

```
{ $M 65520, 0, 655360 }
```

```
program schur4(input, output);
```

```
type    color=array[1..50] of integer;
        sums=array[1..100] of integer;
        answer=array[1..50] of char;
```

```
var     red, white, blue, green : color;
        redsums, whitesums, bluesums, greensums : sums;
        i, n, sol counter, call counter, mcounter : integer;
        solution : answer;
```

```
procedure update(var c : color; var csum : sums; number : integer);
  var i : integer;
  begin
    c[number]:=1; csum[2*number]:=1;
    for i:=1 to 50 do if c[i]=1 then csum[i+number]:=1;
  end;
```

```
procedure initialize;
  var i : integer;
  begin
    sol counter:=0; call counter:=0; mcounter:=0;
    for i:=1 to 50 do
      begin
        red[i]:=0; white[i]:=0; blue[i]:=0; green[i]:=0;
        solution[i]:=' ';
      end;
    for i:=1 to 100 do
      begin
        redsums[i]:=0; whitesums[i]:=0; bluesums[i]:=0; greensums[i]:=0;
      end;
    end;
```

```
procedure present(sol : answer; rsums, wsums, bsums, gsums : sums);
  var i : integer;
  begin
    sol counter:=sol counter+1;
    write(' Solution ', sol counter, ' : ');
    for i:=1 to 50 do write(sol[i]); writeln; writeln;
    write(' Red sums: ');
    for i:=1 to 100 do if rsums[i]=1 then write(i, ' '); writeln; writeln;
    write(' White sums: ');
    for i:=1 to 100 do if wsums[i]=1 then write(i, ' '); writeln; writeln;
    write(' Blue sums: ');
    for i:=1 to 100 do if bsums[i]=1 then write(i, ' '); writeln; writeln;
    write(' Green sums: ');
    for i:=1 to 100 do if gsums[i]=1 then write(i, ' '); writeln;
    writeln; writeln;
  end;
```

s4mar15.pas

```
procedure try(x,k : integer; r,w,b,g : color; rsums, wsums, bsums, gsums : sums);
begin
  if callcounter=32000 then
    begin
      mcounter:=mcounter+1;
      callcounter:=0;
    end
  else callcounter:=callcounter+1;

  if x=1 then update(r,rsums,k-1);
  if x=2 then update(w,wsums,k-1);
  if x=3 then update(b,bsums,k-1);
  if x=4 then update(g,gsums,k-1);
  if k=45 then
    begin
      present(solution,rsums,wsums,bsums,gsums); writeln;
    end
  else
    begin
      if rsums[k]=0 then
        begin
          solution[k]='r';
          try(1,k+1,r,w,b,g,rsums,wsums,bsums,gsums);
        end;
      if wsums[k]=0 then
        begin
          solution[k]='w';
          try(2,k+1,r,w,b,g,rsums,wsums,bsums,gsums);
        end;
      if bsums[k]=0 then
        begin
          solution[k]='b';
          try(3,k+1,r,w,b,g,rsums,wsums,bsums,gsums);
        end;
      if gsums[k]=0 then
        begin
          solution[k]='g';
          try(4,k+1,r,w,b,g,rsums,wsums,bsums,gsums);
        end;
    end;

  end;

begin
writeln;
initialize;
solution[1]='r';
update(red,redsums,1);
solution[2]='w';
try(2,3,red,white,blue,green,redsums,whitesums,bluesums,greensums);
writeln('Recursive calls: ',mcounter,'*32,000+',callcounter);
end.
```