



Solutions écrites en Pascal (utilisant Delphi avec "Console Application")

I. Superdiv

20 points

```
program SuperDiv;
{$APPTYPE CONSOLE}
uses
  SysUtils;

var i1,i2,i3,i4,i6,i7,i8,i9 : integer;

function found(a,b,c,d,e,f,g,h : integer) : boolean;
var TEMP : boolean;
begin
  TEMP := true;
  TEMP := TEMP and ((100*a + 10*b + c) mod 3 = 0);
  TEMP := TEMP and ((1000*a + 100*b + 10*c + d) mod 4 = 0);
  TEMP := TEMP and ((100000*a + 10000*b + 1000*c + 100*d + 50 + e) mod 6 = 0);
  TEMP := TEMP and ((1000000*a + 100000*b + 10000*c + 1000*d + 500 +
    10*e + f) mod 7 = 0);
  TEMP := TEMP and ((10000000*a + 1000000*b + 100000*c + 10000*d +
    5000 + 100*e + 10*f + g) mod 8 = 0);

  TEMP := TEMP and (a <> b) and (a <> c) and (b <> c);
  TEMP := TEMP and (a <> d) and (b <> d) and (c <> d);
  TEMP := TEMP and (a <> e) and (b <> e) and (c <> e) and (d <> e);
  TEMP := TEMP and (a <> f) and (b <> f) and (c <> f) and (d <> f) and
    (e <> f);
  TEMP := TEMP and (a <> h) and (b <> h) and (c <> h) and (d <> h) and
    (e <> h) and (g <> h) and (f <> h);
  TEMP := TEMP and (a <> g) and (b <> g) and (c <> g) and (d <> g) and
    (e <> g) and (g <> f);
  TEMP := TEMP and (a <> 5) and (b <> 5) and (c <> 5) and (d <> 5);
  TEMP := TEMP and (e <> 5) and (f <> 5) and (g <> 5) and (h <> 5);
  found := TEMP
end;

begin
  i1 := 1;
  while i1 <= 9 do
    begin
      i2 := 2;
      while i2 < 9 do
        begin
          i3 := 1;
          while i3 <= 9 do
            begin
              i4 := 2;
              while i4 < 9 do
                begin
                  i6 := 2;
                  while i6 < 9 do
```

```

begin
  i7 := 1;
  while i7 <= 9 do
    begin
      i8 := 2;
      while i8 < 9 do
        begin
          i9 := 1;
          while i9 <= 9 do
            begin
              if
                found(i1,i2,i3,i4,i6,i7,i8,i9)
              then writeln
                (i1,i2,i3,i4,5,i6,i7,i8,i9);
              i9 := i9 + 1
            end;
            i8 := i8 + 2
          end;
          i7 := i7 + 1
        end;
        i6 := i6 + 2
      end;
      i4 := i4 + 2
    end;
    i3 := i3 + 1
  end;
  i2 := i2 + 2
end;
i1 := i1 + 1
end;
readln
end.

```

SuperDiv = 381654729

Pour satisfaire la contrainte des 5 secondes accordées, le chiffre du milieu doit être le 5 et les chiffres à des emplacements pairs doivent être pairs !

## II. LookAndSay

20 points

```

program LookAndSay;
{$APPTYPE CONSOLE}
uses SysUtils;

var STR,STR2,CH : string;
    I,J,Z,N : integer;

begin
  readln(N);
  STR := '1';
  writeln(STR);
  for J := 2 to N do
    begin
      STR2 := '';
      while length(STR) > 0 do
        begin
          CH := copy(STR,1,1);
          I := 1;
          Z := 0;
          while copy(STR,I,1) = CH do
            begin
              Z := Z + 1;
              I := I + 1
            end;

```

```

        STR2 := STR2 + inttostr(Z) + CH;
        delete(STR,1,Z);
    end;
    writeln(STR2);
    STR := STR2
end;
readln
end.

```

## III. Sudoku

60 points

```

program Sudoku;                (* ce n'est que du: backtracking ! *)
{$APPTYPE CONSOLE}
uses SysUtils;

type t_su = array [1..9,1..9] of
    record
        zahl : integer;
        fix : boolean
    end;

var su : t_su;
    f,ff : text;

procedure LESE (var su : t_su);
var i,j : integer;
begin
    assign(f,'SUDO_IN.TXT');
    reset(f);
    for i := 1 to 9 do
        for j := 1 to 9 do
            begin
                read(f,su[i,j].zahl);
                if su[i,j].zahl = 0
                then su[i,j].fix := false
                else su[i,j].fix := true
            end;
        close(f)
    end;

procedure AUSGABE (su : t_su);
var i,j : integer;
begin
    for i := 1 to 9 do
        begin
            for j := 1 to 9 do
                write(ff,su[i,j].zahl,' ');
            writeln(ff)
        end;
    writeln(ff)
end;

function OK (r,c,i : integer) : boolean;
var TEMP : boolean;
    x,y,boxx,boxy : integer;
begin
    TEMP := true;
    (* row *)
    for x := 1 to 9 do

```

```

    if x <> r
    then if su[x,c].zahl = i
        then TEMP := false;
(* col *)
for x := 1 to 9 do
    if x <> c
    then if su[r,x].zahl = i
        then TEMP := false;
(* 3x3 box *)
if r <= 3
then boxx := 1
else if r <= 6
    then boxx := 4
    else boxx := 7;
if c <= 3
then boxy := 1
else if c <= 6
    then boxy := 4
    else boxy := 7;
for x := boxx to boxx + 2 do
    for y := boxy to boxy + 2 do
        if (x <> r) and (y <> c)
        then if su[x,y].zahl = i
            then TEMP := false;
    OK := TEMP and (c <= 9) and (c > 0)
end;

function SUDOOK : boolean;
var TEMP : boolean;
    r,c : integer;
begin
    TEMP := true;
    for c := 1 to 9 do
        for r := 1 to 9 do
            begin
                if su[r,c].zahl <> 0
                then begin
                    if not OK(r,c,su[r,c].zahl)
                    then TEMP := false
                    end
                end;
            SUDOOK := TEMP
        end;

function DONE : boolean;
var i,j : integer;
begin
    DONE := true;
    for i := 1 to 9 do
        for j := 1 to 9 do
            if su[i,j].zahl = 0
            then DONE := false
        end;

procedure TRYOUT (r,c : integer);
var i : integer;
begin
    for i := 1 to 9 do
        if OK (r,c,i)
        then begin
            if not su[r,c].fix
            then su[r,c].zahl := i;
            if DONE
            then begin

```

on place un  
chiffre

```

        AUSGABE (su);
        exit
    end;
repeat
    r := r + 1;
    if r > 9
    then begin
        c := c + 1;
        r := 1
    end;
until not su[r,c].fix;
TRYOUT (r,c);
repeat
    r := r - 1;
    if r < 1
    then begin
        c := c - 1;
        r := 9
    end;
until not su[r,c].fix;
su[r,c].zahl := 0
end;
end;
begin
    LESE (su);
    assign(ff, 'SUDO_OUT.TXT');
    rewrite(ff);
    if SUDOOK
    then TRYOUT(1,1)
    else writeln(ff, 'No solution');
    close(ff)
end.

```

appel récursif

le chiffre placé nous a conduit dans cul-de-sac, on l'enlève et on essaye le chiffre suivant

**Tests et résultats:**

	SUDO_IN	SUDO_OUT
Test No 1 (exemple donné)	000005490 346000080 900006030 708030000 000501000 000060304 060200005 080000163 013600000	821375496 346129587 975846231 758432619 634591728 192768354 469213875 287954163 513687942
Test No 2 (impossible)	100000000 000000000 000000000 000000000 000000000 000000000 000000000 000000000 100000000	Pas de solution

Test No 3 (que des zéros)	000000000	147238569
	000000000	258169347
	000000000	369457128
	000000000	471382695
	000000000	582691473
	000000000	693574281
	000000000	714823956
	000000000	825916734
	000000000	936745812
Test No 4 (exemple facile)	070602040	579632841
	000050000	286154793
	301000605	341987625
	960010037	964518237
	050709010	853729416
	120040089	127346589
	608000302	618495372
	000070000	435271968
	090803050	792863154
Test No 5 (exemple moyen)	000000090	842763591
	500000060	513928467
	090000002	796451382
	230010006	239514876
	100080003	175682943
	400007005	468397215
	000009000	654139728
	081270000	981275634
	027006050	327846159
Test No 6 (exemple difficile)	000003000	419873562
	000400000	562491783
	800000001	873256491
	000000050	627348159
	000000340	185629347
	900710000	934715628
	040000000	748532916
	000180200	396184275
	250060000	251967834