# СОДЕРЖАНИЕ

ВВЕДЕНИЕ 2

1. ОБЩАЯ ЧАСТЬ 3

1.1. Цель разработки 3

1.2. Анализ использования разработки 3

1.3. Анализ методов решения задачи 3

1.4. Анализ средств программирования 4

1.4.1. Обзор средств программирования 4

1.4.2. Характеристика программного обеспечения 4

1.4.3. Характеристика ПК 6

1.4.4. Характеристика языка программирования 6

2. СПЕЦИАЛЬНАЯ ЧАСТЬ 8

2.1. Постановка задачи 8

2.2. Экономико – математическая модель 12

2.3. Описания метода Гурвица 13

2.4. Алгоритм задачи 14

2.4.1. Алгоритм основной программы 14

2.4.2. Алгоритм процедуры W\_rezultat 15

2.5. Описание алгоритма 21

2.5.1. Описание алгоритма основной программы 21

2.5.2. Описания основной процедуры W\_rezultat расчета по методу Гурвица 21

2.6. Характеристика программы 24

2.7. Описание процесса отладки 24

2.8. Результаты решения задачи 25

2.9. Оценки результатов решения задачи 27

ЗАКЛЮЧЕНИЕ 28

СПИСОК ЛИТЕРАТУРЫ 29

Приложение 1 Текст программы 30

Приложение 2. Результаты работы программы 36

# ВВЕДЕНИЕ

В курсовом проекте представлена задача теория игр.

Одним из разделов теории игр являются состязательные задачи в условиях неопределённости. Состязательные задачи – это задачи, в которых сталкиваются интересы 2-х или более сторон, преследующих различные цели. Для решения этих задач используется методы теории игр. Для состязательных задач в условиях неопределённости в теории игр разработаны соответствующие принципы, на основании которых неопределенные ситуации преобразуется в детерминированные и решаются методом максимина.

Согласно принципу Гурвица неразумно, приняв во внимание самый маленький выигрыш, не учитывать самый большой, для чего необходимо ввести коэффициенты оптимизма (он выполняет роль вероятности). Значения оптимизма выбирают на основании субъективных соображений. В технических приложениях сложно выбрать коэффициент оптимизма, т.к. трудно найти количественную характеристику для тех долей оптимизма и пессимизма, которые присутствуют при принятии решения. Принцип Гурвица учитывает как пессимистический, так и оптимистический подход к ситуации.

# ОБЩАЯ ЧАСТЬ

## Цель разработки

Данный курсовой проект разрабатывался с целью вывести ситуацию из условия неопределённости; найти максимальный выигрыш, по которому определить оптимальную стратегию каждого игрока и игрока разрешающего конфликтную ситуацию.

## Анализ использования разработки

Данный курсовой проект предназначен для прогнозирования в учебных заведениях

## Анализ методов решения задачи

Для решения состязательных задач в условиях неопределённости в теории игр разработаны соответствующие принципы, на основании которых неопределенные ситуации преобразуются в детерминированные условия и решаются следующими методами: метод максимина, метод минимакса, принцип Гурвица, метод Сэвиджа, метод Бейеса-Лапласа. В данном курсовом проекте использовался принцип Гурвица.

## Анализ средств программирования

### Обзор средств программирования

Данную программу можно написать на различный языках программи­рования : Си, Бейсик, Ассемблер т. д. В данной случае выбран язык Object Pascal(доработанная разновидность языка Pascal), т.к. он отличен от Бейсика и Ассемблера тем, что среда разработки Delphi имеет широкие возможности визуальной разработки (разработчик во время разработки может видеть интерфейс своей программы и размещать на ней компоненты ) и интеллектуальную систему отладки. Язык Ассемблер является неудобным в данном случае, а в языке Бейсик ограничены возможности применения процедур и функций, При написании программы на языке Object Pascal используются все возможности модульности программ, процедуры и функции.

### Характеристика программного обеспечения

Операционная система – это совокупность программ, обеспечивающих управление аппаратной частью компьютера и прикладными программами, а также взаимодействие между собой и пользователями.

Основная функция ОС является её способность управлять устройствами памяти на магнитных дисках.

Операционная система MS-DOS состоит из следующих частей: базовой системы ввода/вывода, загрузчика операционной системы, дисковых файлов IO.SYS и MS-DOS.SYS.

В настоящее время существуют более современные ОС, с гораздо большим набором возможностей. Это ОС MS Windows’ 95/98/2000/Me, OS/2.

Особенность среды MS Windows

* Стандартизация интерфейса пользователя.
* Оптимальное управление оперативной памятью объёмом в несколько гигабайт.
* Поддержка подключаемых устройств.
* Интеграция функций программ.
* Многозадачность
* Использование графического интерфейса с оконной системой организации.

ОС Windows выполняет следующих основные функции :

* Управление файловой системой носителей информации ( отображение, изменение, создание, перемещение, удаление, переименование ).
* Запуск и завершение прикладных программ.
* Предоставление сервисов ( всевозможные настройки, оптимизация работы).
* Управление устройствами и BIOS’ом .

Ядро Windows и ее Функции зависят от состава аппаратный средств, работа с которыми осуществляется с помощью драйверов и BIOS’а.

Назначение BIOS - базовой системы ввода / вывода - состоит в выполнении наиболее простых и универсальный услуг ОС, связанный с осуществлением ввода / вывода и прием символов с клавиатуры, анализ принятой информации, выдача символов на принтер и т.д. BIOS со­держит также тест функционирования компьютера, проверяющий работу памяти и устройств компьютера при включении его электропитания.

Драйверы устройств предназначены для организации обмена данны­ми между внутренней и внешней памятью. Драйверы включаются в работу в тот момент, когда в него из BDOS поступает команда по поиску нужных данных. Но перед этим BIOS настраивает драйвер на работу по включению соответствующего устройства. Физическая связь организуется через контроллер.

### Характеристика ПК

В ПК, на котором разрабатывался курсовой проект можно выделить основные части:

Монитор: Samsung Sync Master 550 (M)S, разрешающая максимальная способность 1024x768 точек на дюйм и частота обновления 75 Гц;

MB: Intel T810B-S ;

Процессор: Intel Celeron 366 (Genuine Intel Семейство 6 Модель 366 MГц Модификация 5);

SDRAM: Samsung 64Mb

HDD:4,2 Gb Quantum;

Дисководы: FDD 1,44Mb, CD-ROM 40x ;

Принтер: Epson FX-1000;

Клавиатура: Расширенная клавиатура PC/AT ( Windows’98 107 клавиш);

Мышь: Стандартная мышь для COM-порта;

### Характеристика языка программирования

В настоящее время наиболее распространенными алгоритмическими языками является Паскаль, Си.

Язык Паскаль был разработан в конце 80-х гадах профессором Н. Виртом. Своё название получил в честь французского математика и философа Б. Паскаля. Язык был создан специально для обучения программированию.

Последняя версия Object Pascal позволила объединить в рамкак единой системы мощный алго­ритмический потенциал языка, методы объектно-ориентированного программирования, современную графику, удобные средства тестирова­ния и отладки программы, а также обеспечить дружественный интерфейс с пользователем.

Основные операторы языка являются хорошей иллюстрацией базовый управляющий конструкций структурного программирования.

Большую помощь программистам оказывает библиотека стандартных подпрограмм Паскаля. Эта библиотека модернизируется и пополняет­ся уже более десяти лет, В нее входят средства для работы с оператив­ной и внешней памятью, клавиатурой, дисплеем и другими внешними устройствами ПЭВМ.

Графический пакет системы программирования Delphi - один из самый мощных пакетов такого типа, т.к. позволяет использовать все функции граф. библиотек OpenGL и Direct3D.

Система программирования Delphi работает по модульному принципу программирования, который лежит в основе всех современных технологий разработки программ, Программа, написанная на Delphi разбита на модули, а те, в свою очередь, состоят из подпрограмм.

Среда в системе программирования Delphi многооконная, на экране дисплея одновременно присутствуют несколько окон ре­дактирования, панель компонент, инспектор объектов, редакторы форм и т, д.

# СПЕЦИАЛЬНАЯ ЧАСТЬ

## Постановка задачи

Теория игр – теория математических модулей, интересы участников которых различны, причём они достигают своих целей различными путями.

Столкновение противоположных интересов участников приводит к возникновению конфликтных ситуаций. Чтобы исключить трудности, возникающие при анализе конфликтных ситуаций, строится упрощенная модель ситуаций. Такая модель называется игрой. Теория игр относится к теории статистических решений.

В задачах теории игр предполагалось, что в них примут участие две стороны, интересы которых противоположны. Поэтому действия каждой стороны направлены на увеличения выигрыша. Но во многих задачах, приводящих к игровым, неопределенность вызвана отсутствием информации об условиях, в которых осуществляется действие. Эти условия зависят не от сознательных действий другого игрока, а от объективной действительности, которую принято называть природой.

Игру с природой описывается с помощью платёжной матрицы, в которой в качестве игрока А выступает статистик (человек, который принимает решения), имеющий m возможных стратегий А1, А2, …, Аm, а в качестве второго игрока выступает природа.

План, по которому игрок совершает выбор в каждой возможной ситуации и при каждой возможной фактической информации называется стратегий игрока.

Главным в исследовании теории игр является выбор оптимальных стратегий игроков. Стратегия игрока является оптимальной, если применение этой стратегии обеспечит ему наибольший гарантированный выигрыш при всевозможных стратегиях другого игрока. В процессе одной игры каждый из игроков выбирает одну стратеги. Стратегии делятся на чистые и смешанные.

Чистая стратегия – это стратегия, имеющая одно единственное значение или решение из множества заданных.

Смешанная (сложная) стратегия – это стратегия, которая берёт m значений с соответствующими вероятностями.

Стороны участвующие в конфликтной ситуации называются игроками, а предполагаемые действия каждого из игроков, направленные на достижение некоторой цели, называется правилами игры.

Платёж – это количественная оценка результатов игры.

Ходом в теории игр называется выбор одного из предложенных правилами игры действий его осуществлении.

Состязательная задача – это задача, разрешающая конфликтные ситуации между двумя или более противниками с целью нахождения оптимальной стратегии для каждого игрока, и в конечном итоге игрока, разрешающего конфликтную ситуацию.

Игру двух игроков можно описать как производственный процесс с помощью следующей функциональной схемы (рис.1).

**Рисунок 2.1.1**

Игрок А

Игрок В

Стратегии

игрок А

Стратегии

игрок В

УС

СУ

ОУ

f(t)

U(t)

Оба игрока по прямой связи U(t) делает ход, выбирая предполагаемую стратегию. Ни один из игроков не знает хода противника. В случае если игрок узнает стратегию своего противника, то по обратной связи f(t) поступает сигнал, что он может отказаться от своей старой стратегии и выбрать другую стратегию. Востановив работу по прямой связи U(t).

Человек А в играх с природой старается действовать осмотрительно, используя, например, минимаксную стратегию, позволяющую получить наименьший проигрыш. Второй игрок В (природа) действует совершенно случайно, возможные стратегии определяются как её состояние. Условия игры задаются в виде матрицы.



Элементы Сij = выигрышу игрока А, если он использует стратегию Аi.

В данном курсовом проекте состязательная задача решается по методу Гурвица.

Пусть в игре принимают участие два игрока А и В.

Рассматривается конфликтная ситуация между двумя сторонами А и В. Игрок А имеет m стратегий, а В имеет n стратегий: А={А1, А1,…, А1}; В={В1, В1,…, В1}.

Взаимосвязь между стратегиями любого из игроков определяется платёжной матрицей С={Cij}m\*n. Cij – выигрыш игрока А. Заданы статистические коэффициенты оптимизации ().



Цель игры состоит в том, чтобы вывести ситуацию из условия неопределённости, найти максимальный выигрыш, по которому определить оптимальную стратегию каждого игрока, а также игрока разрешающего конфликтную ситуацию.

Решение игры и исходные данные сводятся в таблицу Гурвица (табл. 2.1.1).

**Таблица 2.1.1**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | В1 | В2 | … | Вn | Наименьший  выигрыш | Наибольший  выигрыш | Коэффициенты оптимизма | | |
| 1 | … | k |
| А1 | C11 | C12 | … | C1n | a1 | А`1 | V11 | … | V1k |
| А2 | C21 | C22 | … | C2n | a 2 | А`2 | V21 | … | V2k |
| … | … | … | … | … | … | … | … | … | … |
| Аm | Cm1 | Cm2 | … | Cmn | a m | А`m | Vm1 | … | Vmk |

Где j – статистические коэффициенты оптимизации;



к – количество оптимизмов;

Аj – стратегии игрока А;

Вj - стратегии игрока В;

Vij – расчетные условные выигрыши;



С учётом коэффициентом оптимизма вычисляем условные выигрыши



Выбираем решение о выборе стратегии, при , где 0 (для игрок переходит к стратегии «азартного игрока»; для - стратегия абсолютного оптимизма).



.

## Экономико – математическая модель

Основная теорема теории игр, состоит в следующем: любая конечная игра имеет, по крайне мере, одно решение, возможно в области смешанных стратегий. Применение оптимальной стратегии позволяет получить выигрыш равный цене игры: , – цена игры.



Применение игроком А оптимальной стратегии должно обеспечивать ему выигрыш при любых действиях игрока В, не меньше цены . Выполняется соотношение:



, - вероятность использования стратегии игрока А.



Аналогично, для игрока В оптимальная стратегия должна обеспечить при любых стратегиях игрока А проигрыш, не более :



, - вероятность использования стратегии игрока В.



Задача имеет решение игры, если её матрицы не содержит седловой точки ().



Расчет выигрышей производится по целевой функции:



Система ограничения:

## 



## Описания метода Гурвица

### Выбираем по строкам наименьший выигрыш и заполняем колонку а.

### Выбираем по строкам наибольший выигрыши и заполняем колонку



### Производим расчёт выигрыша по формуле: ; результаты заносим в таблицу и получаем матрицу .



### По методу максимина определяется наибольший из всех расчётных выигрышей; по наибольшему значению определяется стратегия данного игрока.



### Для разрешения конфликтной ситуации составляется таблица Гурвица относительно игрока В. В таблице меняем платёжную матрицу.

### Далее также применяем принцип Гурвица и метод максимина относительно игрока В.

### Игрок, разрешающий конфликтную ситуацию определяется по наибольшему расчётному выигрышу из соответствующих оптимальных стратегий игроков.

## Алгоритм задачи

### Алгоритм основной программы

НАЧАЛО

1

Wwod\_koef

W=?

W\_rezultat

W\_tabliza\_A

W\_tabliza\_B

КОНЕЦ

W=1

W=2

2

3

4

5

7

6

рисунок 2.4.1

### Алгоритм процедуры W\_rezultat

ВХОД

1

Цикл 1



Цикл 2



С\_А:=C\_S



Цикл 2

Цикл 1

Цикл 3



Цикл 4



С\_B:=C\_S



Цикл 4

2

3

4

5

6

7

8

9

10

к стр.

рис 2.4.2

Цикл 3

Цикл 5



a\_m:=C\_a



a\_b:=C\_a



Цикл 6



11

12

a\_m:=C\_ a



13

14

15

16

17

да

C\_ a[i,j]< a\_m[i,j]

a\_m:=C\_ a



18

19

да

C\_ a[i,j]> a\_b[i,j]

Цикл 6

20

Цикл 7



21

из стр.

к стр.

**Продолжение рис. 2.4.2**

Цикл 7

23

V\_a[i,j]:=al[j]\*a\_m[i]+ +(1-al[j]) \*a\_b[i]

Цикл 6

24

max\_a:=V\_a[1,1]

H\_A:=1

Цикл 9



Цикл 10



27

28

25

26

Max\_a:=V\_ a



29

30

да

V\_a[i,j]>max\_a

H\_a:=



31

22

из стр.

к стр.

**Продолжение рис. 2.4.2**

Цикл 10

32

Цикл 9

33

Цикл 11



b\_m:=C\_b



b\_b:=C\_b



Цикл 12



34

b\_m:=C\_ b



35

36

37

38

39

да

C\_ b[i,j]< b\_m[i,j]

b\_m:=C\_ b



40

41

да

C\_ b[i,j]> b\_b[i,j]

Цикл 12

42

из стр.

к стр.

**Продолжение рис. 2.4.2**

Цикл 13



43

Цикл 13

45

V\_b[i,j]:=al[j]\*b\_m[i]++(1-al[j])\*b\_b[i]

Цикл 11

46

max\_b:=V\_b[1,1]

H\_B:=1

Цикл 14



Цикл 15



49

50

47

48

Max\_b:=V\_b



51

52

да

V\_b[i,j]>max\_b

44

из стр.

к стр.

к стр.

**Продолжение рис. 2.4.2**

Выход

Седловые точки

H\_b:=



53

Цикл 15

54

Цикл 14

55

57

да

max\_a=max\_b

max\_a>max\_b

58

56

Игрока А разрешит конфликтную ситуацию

да

Нет

59

Нет

Игрока В разрешит конфликтную ситуацию

60

Оптимальная стратегия, max выигрыш игрока А

61

Оптимальная стратегия, max выигрыш игрока В

62

63

из стр.

из стр.

**Продолжение рис 2.4.2**

## Описание алгоритма

### Описание алгоритма основной программы

1. Начало программы
2. Процедура ввод статистических коэффициентов оптимизации
3. Основная процедура расчета по методу Гурвица
4. Оператор вывода расчетных таблиц
5. Процедура вывода расчетной таблицы и платежной матрицы игрока А
6. Процедура вывода расчетной таблицы и платежной матрицы игрока В
7. Конец программы

### Описания основной процедуры W\_rezultat расчета по методу Гурвица

1. Вход в процедуру
2. Начало цикла i от 1 до m
3. Начало цикла j от 1 до n
4. Преобразования символа строки из ячейки таблицы C\_S в целое число матрицы C\_a
5. Конец цикла по j
6. Конец цикла по I
7. Начало цикла i от 1 до n
8. Начало цикла j от 1 до m
9. Преобразования символа строки из ячейки таблицы C\_S в целое число матрицы С\_b
10. Конец цикла по j
11. Конец цикла по I
12. Начало цикла i от 1 до m
13. Массиву a\_m (наименьшие выигрыши)присваивается первый элемент i строки матрицы С\_a (игрока А)
14. Массиву a\_b (наибольшие выигрыши)присваивается первый элемент i строки матрицы С\_a (игрока А)
15. Начало цикла j от 2 до n
16. Проверка условия на нахождения минимального элемента
17. Нахождения минимального элемента
18. Проверка условия на нахождения максимально элемента
19. Нахождения максимально элемента
20. Конец цикла по j
21. Начало цикла j от 1 до k
22. Расчет условно расчетных выигрышей (игрока А)
23. Конец цикла по j
24. Конец цикла по i
25. Максимальному выигрышу max\_a присваивается первый элемент первой строки матрицы условно расчетных выигрышей (игрока А)
26. Оптимальной стратегии H\_a присваивается первая стратегия (игрока А)
27. Начало цикла i от 1 до m
28. Начало цикла j от 1 до k
29. Проверка условия на нахождения максимально выигрыша
30. Нахождения максимально выигрыша
31. Нахождения оптимальной стратегии
32. Конец цикла по j (игрока А)
33. Конец цикла по I (игрока А)
34. Начало цикла i от 1 до n
35. Массиву b\_m (наименьшие выигрыши)присваивается первый элемент i строки матрицы С\_b (игрока В)
36. Массиву b\_b (наибольшие выигрыши)присваивается первый элемент i строки матрицы С\_b (игрока В)

Блок 38 - Проверка условия на нахождения минимального элемента

Блок 39 - Нахождения минимального элемента

Блок 40 - Проверка условия на нахождения максимально элемента

Блок 41 - Нахождения максимально элемента

Блок 44 - Расчет условно расчетных выигрышей (игрока В)

Блок 47 - Максимальному выигрышу max\_b присваивается первый элемент первой строки матрицы условно расчетных выигрышей (игрока B)

Блок 48 - Оптимальной стратегии H\_b присваивается первая стратегия (игрока B)

Блок 51 - Проверка условия на нахождения максимально выигрыша

Блок 52 - Нахождения максимально выигрыша

Блок 53 - Нахождения оптимальной стратегии

Блок 56 - Проверка условия на наличие седловых точек

Блок 58 - Проверка условия на нахождения игрока, разрешающего конфликтную ситуацию

Блок 59 - Вывод игрока А разрешивший конфликтную ситуацию

Блок 60 - Вывод игрока В разрешивший конфликтную ситуацию

Блок 61 - Вывод оптимальной стратегии и набольшего выигрыша игрока А

Блок 62 - Вывод оптимальной стратегии и набольшего выигрыша игрока В

Блок 63 - Выход из процедуры

## Характеристика программы

Программа написана на языке Object Pascal. Она занимает 44,8 Кб оперативной памяти, место на жестком диске 498 Кб. Программа была реализована на компьютере Intel Celeron 366 с помощью OC Windows’ 98, в среде программирования Delphi версия 5.0.

Выходными данными является платёжная матрица, состоящая из вещественных чисел и коэффициенты оптимизма – целые числа. Эти данные будут вводиться пользователем с клавиатуры и идентифицироваться в окне на экране монитора.

Выходными данными будет расчетные таблицы для игроков А и В, максимальный выигрыш, оптимальная стратегия каждого из игроков, а также будет выведен игрок, разрешающий конфликтную ситуацию.

## Описание процесса отладки

Под отладкой понимается процесс поиска и устранения ошибок в программе. Ошибки, которые могут быть в программе, принято делить на три группы: синтаксические ошибки; ошибки времени выполнения; алгоритмические ошибки.

В среде Delphi мощный встроенный отладчик, значительно упрощающий отладку программ.

Основными инструментами отладки является точки контрольного останова и окно наблюдения за переменными. Если программа запущена из среды Delphi, ее работу можно прервать в любой момент или установив точку контрольного останова в той части программы, которая выполняется в данный момент или будет выполнена.

После контрольного останова в окне наблюдения отображаются текущие значения наблюдаемых объектов. Кроме того, можно увидеть текущее значение любой переменной, если в окне редактора укажете на нее мышью.

Для написания моей программы использовался метод тестирования. Метод тестирования основан на обдумываний и заключается в использования тестов. Существую два типа тестов: тесты для тестирования целью которых является обнаруживания заранее не определенной ошибки и тесты для отладки, цель которых обеспечить информации полезной для выявления место нахождения подозреваемой ошибки.

## Результаты решения задачи

Разрешить конфликтную ситуацию двух игроков А и В заданную в неопределенных условиях с статистические коэффициентами оптимизации =0,1; =0,2; =0,3.



Исходные данные и решения задачи сводится в таблицу 2.8.1.

**Таблица 2.8.1**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | В1 | В2 | В3 | Наименьший  выигрыш | Наибольший  выигрыш | Коэффициенты оптимизма | | |
| 0,1 | 0,2 | 0,3 |
| А1 | 1 | 1 | 3 | 1 | 3 | 2,8 | 2,6 | 2,4 |
| А2 | 5 | 6 | 8 | 5 | 8 | 7,7 | 7,4 | 7,1 |
| А3 | 4 | 3 | 5 | 3 | 5 | 4,8 | 4,6 | 4,4 |

Найти игрока, разрешающего конфликтную ситуацию.

Найдём условно расчётные выигрыши игрока А по формуле:



V11=0,1\*1+(1 – 0,1)\*3=2,8

V12=0,2\*1+(1 – 0,2)\*3=2,6

V13=0,3\*1+(1 – 0,3)\*3=2,4

V21=0,1\*5+(1 – 0,1)\*8=7,7

V22=0,2\*5+(1 – 0,2)\*8=7,4

V23=0,3\*5+(1 – 0,3)\*8=7,1

V31=0,1\*3+(1 – 0,1)\*5=4,8

V32=0,2\*3+(1 – 0,2)\*5=4,6

V33=0,3\*3+(1 – 0,3)\*5=4,4

Среди найденных условных расчётных выигрышей найдём максимальный. Он равен 7.7, значит оптимальная стратегия игрока А будет А2.

Далее найдём оптимальная стратегия игрока В, для этого транспонируем матрицу. Результаты заносим в таблицу 2.8.2.

Таблица 2.8.2

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | А1 | А2 | А3 | Наименьший  выигрыш | Наибольший  выигрыш | Коэффициенты оптимизма | | |
| 0,1 | 0,2 | 0,3 |
| В1 | 1 | 5 | 4 | 1 | 5 | 4,6 | 4,2 | 3,8 |
| В2 | 1 | 6 | 3 | 1 | 6 | 5,5 | 5 | 4,5 |
| В3 | 3 | 8 | 5 | 3 | 8 | 7,5 | 7 | 6,5 |

Найдём условно расчётные выигрыши игрока В

V11=0,1\*1+(1 – 0,1)\*5=4,6

V12=0,2\*1+(1 – 0,2)\*5=4,2

V13=0,3\*1+(1 – 0,3)\*5=3,8

V21=0,1\*1+(1 – 0,1)\*6=5,5

V22=0,2\*1+(1 – 0,2)\*6=5

V23=0,3\*1+(1 – 0,3)\*6=4,5

V31=0,1\*3+(1 – 0,1)\*8=7,5

V32=0,2\*3+(1 – 0,2)\*8=7

V33=0,3\*3+(1 – 0,3)\*8=6,5

Среди найденных условных расчётных выигрышей найдём максимальный. Он равен 7.5, значит оптимальная стратегия игрока В будет В3.

Из 2-х оптимальных стратегий, находим наибольший выигрыш, а именно 7,7>7,5; следовательно игрок А разрешит конфликтную ситуацию с максимальным выигрышем равным 7,7, стратегия которого равна 2.

## Оценки результатов решения задачи

Результат решения задачи полностью соответствует заданию курсового проекта. В сравнении результатов решения задачи ручным с результатами автоматизированным методом, получил одинаковые результаты. Что означает что программа работает верно. Преимущество автоматизированного метода над ручным состоит в том, что автоматизированное время выполнения программы меньше, чем ручным.

# ЗАКЛЮЧЕНИЕ

Данная курсовая работа включает в себя два предмета: «Программирование» и «Компьютерное модулирование»

В курсовой работе были рассмотрены следующие вопросы:

* Рассмотрена характеристика «Теории игр» и следующие методы ее решения: метод Гурвица, метод Сэвиджа, метод максимина.
* Рассмотрен и дан алгоритм решения теории игры в условии неопределенности методом Гурвица.
* Дана краткая характеристика ПК, включая анализ средств программирования, описания ОС MS-DOS и MS Windows’
* Рассмотрен выбор языка программирования.
* Написана программа для решения данной задачи.

# СПИСОК ЛИТЕРАТУРЫ

* 1. Г. С. Малик «Основы экономики и математические методы в планировании».
  2. Кузнецов «Математическое программирование».
  3. В. В. Фаронов «Delphi 5. Учебный курс».
  4. Ю. П. Зайченко «Исследование операций в задачах, алгоритмах, программах».

# Приложение 1 Текст программы

Medot\_Gurwiwiza.dpr

program Medot\_Gurwiza;

{*Курсовой проект по предмету "Компьютерное модулирование" по теме "Теория игр"*

*Принцип Гурвица Выполнил студент гр. П-00-1 Юшков Андрей 10.06.02*}

uses

Forms,

osnowa in 'osnowa.pas' {form1},

Unit2 in 'Unit2.pas' {Form2};

{$R \*.RES}

begin

Application.Initialize;

Application.CreateForm(Tform1, form1);

Application.CreateForm(TForm2, Form2);

Application.Run;

end.

unit osnowa;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

Grids, StdCtrls, ToolWin, ComCtrls, Buttons, ActnList, StdActns, Menus,

Mask, ExtCtrls, jpeg;

type

Tform1 = class(TForm)

tabliza: TStringGrid;

Panel1: TPanel;

Button1: TButton;

Edit1: TEdit;

Edit2: TEdit;

Edit3: TEdit;

Label2: TLabel;

Label3: TLabel;

Label4: TLabel;

C\_S: TStringGrid;

Panel2: TPanel;

Label5: TLabel;

Label6: TLabel;

Label7: TLabel;

Label8: TLabel;

Label9: TLabel;

Label10: TLabel;

Label11: TLabel;

Label12: TLabel;

Label13: TLabel;

Label14: TLabel;

Panel3: TPanel;

Panel4: TPanel;

Label17: TLabel;

Label18: TLabel;

Panel5: TPanel;

Label19: TLabel;

Label20: TLabel;

Label21: TLabel;

Label22: TLabel;

Label23: TLabel;

RadioButton7: TRadioButton;

RadioButton8: TRadioButton;

Button3: TButton;

Panel6: TPanel;

Label1: TLabel;

BitBtn1: TBitBtn;

Label15: TLabel;

procedure WWod\_koef(Sender: TObject);

procedure W\_Rezultat(Sender: TObject);

procedure W\_tabliza\_A(Sender: TObject);

procedure W\_tabliza\_B(Sender: TObject);

private

{ Private declarations }

public

{ Public declarations }

end;

var

form1: Tform1;

C\_B,C\_A:array [1..10,1..10] of integer; { *платёжная матрица игрока А,В}*

a\_b,a\_m,b\_b,b\_m:array[1..10] of integer; {*наибольший наименьший выигрыш иг. А,В}*

al:array[1..10] of real; {м*ассив альфа}*

V\_A,V\_B:array[1..10,1..10]of real; {Р*асчётные выигрыши иг. А,В }*

max\_a:real; { *Наибольший выигрыш игрока А}*

max\_b:real; { *Наибольший выигрыш игрока В}*

H\_a:integer; { О*птимальная стратегия игрока А}*

h\_b:integer; { *Оптимальная стратегия игрока В}*

m:Integer; { *Количество стратегий игрока А}*

n:Integer; { *Количество стратегий игрока В}*

k:Integer; { *Количество статистических коэффициентов}*

I,J:Integer;

implementation

uses Unit2;

*{$R \*.DFM}*

*{ вывод коэф., матрицы С\_А}*

procedure WW\_A;

begin

form1.c\_s.Colcount:=n+1;

form1.c\_s.Rowcount:=m+1;

form1.tabliza.Rowcount:=m+1;

for i :=1 to m do

begin

form1.tabliza.Cells[0,i]:='A'+intToStr(i);

form1.C\_S.Cells[0,i]:='A'+intToStr(i);

for j :=1 to n do

begin

form1.C\_S.Cells[j,0]:='B'+intToStr(j);

form1.C\_S.Cells[j,i]:=intToStr(C\_A[i,j]);

end;

end;

with form1 do

begin

label23.caption:='A';

tabliza.cells[1,0]:='a\_малая';tabliza.cells[2,0]:='a\_большая';

end;

end;

{ *Вывод наибольший, наименьший, расчётный выигрыш матрицы V\_А*}

procedure WW\_A1;

begin

WW\_A;

With form1.tabliza Do

begin

for j:=1 to n do

begin

cells[1,j]:=intToStr(a\_m[j]);

cells[2,j]:=intToStr(a\_b[j]);

end;

for i:=1 to m do

for j:=1 to k do

cells[j+2,i]:=floatToStr(V\_a[i,j]);

end;

end;

{*событие на нажатие кнопки 'Ввод коэф..*'}

procedure TForm1.WWod\_koef(Sender: TObject);

begin

try

m:=strToint(edit1.text);

n:=strToint(edit2.text);

k:=strToint(edit3.text);

except

showMessage('Ошибочная запись числа ');

end;

try

Form2 := TForm2.Create(self);

tabliza.Colcount:=3+k;

Form2.ShowModal;

finally

Form2.Close;

WW\_a;

end;

end;

{*событие на нажатие кнопки 'вывод результата'*}

procedure Tform1.W\_Rezultat(Sender: TObject);

begin

Panel6.Visible:=false;

panel3.Visible:=true;

{*Вводим из таблицы C\_A в матрицу игрока А - C\_A} { C\_S[столбец,строка]* }

for i :=1 to m do {*по столбцам m таблицы C\_S}*

for j :=1 to n do {*по строкам n таблицы C\_S}*

C\_a[i,j]:=StrToInt(C\_S.Cells[j,i]);

{*Создаём матрицу C\_B путём транспонирования матрицы игрока А*}

for i :=1 to n do

for j :=1 to m do

C\_b[i,j] :=StrToInt(C\_S.Cells[i,j]);

{*расчет наименьших и наибольших выигрышей игрока A*}

for i:=1 to m do

begin

a\_m[i]:=C\_a[i,1]; {*массив наименьшии выигрыш}*

a\_b[i]:=C\_a[i,1]; {*массив наибольшии выигрыш}*

for j :=2 to n do

begin

if C\_a[i,j]<a\_m[i] then a\_m[i]:=C\_a[i,j];

if C\_a[i,j]>a\_b[i] then a\_b[i]:=C\_a[i,j];

end;

*{вычисления расчетных выигрышей игрока A}*

for j :=1 to k do

V\_a[i,j]:=al[j]\*a\_m[i]+(1-al[j])\*a\_b[i];

end;

*{нахождения оптимальной стратегии и максимального выигрыша игрока A}*

max\_a:=V\_a[1,1];

H\_A:=1;

for i :=1 to m do

for j :=1 to k do

if V\_a[i,j]>max\_A then

begin

max\_a:=V\_a[i,j]; *{ максимальный выигрыш игрока А}*

H\_a:=i { *оптимальная стратегия игрока А}*

end;

*{расчет наименьших и наибольших выигрышей игрока В}*

for i:=1 to n do

begin

b\_m[i]:=C\_b[i,1]; {*массив наименьшии выигрыш*}

b\_b[i]:=C\_b[i,1]; {*массив наибольшии выигрыш}*

for j:=2 to m do

begin

if C\_b[i,j]<b\_m[i] then b\_m[i]:=C\_b[i,j];

if C\_b[i,j]>b\_b[i] then b\_b[i]:=C\_b[i,j];

end;

{*вычисления расчетных выигрышей игрока В*}

for j:=1 to k do

V\_b[i,j]:=al[j]\*b\_m[i]+(1-al[j])\*b\_b[i];

end;

{*нахождения оптимальной стратегии и максимального выигрыша игрока В*}

max\_b:=V\_b[1,1];

H\_b:=1;

for i:=1 to n do

for j:=1 to k do

if V\_b[i,j]>max\_b then

begin

max\_b:=V\_b[i,j]; { *максимальный выигрыш игрока B}*

H\_b:=i { о*птимальная стратегия игрока B}*

end;

*{ нахождения наибольшего расчетного выигрыша одного из игроков* }

if max\_a=max\_b then Panel6.Visible:=true

else

if max\_a>max\_b then

begin

Panel4.Visible:=true;

panel5.Visible:=false

end

else

begin

panel5.Visible:=true;

Panel4.Visible:=false

end;

label11.Caption:=FloatToStr(max\_a);

label12.Caption:=FloatToStr(H\_a);

label14.Caption:=FloatToStr(max\_b);

label13.Caption:=FloatToStr(H\_b);

WW\_A1;

end;

*{просмотр для игрока А}*

procedure Tform1.W\_tabliza\_A(Sender: TObject);

begin

WW\_A1;

end;

{*просмотр для игрока B}*

procedure Tform1.W\_tabliza\_B(Sender: TObject);

begin

with form1 do

Begin

c\_s.Colcount:=m+1;

c\_s.Rowcount:=n+1;

tabliza.Rowcount:=n+1;

for i:=1 to n do

begin

form1.tabliza.Cells[0,i]:='B'+intToStr(i);

form1.C\_S.Cells[0,i]:='B'+intToStr(i);

for j:=1 to m do

begin

form1.C\_S.Cells[j,0]:='A'+intToStr(j);

form1.C\_S.Cells[j,i]:=intToStr(C\_B[i,j]);

end;

end;

label23.caption:='B';

tabliza.cells[1,0]:='b\_малая';tabliza.cells[2,0]:='b\_большая';

for j:=1 to n do

begin

tabliza.cells[1,j]:=intToStr(b\_m[j]);

tabliza.cells[2,j]:=intToStr(b\_b[j]);

end;

for i:=1 to n do

for j:=1 to k do

tabliza.cells[j+2,i]:=floatToStr(V\_b[i,j]);

end;

end;

end.

unit Unit2;

interface

uses

Windows, Messages, SysUtils, Classes, Graphics, Controls, Forms, Dialogs,

Grids, StdCtrls, ExtCtrls, Buttons, Menus;

type

TForm2 = class(TForm)

alpfa: TStringGrid;

Panel1: TPanel;

BitBtn1: TBitBtn;

BitBtn2: TBitBtn;

procedure FormShow(Sender: TObject);

procedure BitBtn2Click(Sender: TObject);

private

*{ Private declarations* }

public

*{ Public declarations }*

end;

var

Form2: TForm2;

i,j:integer;

implementation

uses osnowa;

*{$R \*.DFM}*

*{ Ввод козффициентов оптимизмов}*

procedure TForm2.FormShow(Sender: TObject);

begin

j:=0;

form1.tabliza.Visible:=true;

alpfa.Colcount:=strToInt(form1.edit3.text);

for i:=0 to alpfa.Colcount do

begin

j:=j+1;

alpfa.Cells[i,0]:='Alpha'+intToStr(i+1);

alpfa.Cells[i,1]:=FloatToStr(al[j]);

end;

end;

procedure TForm2.BitBtn2Click(Sender: TObject);

begin

j:=0;

for i:=0 to alpfa.Colcount do

begin

j:=j+1;

try

al[j]:=strToFloat(trim(alpfa.Cells[i,1]));

form1.tabliza.Cells[3+i,0]:=alpfa.Cells[i,1];

except

showMessage('Ошибочная запись числа : '+alpfa.Cells[i,1]);

end; end;

end;

end.

# Приложение 2 Результат работы программы

