МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ
УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

**«БЕЛГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ
ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ им. В. Г. ШУХОВА»
(БГТУ им. В.Г. Шухова)**

Кафедра программного обеспечения вычислительной техники и автоматизированных систем

Лабораторная работа №6

по дисциплине: Исследование операций и теория игр
тема: «Нахождение седловой точки в смешанных стратегиях для матричной игры с нулевой суммой»

Выполнил: ст. группы ПВ-211
Стародубов Алексей Геннадьевич
Проверили:
Куртова Лилиана Николаевна
Вирченко Юрий Петрович

Белгород 2023 г.

Цель работы: освоить метод нахождения седловой точки в смешанных стратегиях с помощью построения пары двойственных задач ЛП.

Вариант- 19

Задания для подготовки к работе

1. Изучить основные понятия теории матричных игр двух игроков с нулевой суммой, анализ игры в чистых стратегиях, понятие смешанной стратегии и седловой точки в смешанных стратегиях, а также метод нахождения седловой точки в смешанных стратегиях с помощью построения пары двойственных задач ЛП.
2. Составить и отладить программу для нахождения седловой точки игры с помощью решения пары симметрично двойственных задач ЛП.
3. Для подготовки тестовых данных решить вручную одну из следующих ниже задач.

$$\begin{pmatrix} 8 & 3 & 6 \\ 7 & 6 & 5 \\ 3 & 9 & 4 \\ 5 & 2 & 10 \end{pmatrix}$$

2.

```
#include <iostream>
#include <vector>
#include <string>
#include <iomanip>

using SimplexTable = std::vector<std::pair<std::string,
std::vector<double>>>;

void outputSimplexTable(const SimplexTable &simplexTable)
{
    std::cout << "BV\tFV\t";

    for (size_t i{1}; i < simplexTable.at(0).second.size(); ++i)
    {
        std::cout << "y" << i << '\t';
```

```
    }
    std::cout << '\n';
    for (size_t i{}; i < simplexTable.size(); ++i)
    {
        std::cout << simplexTable.at(i).first << '\t';
        for (size_t j{}; j < simplexTable.at(i).second.size();
++j)
        {
            std::cout << simplexTable.at(i).second.at(j) << '\t';
        }
        std::cout << '\n';
    }
    std::cout << '\n';
}

void outputMixedStrategyForPlayers(const SimplexTable
&simplexTable, const double &gamePrice, const
std::vector<std::string> &freeVarSecPlayer)
{
    std::cout << ",\tq = (";
    for (size_t i{}; i < freeVarSecPlayer.size(); ++i)
    {
        bool findVar{false};
        for (size_t j{}; !findVar && j < simplexTable.size() - 1;
++j)
        {
            if (freeVarSecPlayer.at(i) ==
simplexTable.at(j).first)
            {
                findVar = true;
                std::cout << simplexTable.at(j).second.at(0) *
gamePrice << ';';
            }
        }
    }
    std::cout << "\b),\t";
    std::cout << "p = (";
    for (size_t i{freeVarSecPlayer.size() + 1}; i <
simplexTable.at(0).second.size(); ++i)
    {
```

```
        std::cout << simplexTable.at(simplexTable.size() -
1).second.at(i) * gamePrice << ';';
    }
    std::cout << "\b)\n";
}

bool objFunctionHasNegative(const SimplexTable &simplexTable,
size_t &minNegativeIndex)
{
    size_t rowIndex{simplexTable.size() - 1};
    size_t minIndex{};
    bool findNegative{false};
    for (size_t i{1}; i < simplexTable.at(0).second.size(); ++i)
    {
        if (simplexTable.at(rowIndex).second.at(i) < 0)
        {
            findNegative = true;
            if (simplexTable.at(rowIndex).second.at(i) <
simplexTable.at(rowIndex).second.at(minIndex))
            {
                minIndex = i;
            }
        }
    }
    minNegativeIndex = minIndex;
    return findNegative;
}

bool exHasPositiveCoeff(const SimplexTable &simplexTable, const
size_t &colIndex, size_t &minCoeffIndex)
{
    size_t minCoeffIn{};
    double minCoeff{static_cast<double>(LONG_LONG_MAX)};
    bool findPositive{false};
    for (size_t i{}; i < simplexTable.size() - 1; ++i)
    {
        if (simplexTable.at(i).second.at(0) /
simplexTable.at(i).second.at(colIndex) > 0)
        {
            findPositive = true;
```

```
            const double coeff{simplexTable.at(i).second.at(0) /
simplexTable.at(i).second.at(colIndex)};
            if (coeff < minCoeff)
            {
                minCoeff = coeff;
                minCoeffIn = i;
            }
        }
    }
    minCoeffIndex = minCoeffIn;
    return findPositive;
}

double maxValueOfTheObjFunctionWithTableDisplay(SimplexTable
&simplexTable)
{
    size_t minIndex{};
    size_t minCoeffIndex{};
    while (objFunctionHasNegative(simplexTable, minIndex))
    {
        std::cout << "Simplex table :\n";
        outputSimplexTable(simplexTable);
        if (exHasPositiveCoeff(simplexTable, minIndex,
minCoeffIndex))
        {
            const double
divider{simplexTable.at(minCoeffIndex).second.at(minIndex)};
            for (size_t i{}; i <
simplexTable.at(minCoeffIndex).second.size(); ++i)
            {
                simplexTable.at(minCoeffIndex).second.at(i) /=
divider;
            }
            for (size_t i{}; i < simplexTable.size(); ++i)
            {
                if (i != minCoeffIndex)
                {
                    const double divide{-
simplexTable.at(i).second.at(minIndex) /
simplexTable.at(minCoeffIndex).second.at(minIndex)};
```

```
                    for (size_t j{}; j <
simplexTable.at(i).second.size(); ++j)
                    {
                        simplexTable.at(i).second.at(j) +=
(divide * simplexTable.at(minCoeffIndex).second.at(j));
                    }
                }
            }
            simplexTable.at(minCoeffIndex).first = "y" +
std::to_string(minIndex);
        }
        else
        {
            std::cout << "The problem does not have solution(The
objective function is unbounded on the range of admissible values
of solutions)";
            std::exit(1);
        }
    }
    std::cout << "Simplex table :\n";
    outputSimplexTable(simplexTable);
    return simplexTable.at(simplexTable.size() - 1).second.at(0);
}

int main(int argc, char **argv)
{
    size_t numberOfFreeVar{};
    std::cout << "Number of free variables = ";
    std::cin >> numberOfFreeVar;
    std::cout << "Enter variables that are free (In ascending
order of indices) : ";
    std::vector<std::string> freeVar(numberOfFreeVar);
    for (size_t i{}; i < numberOfFreeVar; ++i)
    {
        std::cin >> freeVar.at(i);
    }
    size_t numberOfRows{};
    size_t numberOfCols{};
    std::cout << "Number of rows in simplex table =  ";
    std::cin >> numberOfRows;
    std::cout << "Number of cols in simplex table = ";
```

```
    std::cin >> numberOfCols;
    std::cout << "Enter simplex table(with the names of basic 
variables) : \n";
    SimplexTable simplexTable(numberOfRows);
    for (size_t i{}; i < numberOfRows; ++i)
    {
        std::string basisVarName{};
        std::cin >> basisVarName;
        simplexTable.at(i).first = basisVarName;
        for (size_t j{}; j < numberOfCols; ++j)
        {
            double value{};
            std::cin >> value;
            simplexTable.at(i).second.push_back(value);
        }
    }

    std::cout << std::setprecision(2);
    double gamePrice{1 / 
maxValueOfTheObjFunctionWithTableDisplay(simplexTable)};
    std::cout << "u = v = " << gamePrice;
    outputMixedStrategyForPlayers(simplexTable, gamePrice, 
freeVar);
    return 0;
}
```

3.

Первый игрок:

$$z = v \rightarrow max$$

$$\begin{cases} 8p_1 + 7p_2 + 3p_3 + 5p_4 \geq v \\ 3p_1 + 6p_2 + 9p_3 + 2p_4 \geq v \\ 6p_1 + 5p_2 + 4p_3 + 10p_4 \geq v \\ p_1 + p_2 + p_3 + p_4 = 1 \end{cases}$$

$$p_i \geq 0, \left(i = \overline{1,4}\right)$$

Рассмотрим ограничение $p_1 + p_2 + p_3 + p_4 = 1$ и разделим обе его части на v, обозначив $x_i = \frac{p_i}{v} \left(i = \overline{1,4}\right)$. Поскольку в нашей задаче $\mathrm{v} \rightarrow max$, мы получим задачу на минимум для новой целевой функции:

$$z_1 = \frac{1}{v} = x_1 + x_2 + x_3 + x_4 \rightarrow min$$

Разделив все неравенства системы ограничений на v, получим для переменных x_i условия:

$$\begin{cases} 8x_1 + 7x_2 + 3x_3 + 5x_4 \geq 1 \\ 3x_1 + 6x_2 + 9x_3 + 2x_4 \geq 1 \\ 6x_1 + 5x_2 + 4x_3 + 10x_4 \geq 1 \end{cases}$$

$$x_i \geq 0, \left(i = \overline{1,4}\right)$$

Второй игрок:

$$f = u \rightarrow min$$

$$\begin{cases} 8q_1 + 3q_2 + 6q_3 \leq u \\ 7q_1 + 6q_2 + 5q_3 \leq u \\ 3q_1 + 9q_2 + 4q_3 \leq u \\ 5q_1 + 2q_2 + 10q_3 \leq u \\ q_1 + q_2 + q_3 = 1 \end{cases}$$

$$q_j \geq 0, \left(j = \overline{1,3}\right)$$

Рассмотрим ограничение $q_1 + q_2 + q_3 = 1$ и разделим обе его части на u, обозначив $y_j = \frac{q_j}{u} \left(j = \overline{1,3}\right)$. Поскольку в нашей задаче $u \rightarrow min$, мы получим задачу на максимум для новой целевой функции:

$$f_1 = \frac{1}{u} = y_1 + y_2 + y_3 \rightarrow max$$

Разделив все неравенства системы ограничений на u, получим для переменных y_j условия:

$$\begin{cases} 8y_1 + 3y_2 + 6y_3 \le 1 \\ 7y_1 + 6y_2 + 5y_3 \le 1 \\ 3y_1 + 9y_2 + 4y_3 \le 1 \\ 5y_1 + 2y_2 + 10y_3 \le 1 \end{cases}$$

$$y_j \ge 0, \left(j = \overline{1,3}\right)$$

Решаем задачу на max симплекс методом в чистом виде:

$$f_1 = y_1 + y_2 + y_3 \to max$$

$$\begin{cases} 8y_1 + 3y_2 + 6y_3 + y_4 = 1 \\ 7y_1 + 6y_2 + 5y_3 + y_5 = 1 \\ 3y_1 + 9y_2 + 4y_3 + y_6 = 1 \\ 5y_1 + 2y_2 + 10y_3 + y_7 = 1 \end{cases}$$

$$y_j \ge 0, \left(j = \overline{1,3}\right)$$

Таблица 1

Б	C	y1↓	y2	y3	y4	y5	y6	y7	Отн
←y4	1	8	3	6	1	0	0	0	1/8
y5	1	7	6	5	0	1	0	0	1/7
y6	1	3	9	4	0	0	1	0	1/3
y7	1	5	2	10	0	0	0	1	1/5
f1	0	-1	-1	-1	0	0	0	0	

Таблица 2

Б	C	y1	y2↓	y3	y4	y5	y6	y7	Отн
y1	1/8	1	3/8	3/4	1/8	0	0	0	1/3
←y5	1/8	0	3 3/8	- 1/4	- 7/8	1	0	0	1/27
y6	5/8	0	7 7/8	1 3/4	- 3/8	0	1	0	5/63
y7	3/8	0	1/8	6 1/4	- 5/8	0	0	1	3
f1	1/8	0	- 5/8	- 1/4	1/8	0	0	0	

Таблица 3

Б	C	y1	y2	y3↓	y4	y5	y6	y7	Отн
y1	1/9	1	0	7/9	2/9	- 1/9	0	0	1/7
y2	1/27	0	1	- 2/27	- 7/27	8/27	0	0	
y6	1/3	0	0	2 1/3	1 2/3	-2 1/3	1	0	1/7
←y7	10/27	0	0	6 7/27	- 16/27	- 1/27	0	1	10/169
f1	4/27	0	0	- 8/27	- 1/27	5/27	0	0	

Таблица 4

Б	C	y1	y2	y3	y4↓	y5	y6	y7	Отн
y1	11/169	1	0	0	50/169	- 18/169	0	- 21/169	11/50
y2	7/169	0	1	0	- 45/169	50/169	0	2/169	
←y6	33/169	0	0	0	1 150/169	-2 54/169	1	- 63/169	3/29
y3	10/169	0	0	1	- 16/169	- 1/169	0	27/169	
f1	28/169	0	0	0	- 11/169	31/169	0	8/169	

Таблица 5

Б	C	y1	y2	y3	y4	y5	y6	y7
y1	1/29	1	0	0	0	82/319	- 50/319	- 21/319
y2	2/29	0	1	0	0	- 10/319	45/319	- 13/319
y4	3/29	0	0	0	1	-1 73/319	169/319	- 63/319
y3	2/29	0	0	1	0	- 39/319	16/319	45/319
f1	5/29	0	0	0	0	3/29	1/29	1/29

$$f_{1max} = z_{1min} = \frac{5}{29}$$

$$u = v = \frac{1}{f_{1max}} = \frac{29}{5}$$

$$q_j = y_j * u$$

$$\vec{q}^0 = \left(\frac{1}{29} * \frac{29}{5}; \frac{2}{29} * \frac{29}{5}; \frac{2}{29} * \frac{29}{5}\right) = \left(\frac{1}{5}; \frac{2}{5}; \frac{2}{5}\right)$$

$$y_4 \to x_1$$

$$y_5 \to x_2$$

$y_6 \rightarrow x_3$

$y_7 \rightarrow x_4$

$p_i = x_i * v$

$$\vec{p}^0 = \left(0; \frac{3}{29} * \frac{29}{5}; \frac{1}{29} * \frac{29}{5}; \frac{1}{29} * \frac{29}{5}\right) = \left(0; \frac{3}{5}; \frac{1}{5}; \frac{1}{5}\right)$$

Тестовые данные:

```
PS D:\VS CODE CPlusPlus> .\rooster.exe
Number of free variables = 3
Enter variables that are free (In ascending order of indices) : y1 y2 y3
Number of rows in simplex table =  5
Number of cols in simplex table = 8
Enter simplex table(with the names of basic variables) :
y4 1 8 3 6 1 0 0 0
y5 1 7 6 5 0 1 0 0
y6 1 3 9 4 0 0 1 0
y7 1 5 2 10 0 0 0 1
f1 0 -1 -1 -1 0 0 0 0
Simplex table :
BV      FV      y1      y2      y3      y4      y5      y6      y7
y4      1       8       3       6       1       0       0       0
y5      1       7       6       5       0       1       0       0
y6      1       3       9       4       0       0       1       0
y7      1       5       2       10      0       0       0       1
f1      0       -1      -1      -1      0       0       0       0

Simplex table :
BV      FV      y1      y2      y3      y4      y5      y6      y7
y1      0.12    1       0.38    0.75    0.12    0       0       0
y5      0.12    0       3.4     -0.25   -0.88   1       0       0
y6      0.62    0       7.9     1.8     -0.38   0       1       0
y7      0.38    0       0.12    6.2     -0.62   0       0       1
f1      0.12    0       -0.62   -0.25   0.12    0       0       0

Simplex table :
BV      FV      y1      y2      y3      y4      y5      y6      y7
y1      0.11    1       0       0.78    0.22    -0.11   0       0
y2      0.037   0       1       -0.074  -0.26   0.3     0       0
y6      0.33    0       0       2.3     1.7     -2.3    1       0
y7      0.37    0       0       6.3     -0.59   -0.037  0       1
f1      0.15    0       0       -0.3    -0.037  0.19    0       0

Simplex table :
BV      FV      y1      y2      y3      y4      y5      y6      y7
y1      0.065   1       0       0       0.3     -0.11   0       -0.12
y2      0.041   0       1       0       -0.27   0.3     0       0.012
y6      0.2     0       0       0       1.9     -2.3    1       -0.37
y3      0.059   0       0       1       -0.095  -0.0059 0       0.16
f1      0.17    0       0       0       -0.065  0.18    0       0.047
```

```
Simplex table :
BV      FV      y1      y2      y3      y4      y5      y6      y7
y1      0.034   1       0       0       0       0.26    -0.16   -0.066
y2      0.069   0       1       0       0       -0.031  0.14    -0.041
y4      0.1     0       0       0       1       -1.2    0.53    -0.2
y3      0.069   0       0       1       0       -0.12   0.05    0.14
f1      0.17    0       0       0       0       0.1     0.034   0.034

u = v = 5.8,    q = (0.2;0.4;0.4),      p = (0;0.6;0.2;0.2)
PS D:\VS CODE CPlusPlus>
```

Результат, полученный при решении задачи «вручную», совпал с результатом, полученным программой.

Вывод: освоил метод нахождения седловой точки в смешанных стратегиях с помощью построения пары двойственных задач ЛП.