

**Московский Энергетический Институт
(Технический Университет)**

Кафедра Прикладной Математики

**Лабораторная работа №1
по дисциплине
«Теория игр и исследование операций»**

**Проектирование с помощью инструментального
комплекса СИМПР**

**Выполнила: студентка группы А-13-01
Калашникова Надежда**

**Преподаватели: Гречкина П. В.,
Еремеев А. П.**

Москва, 2005

Содержание

Постановка задачи.....	3
Общее описание предметной области.....	3
Описание табличной модели.....	4
Пример работы программы.....	10
Код программы.....	12

Постановка задачи

Сформировать модель принятия решений в виде совокупности таблиц решений и написать модуль-интерпретатор для данной модели. Объектом моделирования является игра «Крестики - Нолики» на поле 3x3.

Общее описание предметной области

Игра «Крестики - Нолики» является антагонистической игрой двух лиц. На листе бумаги рисуется сетка размера 3x3. Игрок, делающий первый ход, рисует в любой из клеток какой-либо символ, обычно «крестик». Второй игрок делает ответный ход: рисует другой символ, например «нолик», в любой из оставшихся свободных клеток. Далее действия игроков повторяются. Выигравшим считается тот игрок, который первый сумеет поставить 3 своих символа по горизонтали, вертикали или диагонали. Если ни одному из игроков это не удалось, а все девять клеток заняты, то игра закончилась в ничью.

В электронной версии игры «Крестики - Нолики» в качестве игроков рассматриваются человек и компьютер. Компьютер всегда рисует в свободной клеточке синий круг, а человек – красный. Ход человека осуществляется щелчком мыши на той клетке, которая должна быть отмечена. Чтобы начать очередную игру, выберите пункт «Новая» меню «Игра» (рис.1). По умолчанию первый ход осуществляет компьютер. Для изменения права первого хода используйте радиогруппу «Первый ходит человек» и «Первый ходит компьютер» меню «Игра» (рис.1).

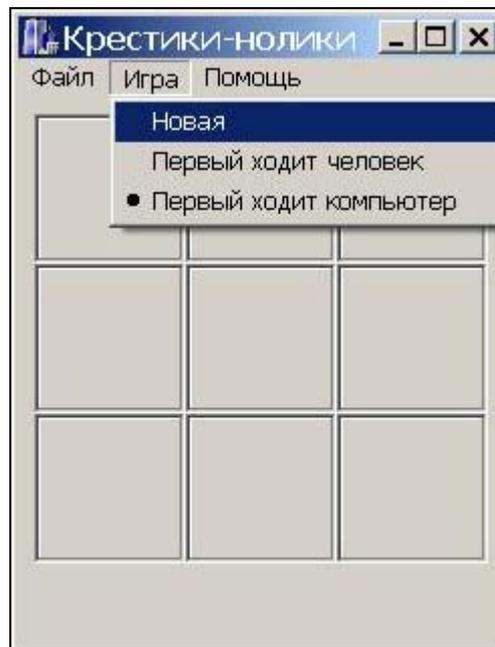


Рисунок 1. Меню "Игра" электронной версии игры «Крестики-нолики»

Описание табличной модели

Для описания модели на языке таблиц решений СИМПР используется следующие таблицы:

- 1) таблица «Игра» - служит для отслеживания начала/конца игры, регулирования ходов, проверки на ничью и останова (рис. 2-3);
- 2) таблица «Выигрыш» - служит для отслеживания выигрыша и передачи хода от одного игрока другому (рис. 4-5);
- 3) таблица «Ход компьютера» - служит для осуществления выбора той ячейки, куда компьютер поставит синий шар (таб. 1-4).

Описание таблицы	
Название таблицы	Игра
Условие 1	Приложение закрыто?
Условие 2	Игра началась?
Условие 3	Конец игры?
Условие 4	Есть свободные клетки?
Условие 5	Ход компьютера?
Условие 6	Ход человека?
Условие 7	Человек сделал ход?
Действие 1	Ждать пока человек думает
Действие 2	Ничья, конец игры

Рисунок 2. Условия и действия таблицы "Игра"

	1	2	3	4	5	6	7
C1	T	F	F	F	F	F	F
C2		F	T	T	T	T	T
C3			T	F	F	F	F
C4				T	T	T	F
C5				T			
C6					T	T	
C7					F	T	
R1		1	1		1		
R2							1
+	S			3		2	

Рисунок 3. Таблица "Игра"

Логическое отношения для таблицы «Игра»: $c5 \rightarrow !c6$

Таблица полна и непротиворечива, о чем свидетельствует диагностика СИМПР (рис. 3).

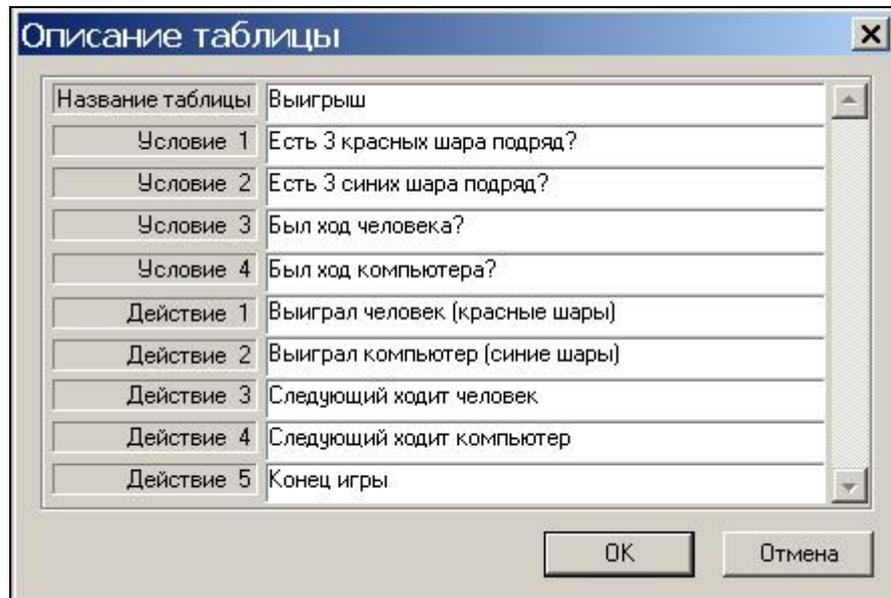


Рисунок 4. Условия и действия таблицы "Выигрыш"

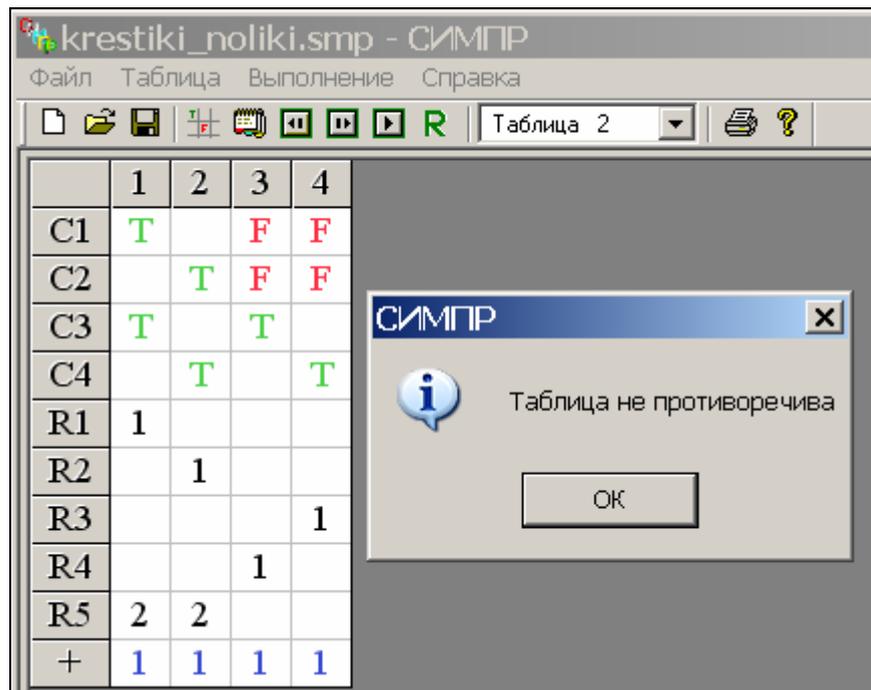


Рисунок 5. Таблица "Выигрыш"

Логические отношения для таблицы «Выигрыш»:

$$c1 \rightarrow !c4,$$

$$c2 \rightarrow !c3,$$

$$c3 \rightarrow !c4.$$

Таблица полна и непротиворечива, о чем свидетельствует диагностика СИМПР (рис. 5).

	Название таблицы	Ход компьютера
C1	Условие 1	Красный шар в ячейке 11?
C2	Условие 2	Красный шар в ячейке 12?
C3	Условие 3	Красный шар в ячейке 13?
C4	Условие 4	Красный шар в ячейке 21?
C5	Условие 5	Красный шар в ячейке 22?
C6	Условие 6	Красный шар в ячейке 23?
C7	Условие 7	Красный шар в ячейке 31?
C8	Условие 8	Красный шар в ячейке 32?
C9	Условие 9	Красный шар в ячейке 33?
C10	Условие 10	Синий шар в ячейке 11?
C11	Условие 11	Синий шар в ячейке 12?
C12	Условие 12	Синий шар в ячейке 13?
C13	Условие 13	Синий шар в ячейке 21?
C14	Условие 14	Синий шар в ячейке 22?
C15	Условие 15	Синий шар в ячейке 23?
C16	Условие 16	Синий шар в ячейке 31?
C17	Условие 17	Синий шар в ячейке 32?
C18	Условие 18	Синий шар в ячейке 33?
R1	Действие 1	Поставить синий шар в ячейку 11
R2	Действие 2	Поставить синий шар в ячейку 12
R3	Действие 3	Поставить синий шар в ячейку 13
R4	Действие 4	Поставить синий шар в ячейку 21
R5	Действие 5	Поставить синий шар в ячейку 22
R6	Действие 6	Поставить синий шар в ячейку 23
R7	Действие 7	Поставить синий шар в ячейку 31
R8	Действие 8	Поставить синий шар в ячейку 32
R9	Действие 9	Поставить синий шар в ячейку 33
R10	Действие 10	Поставить синий шар в пустую ячейку

Таблица 1. Условия и действия таблицы «Ход компьютера»

Логические отношения для таблицы «Ход компьютера»:

$$\begin{aligned}
 c1 &\rightarrow !c10, \\
 c2 &\rightarrow !c11, \\
 c3 &\rightarrow !c12, \\
 c4 &\rightarrow !c13, \\
 c5 &\rightarrow !c14, \\
 c6 &\rightarrow !c15, \\
 c7 &\rightarrow !c16, \\
 c8 &\rightarrow !c17, \\
 c9 &\rightarrow !c18.
 \end{aligned}$$

Таблица полна и непротиворечива, о чем свидетельствует диагностика СИМПР.

	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13	P14	P15
C1	F	T	F	F	F	F	F	F	F	F	T	T	F	F	F
C2	F	F	T	F	F	F	F	F	F	F	T	F	T	F	F
C3	F	F	F	T	F	F	F	F	F	F	F	T	T	F	F
C4	F	F	F	F	T	F	F	F	F	F	F	F	F	T	T
C5	F	F	F	F	F	T	F	F	F	F	F	F	F	T	F
C6	F	F	F	F	F	F	T	F	F	F	F	F	F	F	T
C7	F	F	F	F	F	F	F	T	F	F	F	F	F	F	F
C8	F	F	F	F	F	F	F	F	T	F	F	F	F	F	F
C9	F	F	F	F	F	F	F	F	F	T	F	F	F	F	F
C10						F							F		
C11												F			
C12											F				
C13															
C14		F	F	F	F		F	F	F	F					F
C15														F	
C16															
C17															
C18															
R1						1							1		
R2												1			
R3											1				
R4															
R5	1	1	1	1	1		1	1	1	1					1
R6														1	
R7															
R8															
R9															
R10															
+	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Таблица 2. Таблица «Ход компьютера», правила P1 – P15

	P16	P17	P18	P19	P20	P21	P22	P23	P24	P25	P26	P27	P28	P29	P30
C1	F	F	F	F	T	T	F	F	F	F	F	F	F	T	T
C2	F	F	F	F	F	F	F	T	T	F	F	F	F	F	F
C3	F	F	F	F	F	F	F	F	F	F	T	T	F	F	F
C4	F	F	F	F	T	F	T	F	F	F	F	F	F	F	F
C5	T	F	F	F	F	F	F	T	F	T	F	F	F	T	F
C6	T	F	F	F	F	F	F	F	F	F	T	F	T	F	F
C7	F	T	T	F	F	T	T	F	F	F	F	F	F	F	F
C8	F	T	F	T	F	F	F	F	T	T	F	F	F	F	F
C9	F	F	T	T	F	F	F	F	F	F	F	T	T	F	F
C10							F								
C11										F					
C12													F		
C13	F					F									
C14									F						F
C15												F			
C16				F	F										
C17			F					F							
C18		F									F			F	
R1							1								
R2										1					
R3													1		
R4	1					1									
R5									1						1
R6												1			
R7				1	1										
R8			1					1							
R9		1									1			1	
R10															
+	2	2	2	2	2	2	2	2	2	2	2	2	2	2	2

Таблица 3. Таблица «Ход компьютера», правила P15 – P30

	P31	P32	P33	P34	P35	P36	P37	P38	P39	P40	P41	P42	E
C1	F	F	F	F	T	F	F	F	F	F	F	F	
C2	F	F	F	F	F	T	F	F	F	F	F	F	
C3	F	T	T	F	F	F	T	F	F	F	F	F	
C4	F	F	F	F	F	F	F	T	F	F	F	F	
C5	T	T	F	T	F	F	F	F	F	F	F	F	
C6	F	F	F	F	F	F	F	F	T	F	F	F	
C7	F	F	T	T	F	F	F	F	F	T	F	F	
C8	F	F	F	F	F	F	F	F	F	F	T	F	
C9	T	F	F	F	F	F	F	F	F	F	F	T	
C10	F					F				F			
C11													
C12				F	F							F	
C13													
C14			F		T	T	T	T	T	T	T	T	
C15													
C16		F						F					
C17													
C18							F		F		F		
R1	1					1				1			
R2													
R3				1	1							1	
R4													
R5			1										
R6													
R7		1						1					
R8													
R9							1		1		1		
R10													1
+	2	2	2	2	2	2	2	2	2	2	2	2	2

Таблица 4. Таблица «Ход компьютера», правила P31 – P42, E

Ниже представлены результаты трансляции таблиц.

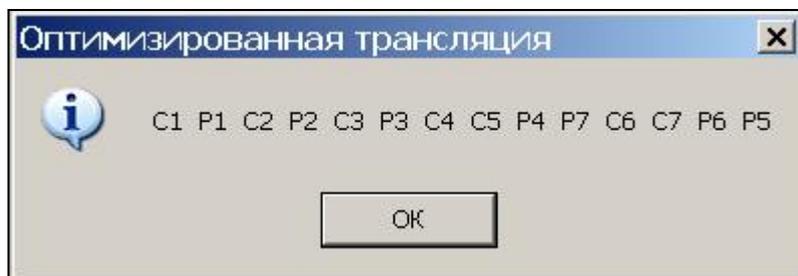


Рисунок 2. Трансляция таблицы "Игра"

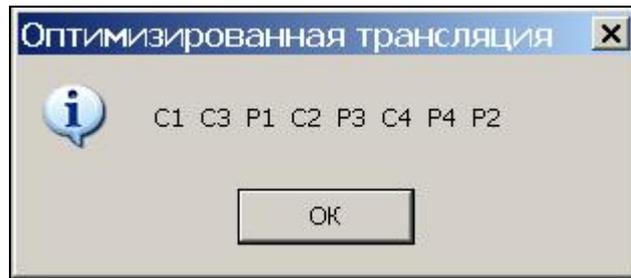


Рисунок 3. Трансляция таблицы "Выигрыш"

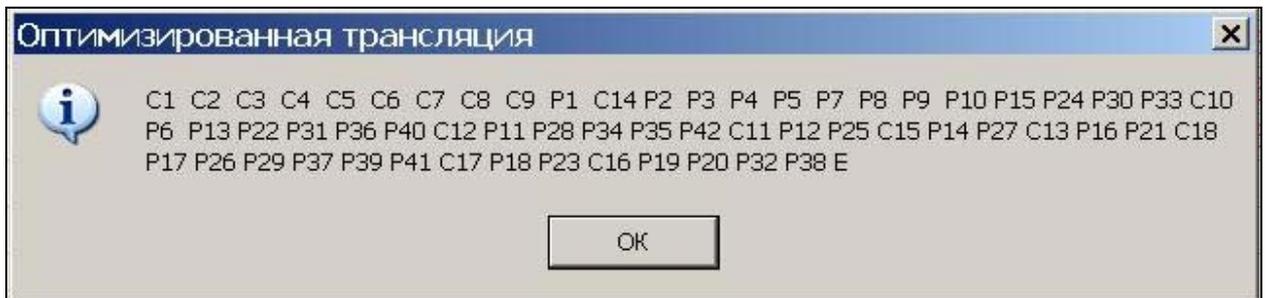


Рисунок 4. Трансляция таблицы "Ход компьютера"

Пример работы программы

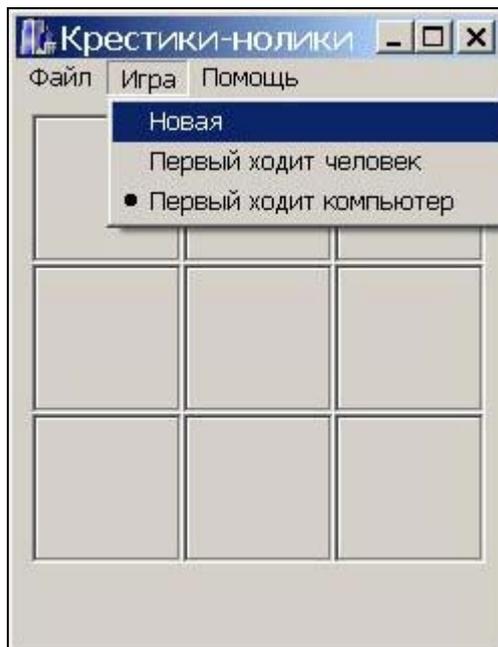


Рисунок 6. Начало игры

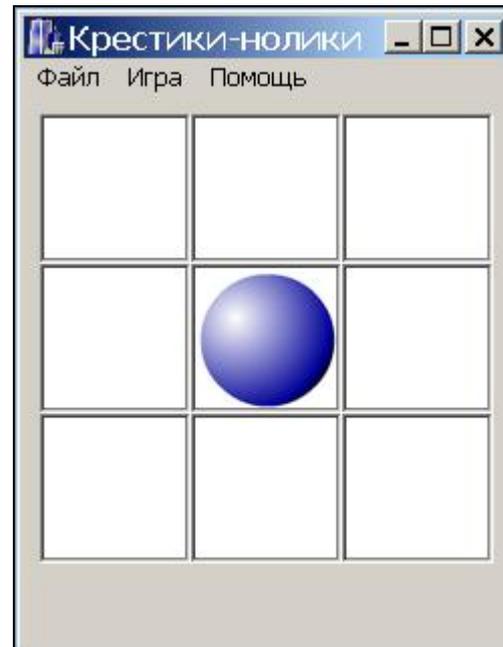


Рисунок 7. Ход 1, компьютер, правило P1

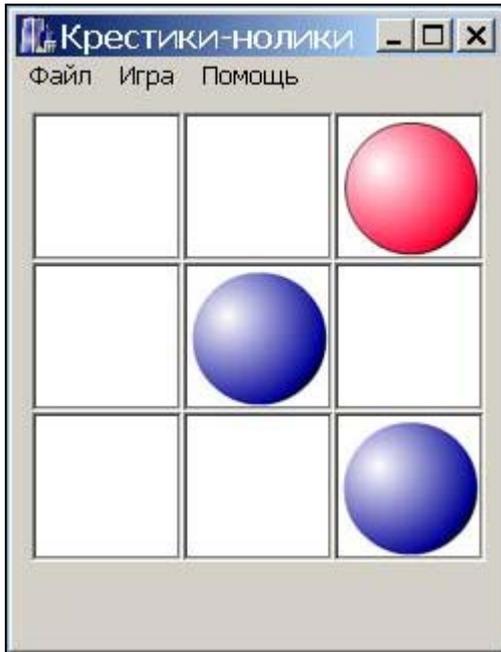


Рисунок 8. Ход 2, человек
Ход 3, компьютер, правило P37

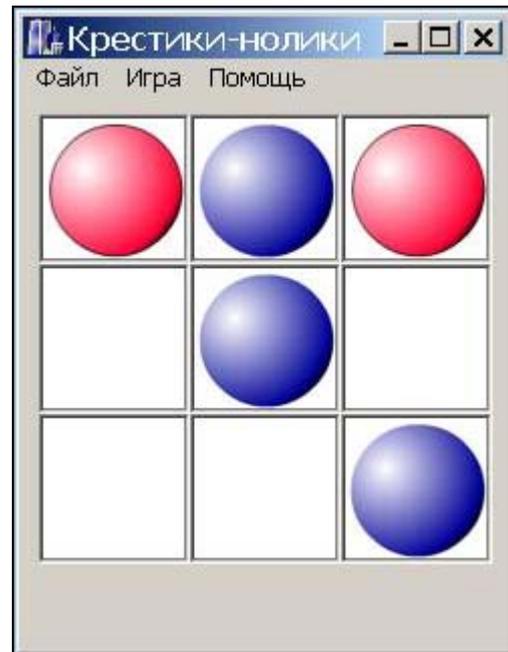


Рисунок 9. Ход 4, человек
Ход 5, компьютер, правило E

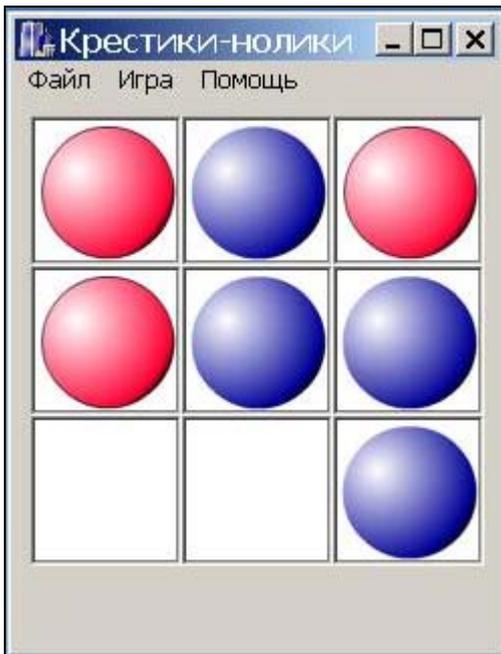


Рисунок 10. Ход 6, человек
Ход 7, компьютер, правило E

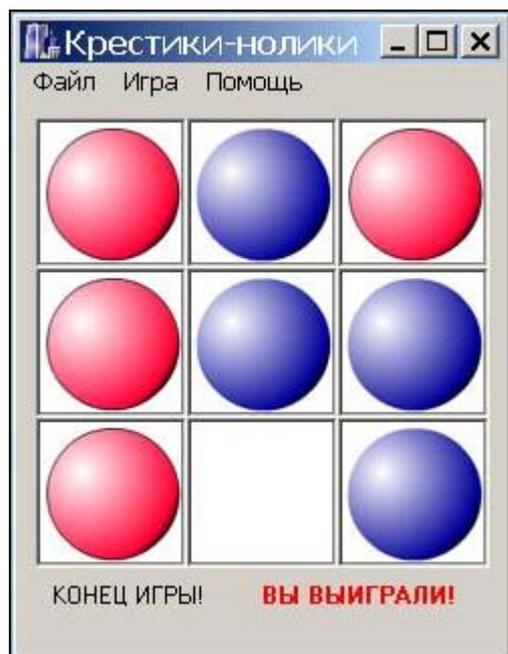


Рисунок 11. Ход 8, человек выиграл

Код программы

Ниже приведена основная часть программного кода: обработка сообщений СИМПР.

```
#define USERMESSAGE "Krestiki_noliki"
#define YES 1
#define NO 0
#define CONDITIONS 0
#define ACTIONS 1
//-----
//регистрация сообщения, используемого для обмена с СИМПР
void __fastcall TForm1::FormCreate(TObject *Sender)
{
    OurMessage = RegisterWindowMessage(USERMESSAGE);
}
//-----
void __fastcall TForm1::WndProc(Messages::TMessage &Message)
{
    if(Message.Msg == OurMessage)
    {
        Message.Result = SIMPR(Message.WParam, Message.LParam);
    }
    else TCustomForm::WndProc(Message);
}
//-----
//обработка сообщений СИМПР
int TForm1::SIMPR(WPARAM wParam, LPARAM lParam)
{
    switch(HIWORD(wParam))
    {
        case CONDITIONS:
        {
            return SIMPR_Conditions(wParam, lParam);
        }
        case ACTIONS:
        {
            return SIMPR_Actions(wParam, lParam);
        }
    }
}
//-----
//обработка условий СИМПР
int TForm1::SIMPR_Conditions(WPARAM wParam, LPARAM lParam)
{
    switch(LOWORD(wParam))
    {
        case 1: //таблица №1 "Игра"
        {
            switch(lParam)
            {
                case 1: //приложение закрыто?
                {
                    if(AppClose){return YES;}
                    return NO;
                }
                case 2: //Игра началась?
                {
                    if(StartPlay){return YES;}
                    return NO;
                }
                case 3: //конец игры?
```

```

{
    if (EndPlay) {return YES;}
    return NO;
}
case 4: //есть свободные клетки?
{
    for(int i=0; i<3; i++)
    {
        for(int j=0; j<3; j++)
        {
            if (pole[i][j]==0)
            {
                //запомнили координаты пустой клетки
                EmptyX = i;
                EmptyY = j;
                return YES;
            }
        }
    }
    return NO;
}
case 5: //ход компьютера?
{
    if (NextStep==0) {return YES;}
    return NO;
}
case 6: //ход человека?
{
    if (NextStep==1) {return YES;}
    return NO;
}
case 7: //Человек уже сделал ход?
{
    if (PrevStep==1) {return YES;}
    return NO;
}
} //switch
break;
} //case таблица №1 "Игра"
case 2: //таблица №2 "Выигрыш"
{
    switch (lParam)
    {
        case 1: //есть 3 красных шара подряд?
        {
            if (Winner(1)) {return YES;}
            return NO;
        }
        case 2: //есть 3 синих шара подряд?
        {
            if (Winner(2)) {return YES;}
            return NO;
        }
        case 3: //был ход человека?
        {
            if (PrevStep==1) {return YES;}
            return NO;
        }
        case 4: //был ход компьютера?
        {
            if (PrevStep==0) {return YES;}
            return NO;
        }
    }
} //switch

```

```

break;
} //case таблица №2 "Выигрыш"
case 3: //таблица №3 "Ход компьютера"
{
    switch(lParam)
    {
        case 1: //красный шар в 11?
        {
            if (pole[0][0]==1) {return YES;}
            return NO;
        }
        case 2: //красный шар в 12?
        {
            if (pole[0][1]==1) {return YES;}
            return NO;
        }
        case 3: //красный шар в 13?
        {
            if (pole[0][2]==1) {return YES;}
            return NO;
        }
        case 4: //красный шар в 21?
        {
            if (pole[1][0]==1) {return YES;}
            return NO;
        }
        case 5: //красный шар в 22?
        {
            if (pole[1][1]==1) {return YES;}
            return NO;
        }
        case 6: //красный шар в 23?
        {
            if (pole[1][2]==1) {return YES;}
            return NO;
        }
        case 7: //красный шар в 31?
        {
            if (pole[2][0]==1) {return YES;}
            return NO;
        }
        case 8: //красный шар в 132?
        {
            if (pole[2][1]==1) {return YES;}
            return NO;
        }
        case 9: //красный шар в 33?
        {
            if (pole[2][2]==1) {return YES;}
            return NO;
        }
        case 10: //синий шар в 11?
        {
            if (pole[0][0]==2) {return YES;}
            return NO;
        }
        case 11: //синий шар в 12?
        {
            if (pole[0][1]==2) {return YES;}
            return NO;
        }
        case 12: //синий шар в 13?
        {
            if (pole[0][2]==2) {return YES;}

```

```

        return NO;
    }
    case 13: //синий шар в 21?
    {
        if(pole[1][0]==2){return YES;}
        return NO;
    }
    case 14: //синий шар в 22?
    {
        if(pole[1][1]==2){return YES;}
        return NO;
    }
    case 15: //синий шар в 23?
    {
        if(pole[1][2]==2){return YES;}
        return NO;
    }
    case 16: //синий шар в 31?
    {
        if(pole[2][0]==2){return YES;}
        return NO;
    }
    case 17: //синий шар в 32?
    {
        if(pole[2][1]==2){return YES;}
        return NO;
    }
    case 18: //синий шар в 33?
    {
        if(pole[2][2]==2){return YES;}
        return NO;
    }
    }//switch
    break;
} //case  таблица №3 "Ход компьютера"
} //switch
return 0;
}
//-----
//обработка действий СИМПР
int TForm1::SIMPR_Actions(WPARAM wParam, LPARAM lParam)
{
    switch(LOWORD(wParam))
    {
        case 1:// таблица №1 "Игра"
        {
            switch(lParam)
            {
                case 0:
                {
                    return YES;
                }
                case 1: //ждать пока человек думает
                {
                    return YES;
                }
                case 2: //ничья, конец игры
                {
                    Label1->Font->Color = clBlack;
                    Label1->Caption = "НИЧЬЯ!!!";
                    Label2->Caption = "КОНЕЦ ИГРЫ!";
                    EndPlay = true;
                    StartPlay = false;
                    N4->Enabled = true;
                }
            }
        }
    }
}

```

```

        N2->Enabled = true;
        Image1->Enabled = false;
        Image2->Enabled = false;
        Image3->Enabled = false;
        Image4->Enabled = false;
        Image5->Enabled = false;
        Image6->Enabled = false;
        Image7->Enabled = false;
        Image8->Enabled = false;
        Image9->Enabled = false;
        return YES;
    }
} //switch
break;
} //case таблица №1 "Игра"
case 2://таблица №2 "ВЫИГРЫШ"
{
    switch(lParam)
    {
        case 0:
        {
            return YES;
        }
        case 1: //выиграл человек (красные шары)
        {
            Label1->Font->Color = clRed;
            Label1->Caption = "ВЫ ВЫИГРАЛИ!";
            return YES;
        }
        case 2: //выиграл компьютер (синие шары)
        {
            Label1->Font->Color = clBlue;
            Label1->Caption = "Я ВЫИГРАЛ!";
            return YES;
        }
        case 3: //следующий ходит человек
        {
            NextStep = 1;
            return YES;
        }
        case 4: //следующий ходит компьютер
        {
            NextStep = 0;
            return YES;
        }
        case 5: //конец игры
        {
            Label2->Caption = "КОНЕЦ ИГРЫ!";
            Image1->Enabled = false;
            Image2->Enabled = false;
            Image3->Enabled = false;
            Image4->Enabled = false;
            Image5->Enabled = false;
            Image6->Enabled = false;
            Image7->Enabled = false;
            Image8->Enabled = false;
            Image9->Enabled = false;
            EndPlay = true;
            StartPlay = false;
            N4->Enabled = true;
            N2->Enabled = true;
            return YES;
        }
    }
} //switch

```

```
break;
} //case таблица №2 "Выигрыш"
case 3: //таблица №3 "Ход компьютера"
{
    switch(lParam)
    {
        case 0:
        {
            return YES;
        }
        case 1: //поставить синий шар в 11
        {
            pole[0][0] = 2;
            Image1->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 2: //поставить синий шар в 12
        {
            pole[0][1] = 2;
            Image2->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 3: //поставить синий шар в 13
        {
            pole[0][2] = 2;
            Image3->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 4: //поставить синий шар в 21
        {
            pole[1][0] = 2;
            Image4->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 5: //поставить синий шар в 22
        {
            pole[1][1] = 2;
            Image5->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 6: //поставить синий шар в 23
        {
            pole[1][2] = 2;
            Image6->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 7: //поставить синий шар в 31
        {
            pole[2][0] = 2;
            Image7->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
            return YES;
        }
        case 8: //поставить синий шар в 32
        {
            pole[2][1] = 2;
            Image8->Picture->LoadFromFile("BlueBall.bmp");
            PrevStep = 0;
        }
    }
}
```

```

        return YES;
    }
    case 9: //поставить синий шар в 33
    {
        pole[2][2] = 2;
        Image9->Picture->LoadFromFile("BlueBall.bmp");
        PrevStep = 0;
        return YES;
    }
    case 10: //поставить синий шар в пустую клетку
    {
        PutBallInEmpty(); //ставит синий шар в пустую клетку
        PrevStep = 0;
        return YES;
    }
    } //switch
    break;
} //case таблица №3 "Ход компьютера"
} //switch
return 0;
}
//-----
//Человек ставит красный шар в клетку 11
void __fastcall TForm1::Image1Click(TObject *Sender)
{
    //отмечаем, что красный шар поставлен в 11
    pole[0][0] = 1;
    //рисуем
    Image1->Picture->LoadFromFile("RedBall.bmp");
    PrevStep = 1;
}

```