Министерство образования и науки РФ

Федеральное государственное автономное образовательное учреждение высшего образования

«Омский государственный технический университет»

|  |  |
| --- | --- |
| Факультет (институт) | *Информационных технологий и компьютерных систем* |
|  |  |
| Кафедра | *Прикладная математика и фундаментальная информатика* |
|  |  |

**Расчетно-графическая работа**

|  |  |
| --- | --- |
| по дисциплине | ***Алгоритмизация и программирование*** |
|  |  |
| на тему | Разработка программы |

Пояснительная записка

|  |  |
| --- | --- |
| **Шифр проекта** | 020-РГР-02.03.02-№ 25-ПЗ |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  | **Студента** | | Харлов Никита Станиславович | | | | | |
|  |  |  |  | | фамилия, имя, отчество полностью | | | | | |
|  |  |  | Курс | *1* |  | Группа | | ФИТ-**232** | | |
|  |  |  |  |  |  | |  |  | |  |
|  | | | **Направление (специальность)** | | | | | ***02.03.02*** | | |
|  | | | *Фундаментальная информатика и информационные технологии* | | | | | | | |
|  |  |  | код, наименование | | | | | | | |
|  |  |  | Руководитель | | ***ст. преподаватель*** | | | | | |
|  |  |  | ученая степень, звание | | | | | |
|  |  |  | ***Федотова И.В.*** | | | | | | | |
|  |  |  | фамилия, инициалы | | | | | | | |
|  |  |  | Выполнил | |  | | | | | |
|  |  |  | дата, подпись студента | | | | | |
|  |  |  | **Работа защищена с количеством баллов** | | | | | | | |
|  |  |  |  | | | | | |  | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | дата, подпись руководителя |  |  |  |

Омск 2023

Содержание

[1. ВВЕДЕНИЕ 3](#_Toc156173626)

[2. ТЕОРИЯ 4](#_Toc156173627)

[3. РАЗРАБОТКА И КОД ПРОГРАММЫ 5](#_Toc156173628)

[4. РЕЗУЛЬТАТ 9](#_Toc156173629)

[5. ЗАКЛЮЧЕНИЕ 11](#_Toc156173630)

[6. СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ 12](#_Toc156173631)

# **1. ВВЕДЕНИЕ**

Разработка консольных приложений является важным аспектом в современной сфере программирования. В данной расчетно-графической работе мы сосредоточимся на разработке консольного приложения.

В данной работе будет рассмотрена разработка консольного приложения, которое представляет собой программу, выполняемую в текстовом интерфейсе командной строки. Консольные приложения взаимодействуют с пользователем через команды и текстовый ввод-вывод.

Целью данной работы является изучение принципов разработки консольных приложений и применение их на практике, принципы работы с командной строкой, обработка и вывод данных, и другие аспекты, связанные с консольной разработкой

# **2. ТЕОРИЯ**

В данной работе, разработка приложения будет производиться на языке C#. Задача для выполнения: Программа с помощью датчика случайных чисел выбирает число в диапазоне от 0 до N. Угадать это число с ограничением числа попыток и без ограничения. После каждой попытки сообщается, больше или меньше названное число задуманного.

Для выполнения данной работы мне понадобятся знания о работе с консолью: ввод-вывод, умение создавать алгоритмы и умение работы с датчиком случайных чисел.

# **3. РАЗРАБОТКА И КОД ПРОГРАММЫ**

Код выполнен в программе: Visual Studio 2022.

В основе программы лежит следующая схема: пользователь выбирает режим игры: с ограничением или без ограничения, пользователь выбирает максимальное натуральное число, в случае режима с ограничениями программа выбирает количество попыток, зависящее от выбранного пользователем число, далее начинается игра: каждый ход пользователь вводит число и получает ответ, в котором указано меньше искомое число данного или больше. В случае нахождения числа игра заканчивается победой. В игре с ограничением, когда закончатся попытки будет поражение.

Во время разработки, я придумал алгоритм, по которому будет выбираться количество попыток. Вспомнив эффективность алгоритма бинарного поиска, мы узнаем, что найти число в списке из n элементов можно за log2(n) итераций, но для человека это число умножается на коэффициент.

countOfTrurns = Convert.ToInt16(Math.Ceiling(Math.Log2(n) \* 1.2));

Структура программы.

Первый блок программы выбирает режим игры:

Console.WriteLine("+------- Choose Game Mode:");

Console.WriteLine("| \* 1 - Game with limit of turns");

Console.WriteLine("| \* 2 - Game without limit of turns");

Console.Write("| ");

if (Console.ReadLine() == "1")

{

GameModeTurns = true;

Console.WriteLine("| \* Game with limits \*");

}

else

{

GameModeTurns = false;

Console.WriteLine("| \* Game without limits \*");

}

Второй блок программы выбирает максимальное натуральное число (по умолчанию 100) и при режиме с ограничением устанавливает число попыток по описанному выше правилу:

Console.WriteLine("| Enter Limit of Numbers: (by default = 100)");

Console.Write("| ");

string s = Console.ReadLine();

if (s.Length > 0)

{

n = int.Parse(s);

}

else

{

n = 100;

}

if (GameModeTurns)

{

countOfTrurns = Convert.ToInt16(Math.Ceiling(Math.Log2(n) \* 1.2));

Console.WriteLine("| Count of Turns set on " + countOfTrurns);

}

else

{

countOfTrurns = 1000;

}

Затем программа загадывает число:

Random r = new Random();

numberToFind = r.Next(n);

И в третьем блоке происходит сама игра:

Console.WriteLine("+---- Game Starts ");

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

{

Console.Write("| Enter supposed number: ");

playersNumber = int.Parse(Console.ReadLine());

if (playersNumber != numberToFind)

{

if (playersNumber > numberToFind)

{

Console.WriteLine("| Number you need is less");

}

else

{

Console.WriteLine("| Number you need is bigger");

}

}

else if (playersNumber == numberToFind)

{

Console.WriteLine("+---- Success number that you need " + numberToFind );

suc = false;

break;

}

if (GameModeTurns) { Console.WriteLine("| Turns left: " + (countOfTrurns-i-1)); }

}

if (suc) { Console.WriteLine("+--- You lost, sorry "); }

Весь код программы:

using System;

namespace nums

{

class Play

{

static void Main()

{

Random r = new Random();

bool suc = true;

int n;

int playersNumber;

int numberToFind;

int countOfTrurns;

bool GameModeTurns;

Console.WriteLine("+------- Choose Game Mode:");

Console.WriteLine("| \* 1 - Game with limit of turns");

Console.WriteLine("| \* 2 - Game without limit of turns");

Console.Write("| ");

if (Console.ReadLine() == "1")

{

GameModeTurns = true;

Console.WriteLine("| \* Game with limits \*");

}

else

{

GameModeTurns = false;

Console.WriteLine("| \* Game without limits \*");

}

Console.WriteLine("| Enter Limit of Numbers: (by default = 100)");

Console.Write("| ");

string s = Console.ReadLine();

if (s.Length > 0)

{

n = int.Parse(s);

}

else

{

n = 100;

}

if (GameModeTurns)

{

countOfTrurns = Convert.ToInt16(Math.Ceiling(Math.Log2(n) \* 1.2));

Console.WriteLine("| Count of Turns set on " + countOfTrurns);

}

else

{

countOfTrurns = 1000;

}

numberToFind = r.Next(n);

//Игра Начинается

Console.WriteLine("+---- Game Starts ");

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

{

Console.Write("| Enter supposed number: ");

playersNumber = int.Parse(Console.ReadLine());

if (playersNumber != numberToFind)

{

if (playersNumber > numberToFind)

{

Console.WriteLine("| Number you need is less");

}

else

{

Console.WriteLine("| Number you need is bigger");

}

}

else if (playersNumber == numberToFind)

{

Console.WriteLine("+---- Success number that you need " + numberToFind );

suc = false;

break;

}

if (GameModeTurns) { Console.WriteLine("| Turns left: " + (countOfTrurns-i-1)); }

}

if (suc) { Console.WriteLine("+--- You lost, sorry "); }

}

}

}

# **4. РЕЗУЛЬТАТ**

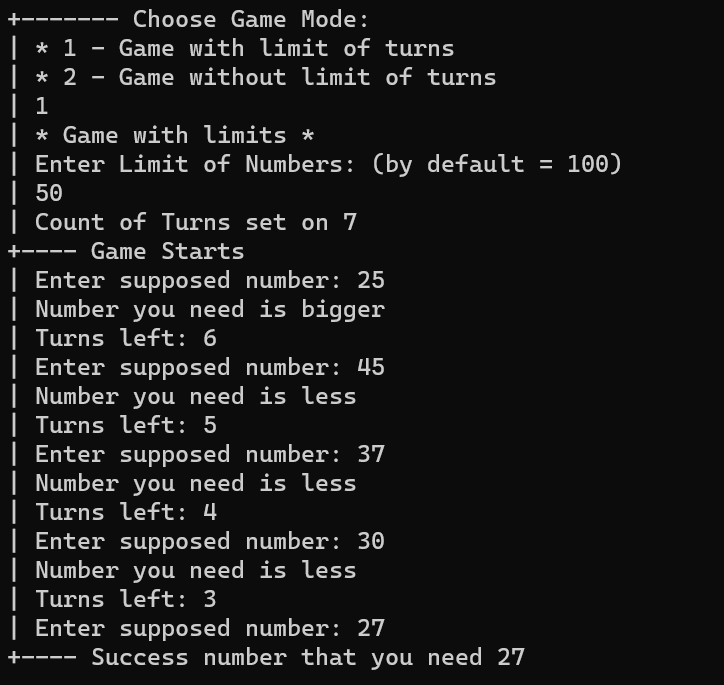
Результат работы программы в режиме ограниченного числа попыток представлен на рисунке 1.

Рисунок 1 — Режим ограниченного числа попыток

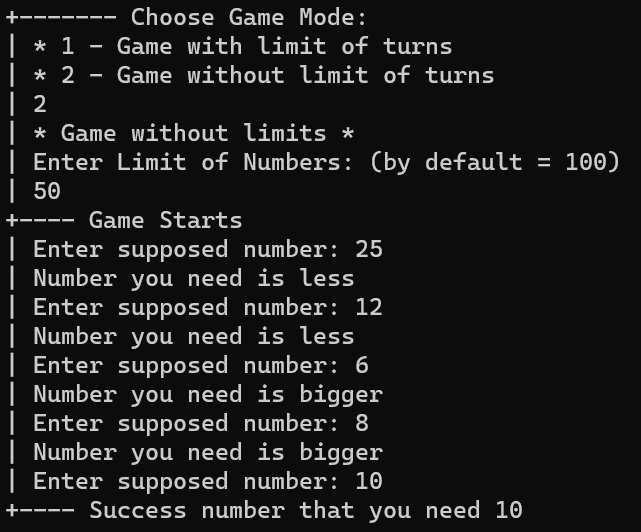
Результат работы программы в режиме неограниченного числа попыток представлен на рисунке 2.

Рисунок 2 — Режим неограниченного числа попыток

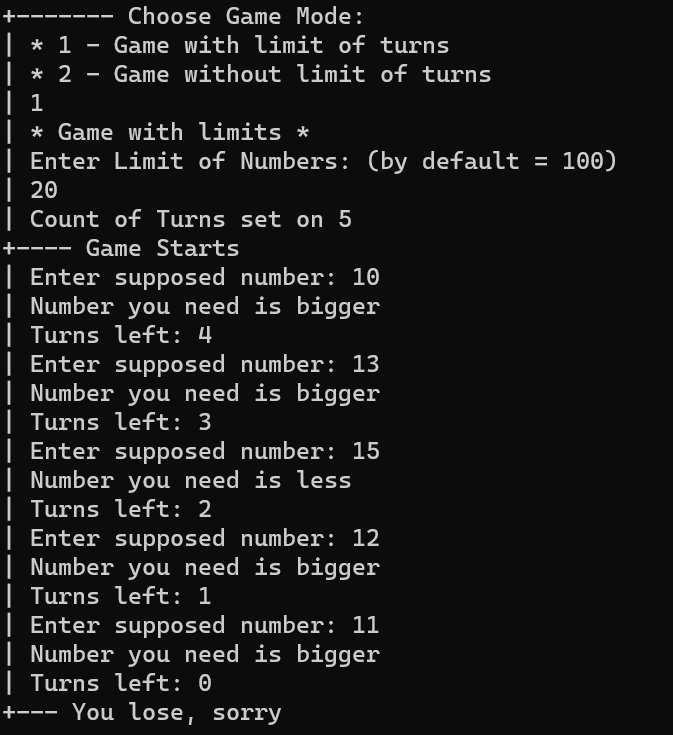
Результат работы программы в случае проигрыша пользователя представлен на рисунке 3.

Рисунок 3 — Проигрыш пользователя

# **5. ЗАКЛЮЧЕНИЕ**

В ходе выполнения данной расчётно-графической работы, мною было создано консольное приложение, которое отвечает требованиям, поставленным в задании.

Я разработал консольное приложение, которые обрабатывает входные данные и выводит информацию.

# **6. СПИСОК ИСПОЛЬЗОВАННОЙ ЛИТЕРАТУРЫ**

1) learn.microsoft.com [Электронный ресурс].

URL: <https://learn.microsoft.com/ru-ru/dotnet/api/system.random?view=net-8.0>

(Дата обращения: 09.01.2024).

2) learn.microsoft.com [Электронный ресурс].

URL: <https://learn.microsoft.com/ru-ru/dotnet/api/system.math.log?view=net-8.0>

(Дата обращения: 09.01.2024).