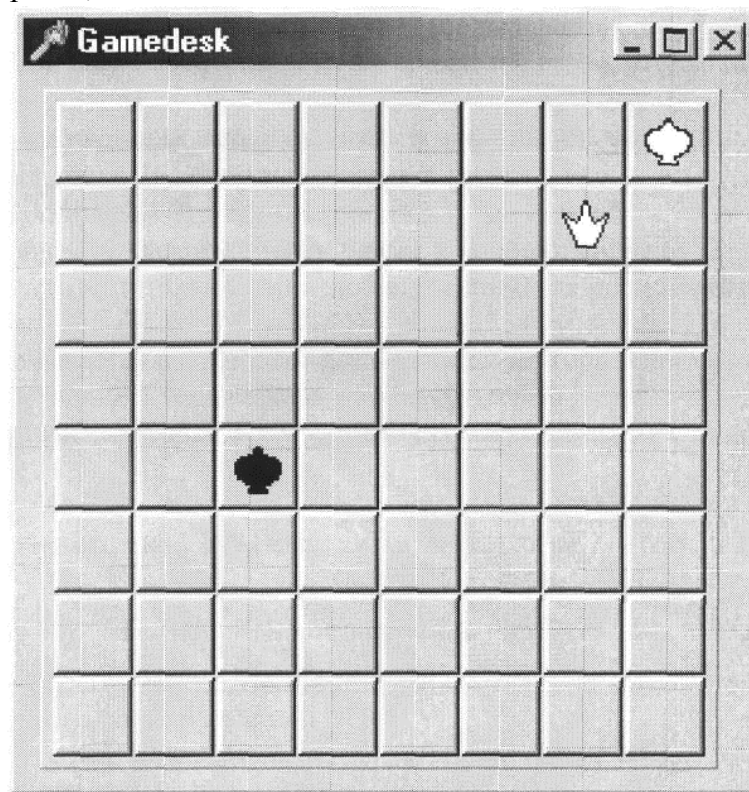


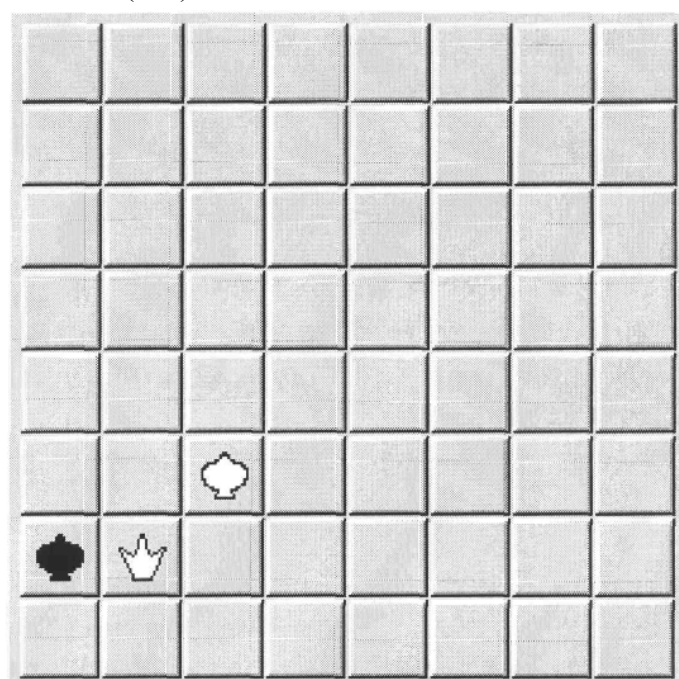
1. Постановка задачи.

Необходимо решить шахматную задачу постановки мата одинокому черному королю, У белых король и ферзь. Начальная позиция - белый король (в дальнейшем W) в правом верхнем углу (8,8), белый ферзь (Q) в точке (7,7), черный король (B) в произвольном допустимом месте. Допустимые ходы фигур и заключительное состояние (мат) определяются по шахматным правилам. Ход белых.

Таким образом, начальное положение:



Конечное состояние (мат):



В данном случае система СИМПР определяет ходы белых (Таблица 1), а возможность совершать ходы за черных (Таблица 2) дана пользователю (ЛПР). Таким образом, повышается наглядность процесса ПР.

2. Алгоритм ПР

В двух словах алгоритм постановки мата выглядит следующим образом.

а) Вражеский король загоняется в угол (белый ферзь сохраняет Г-образную позицию). При необходимости совершаются выжидательные ходы белым королем вверх-вниз.

б) После выполнения первого пункта возможны две ситуации:

- * ферзь в точке (2,4), черный король в точке (1,1)

- * ферзь в точке (4,7), черный король в точке (1,8)

То есть ферзь не может подойти ближе к королю, иначе будет съеден, а королю черных оставлено только одно свободное поле для предотвращения патовой ситуации.

Теперь начинается следующая фаза - движение короля на матовую позицию (соответственно 6 горизонталь или 6 вертикаль)

в) Последний ход ферзя осуществляется соответственно в точку (2,2) или (7,2). Мат.

3. Основная таблица СИМПРа.

Условия:

C1: В отстоит от Q более, чем на две вертикали

C2: В находится на соседней вертикали с Q

C3: В на две горизонтали ниже Q

C4: В в левом нижнем углу (1,1)

C5: В на соседней горизонтали

C6: W на 8 горизонтали

C7: Q в точке (2,4)

C8: В на 8 горизонтали

C9: Q на 8 горизонтали

C10: Q в точке (4,7)

C11: В в точке (1,8)

C12: W на третьей вертикали

C13: W на 6 горизонтали

C14: W на 8 вертикали

Действия:

A1: Ферзь движется на две клетки влево

A2: Белый король движется вниз A3:

Белый король движется вверх A4: Ферзь

движется на две клетки вниз A5: Ферзь

движется на 4 горизонталь A6: Белый

король движется влево

A7: Ферзь движется на одну клетку вправо A8:
 Ферзь движется на одну клетку вниз A9:
 Ферзь движется на одну клетку вверх A10:
 Ферзь движется в точку (4,7) A11: Ферзь
 движется в точку (2,2) A12: Ферзь движется в
 точку (2,7)

Правила:

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	E
1	T	F	F	F	F	F	F	F	F		T	T	F	F						
2	F	T	T	T	T	T	F	F	F											
3		T	T	F	F	F		F	F											
4				F	T	T														
5		F	F				F	T	T											
6		T	F										T	F						
7					F	T									T	T				
8	F	F	F	F	F	F	F	F	F	T	T	T	T	T						
9								T	F											
10										T	F	F	F	F	F	F	T	T	T	
11											T	F								
12														F	T		F	T		
13																	F	T	T	
14	T	T	T	T	T	T	T	T	T	T	T	T	T	T	F	F	F	F	F	
1	1										1									
2		1				1						1		1		1				
3			1										1							
4				1																
5					1															
6						2			1					2		2	1			
7							1		1											
8								1												
9									2											
10										1										
11															1					
12																			1	
+	2	2	2	2	2	2	2	2	2	2	2	2	2	2	S	2	2	S	S	

И, естественно:



И, дополнив её стоповым состоянием по ELSE, получаем:

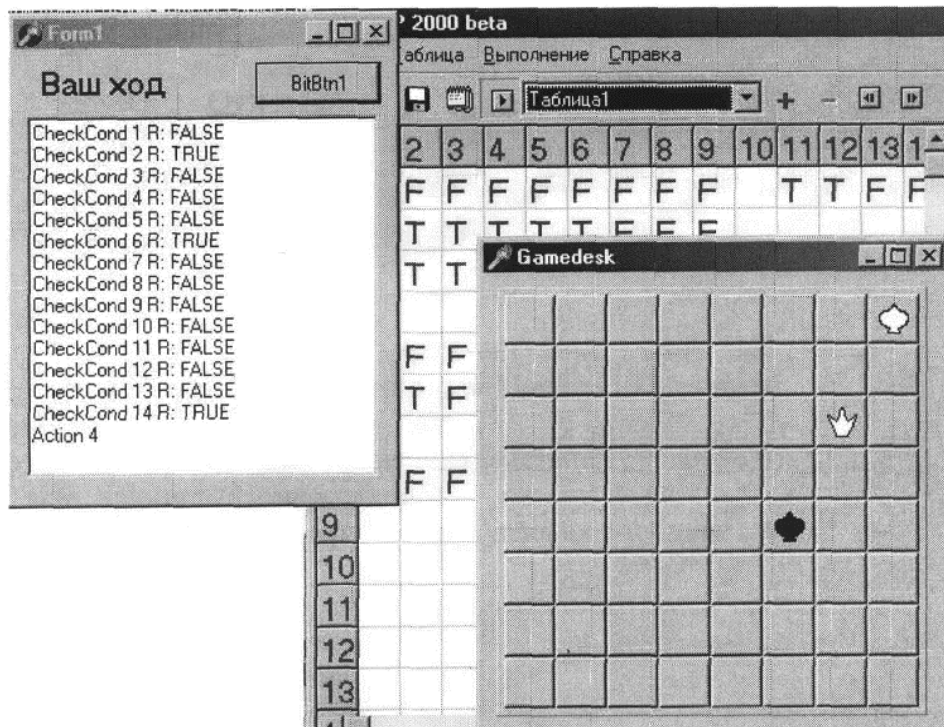


4. Дополнительная таблица

	1	E
1		
1		
+ 1	S	

Дополнительная таблица предназначена для реализации хода черных. Возможность хода предоставлена пользователю в программе-интерпретаторе.

5. Модуль-интерпретатор



Текст программы:

```
const YES =
  3; NO = 2;
```

```
var Form1:
  TForm1;
```

implementation uses

```
GameDeskUnit; {$R *.DFM}
```

```
procedure TForm1.WndProc(var Message: TMessage);
begin
  RegMessage:=RegisterWindowMessage('MyMessage');
  if Message.Msg = RegMessage then
    begin OnOurMessage(Message); //Обрабатываем наше сообщение
    end //Завершаем обработку сообщения !!!!
  else //Вызываем стандартный обработчик родительского класса
    inherited WndProc(Message);
end;
```

```
procedure TForm1.ShowDesk;
begin
  with GameDesk do begin
```

```

Visible:=false; ClearAll;
MoveQ(Qx,Qy);
MoveW(Wx,Wy);
MoveB(Bx,By);
ShowModal; Visible:=true;
end; end;

```

```

procedure TForm1.OnOurMessage(var Message : TMessage);

```

```

var
Mtype : integer; // Тип сообщения (действие(1)/условие(0))
Table : integer; // Номер таблицы
Number : integer; // Номер действия/условия в таблице
  Res : integer;
  begin
Mtype:= Message.WParamHi;
Table:= Message.WParamLo;
Number:= Message.LParam;
case Mtype of
// Условие
0:  if Table=1 then
      if CheckCond(Number) then
        Res:=YES else
        Res:=No else
        begin ShowDesk;
        Res:=YES; end;
// Действие
1:  begin
if Table=1 then
case Number of
0:  begin
      Label1.Caption:='Mat!';
      ShowDesk; Close; end;
1:  Qx:=Bx+2;
2:  Wy:=Wy-1;
3:  Wy:=Wy+1;
4:  Qy:=By+2;
5:  Qy:=4;
6:  Wx:=Wx-1;
7:  Qx:=Qx-1;
8:  Qy:=Qy-1;
9:  Qy:=Qy+1;
10: begin Qx:=4;

```

```

    Qy:=7
    end;
    11: begin
    Qx:=2;
    Qy:=2
    end;
    12: begin
    Qx:=2;
    Qy:=7
    end;
    end;
    Res:=YES;
    Memo1.Lines.Add('Action '+IntToStr(Number)) end
    else begin
    {    ShowDesk;}
    Res:=YES;
    end; end;
    Message.Result:= Res;
end;

```

```

function TForm1.CheckCond(N: integer):boolean;
begin

```

```

case N of
    1: Result:=(abs(Bx-Qx)>2);
    2: Result:=(abs(Bx-Qx)=1);
    3: Result:=(By=Qy-2);
    4: Result:=(Bx=1) and (By=1);
    5: Result:=(abs(By-Qy)=1);
    6: Result:=(Wy=8);
    7: Result:=(Qx=2) and (Qy=4);
    8: Result:=(By=8);
    9: Result:=(Qy=8);
    10: Result:=(Qx=4) and (Qy=7);
    11: Result:=(Bx=1) and (By=8);
    12: Result:=(Wx=3);
    13: Result:=(Wy=6);
    14: Result:=(Wx=8);
    end;
    if Result then
        Memo1.Lines.Add('CheckCond '+IntToStr(N)+' R: TRUE')
    else
        Memo1.Lines.Add('CheckCond '+IntToStr(N)+' R: FALSE')
    end;
end;

```