**Документация бригады №2 группы А-06-18**

1. **Задание**

Разработка алгоритма для анализа и рационализации матричных игр:

* Ввод, вывод и представление матричных данных;
* Поиск максимина, минимакса и седловой точки в матричных моделях;

Полный код программы представлен в приложении 1.

1. **Реализуемые функции**
   1. **Функция Minimax**

Данная функция находит максимумы в каждом столбце матрицы, которые записываются в массив. В массиве выбирается минимальное число, значение которого возвращает функция.

* 1. **Функция Maximin**

Данная функция находит минимумы в каждой строке матрицы, которые записываются в массив. В массиве выбирается максимальное число, значение которого возвращает функция.

* 1. **Функция Ma()**

Данная функция:

* принимает в себя наименование стратегий и выигрыш игроков для матрицы размером 2х3;
* выводит матрицу на экран;
* находит максимин, минимакс и седловую точку для заданных значений.
  1. **Функция Mb()**

Данная функция:

* принимает в себя наименование стратегий и выигрыш игроков для матрицы размером 2х2;
* выводит матрицу на экран;
* находит максимин, минимакс и седловую точку для заданных значений.

1. **Основная часть**

Код основной части состоит из вывода меню на экран и реализации выбора одного из его пунктов через цикл. Пока не выбран выход из меню, программа работает и может выполнять задания.

1. **Ограничения**

Так как для хранения выигрыша используется тип данных int, то вводимые значения не должны превышать граничных значений для этого типа.

В выигрыш не может быть введено ничего кроме целого числа.

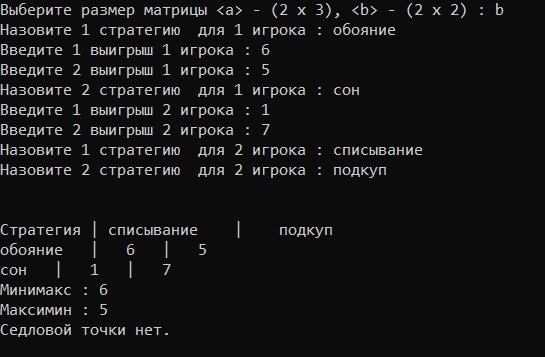
1. **Запуск**

После запуска появляется интерактивное взаимодействие с пользователем, пока не появится результат.

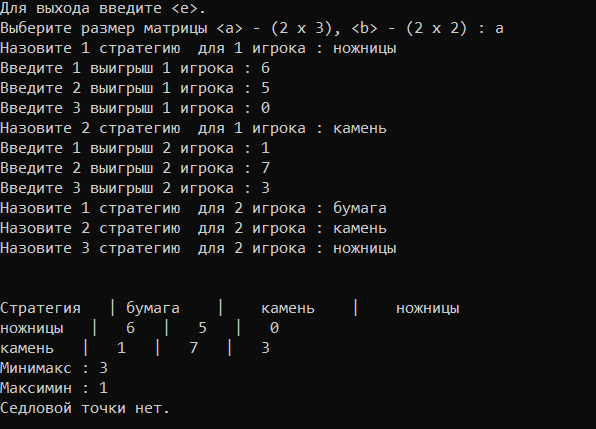
1. **Тестирование**

В результате тестирования было выявлено, что работа соответствует ожидаемым результатам. Примеры тестов представлены далее.

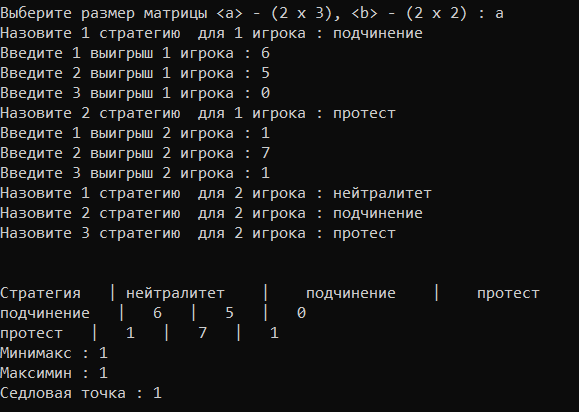
**Тест1= нет седловой точки в 2х2**

****

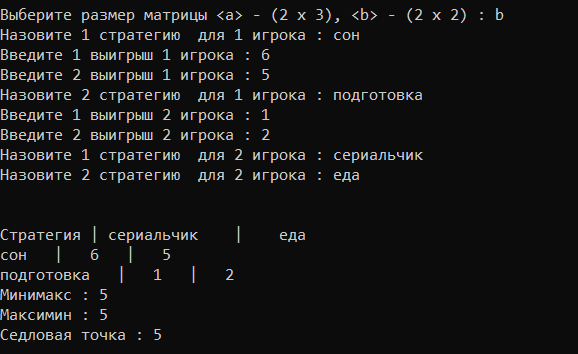
**Тест2= нет седловой точки в 2х3**

****

**Тест3= есть седловая точка в 2х3**

****

**Тест4= есть седловая точка в 2х2**

****

1. **Приложение 1**

#include <iostream>

#include <vector>

#include <string>

#include<stdio.h>

#include<windows.h>

int Minimax(int\*\* arr, int ii, int jj)

{

std::vector<int> vec;

for (int j = 0; j < jj; j++)

{

vec.push\_back(arr[0][j]);

}

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

{

for (int j = 0; j < jj; j++)

{

if (arr[i][j] > vec[j])

vec[j] = arr[i][j];

}

}

int mima = vec[0];

for (int i = 0; i < vec.size(); i++)

if (vec[i] < mima)

mima = vec[i];

return mima;

}

int Maximin(int\*\* arr, int ii, int jj)

{

std::vector<int> vec;

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

{

vec.push\_back(arr[i][0]);

}

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

{

for (int j = 0; j < jj; j++)

{

if (arr[i][j] < vec[i])

vec[i] = arr[i][j];

}

}

int mami = vec[0];

for (int i = 0; i < vec.size(); i++)

if (vec[i] > mami)

mami = vec[i];

return mami;

}

void Ma()

{

int\*\* arr = new int\* [2]; // две строки в массиве

for (int count = 0; count < 2; count++)

arr[count] = new int[3];

std::vector<std::string> p1;

std::vector<std::string> p2;

std::string str;

for (int i = 0; i < 2; i++)

{

std::cout << "Назовите " << i + 1 << " стратегию для 1 игрока : ";

std::cin >> str;

p1.push\_back(str);

for (int j = 0; j < 3; j++)

{

std::cout << "Введите " << j + 1 << " выигрыш " << i + 1 << " игрока : ";

std::cin >> arr[i][j];

}

}

for (int j = 0; j < 3; j++)

{

std::cout << "Назовите " << j + 1 << " стратегию для 2 игрока : ";

std::cin >> str;

p2.push\_back(str);

}

/\*for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 3; j++)

{

std::cout << arr[i][j] << " ";

}

std::cout << '\n';

}\*/

std::cout << "\n\n";

std::cout << "Стратегия | " << p2[0] << " | " << p2[1] << " | " << p2[2] << std::endl;

std::cout << p1[0] << " | " << arr[0][0] << " | " << arr[0][1] << " | " << arr[0][2] << std::endl;

std::cout << p1[1] << " | " << arr[1][0] << " | " << arr[1][1] << " | " << arr[1][2] << std::endl;

int a = Minimax(arr, 2, 3);

int b = Maximin(arr, 2, 3);

std::cout << "Минимакс : " << a << std::endl;

std::cout << "Максимин : " << b << std::endl;

if (a == b)

std::cout << "Седловая точка : " << a << std::endl;

else std::cout << "Седловой точки нет." << std::endl;

std::cout << "\n\n";

for (int count = 0; count < 2; count++)

delete[] arr[count];

}

void Mb()

{

int\*\* arr = new int\* [2]; // две строки в массиве

for (int count = 0; count < 2; count++)

arr[count] = new int[2];

std::vector<std::string> p1;

std::vector<std::string> p2;

std::string str;

//bool f{ 1 };

for (int i = 0; i < 2; i++)

{

std::cout << "Назовите " << i + 1 << " стратегию для 1 игрока : ";

std::cin >> str;

p1.push\_back(str);

for (int j = 0; j < 2; j++)

{

std::cout << "Введите " << j + 1 << " выигрыш " << i + 1 << " игрока : ";

std::cin >> arr[i][j];

}

}

for (int j = 0; j < 2; j++)

{

std::cout << "Назовите " << j + 1 << " стратегию для 2 игрока : ";

std::cin >> str;

p2.push\_back(str);

}

/\*for (int i = 0; i < 2; i++)

{

for (int j = 0; j < 2; j++)

{

std::cout << arr[i][j] << " ";

}

std::cout << '\n';

}\*/

std::cout << "\n\n";

std::cout << "Стратегия | " << p2[0] << " | " << p2[1] << std::endl;

std::cout << p1[0] << " | " << arr[0][0] << " | " << arr[0][1] << std::endl;

std::cout << p1[1] << " | " << arr[1][0] << " | " << arr[1][1] << std::endl;

int a = Minimax(arr, 2, 2);

int b = Maximin(arr, 2, 2);

std::cout << "Минимакс : " << a << std::endl;

std::cout << "Максимин : " << b << std::endl;

if (a == b)

std::cout << "Седловая точка : " << a << std::endl;

else std::cout << "Седловой точки нет." << std::endl;

std::cout << "\n\n";

for (int count = 0; count < 2; count++)

delete[] arr[count];

}

int main()

{

SetConsoleOutputCP(1251);

SetConsoleCP(1251);

char c;

std::cout << "Для выхода введите <e>.\n";

std::cout << "Выберите размер матрицы <a> - (2 x 3), <b> - (2 x 2) : ";

std::cin >> c;

while (c != 'e')

{

if (c == 'a')

{

Ma();

}

else if (c == 'b')

{

Mb();

}

else

{

std::cout << "Ввод не распознан!\n";

}

std::cout << "Выберите размер матрицы <a> - (2 x 3), <b> - (2 x 2) : ";

std::cin >> c;

}

}