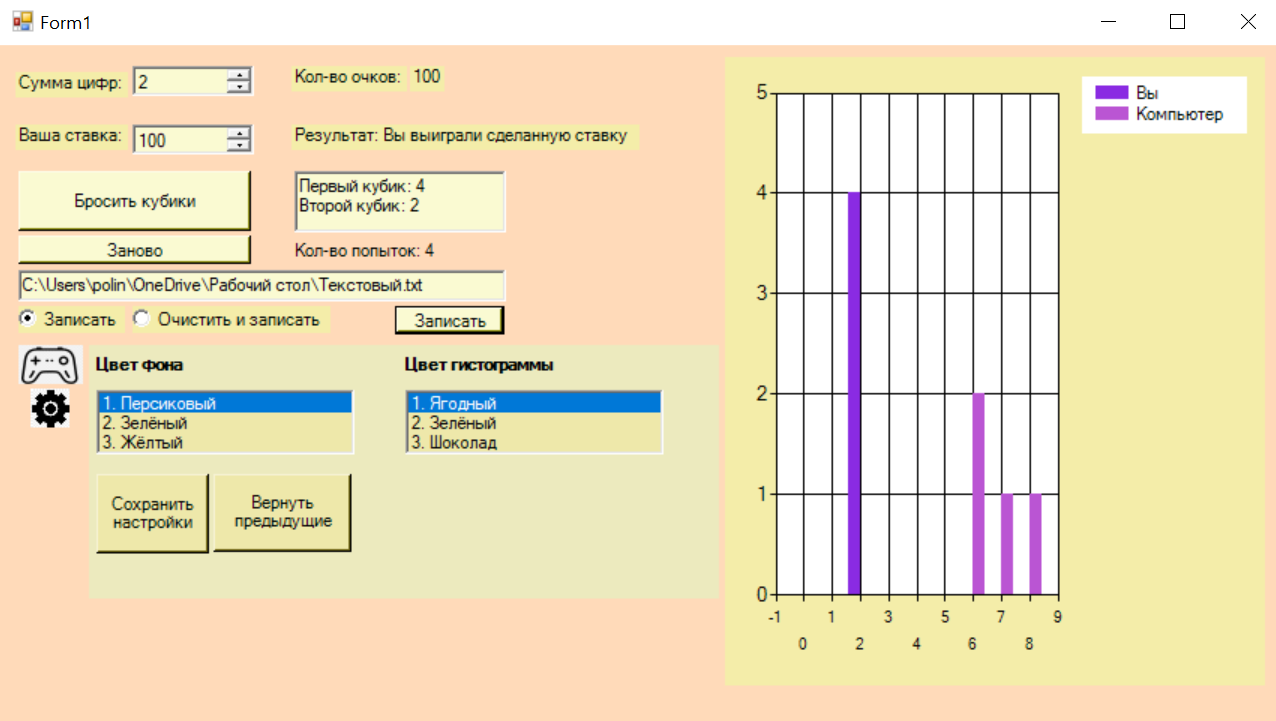


Рисунок 19.1 Результат тестирования 3

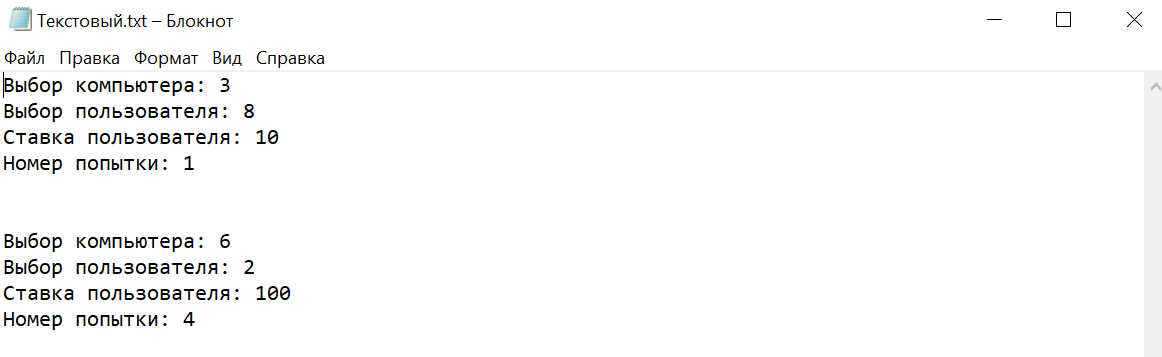


Рисунок 19.1 Результат тестирования 3

**Результаты тестирования**

Программа работает согласно задуманному алгоритму. Интерфейс игры позволяет игроку сразу понять задачу и функцию каждого элемента.