

Toto je ukázka mých maker pro sazbu programů v Pascalu. Makra vznikla ve dnech 29.–30. srpna 2007 jako zkouška toho, co je v T_EXu možné. Nebyla nijak tvrdě testována, ale nemyslím si, že by měla obsahovat závažné chyby.

Vyzkoušíme psaní do řádku: **var** N : integer; $P^X := 1$; *Hello_world*(N); **end** (bez || bychom vložili další mezeru, takto je možné ji zamaskovat), zkusíme komentáře: **if** $\Sigma > 0$ **then** *write*(*{ komentáře fungují }*); *{ a toto taky }*.

Nyní zkusíme display Pascal mode (odsazení je určeno hodnotou \PASindent, zde nastavenou na \parindent):

```

program UKAZKA;
uses crt;
var  $M, N, \alpha$  : longint;  $\Sigma$  : real;  $P$  : array [1..10] of real;

{ Značka dolaru '$' je v Pascalu vyhrazena pro metaprikazy, ja ji vsak mam jako jednoduchy pro-
  stredek, jak do programu dostat nejake tokeny bez preruseni seznamu prikazu Pascalu. Kod $#1$
  vlozi #1 do matematickeho seznamu. Je treba myslet na to, ze kategorie nekterych znaku (#, $,
  &, {, }, |, _) jsou pozmeneny. Například pro  $\alpha$  ve 'var  $\alpha$  : longint' jsem napsal $\alpha$. }

type bod = record
     $X$  : integer;           { Občas je TEXu potřeba pomoci se zarovnáním. }
     $Y$  : integer;
end;                      { Kod &#1& vlozi mezeru o velikosti #1 pt. }

function NSD( $M, N$  : integer) : integer; { Vsimnete si, ze nerovnitka jsou takova, jaka mají
  byt, a komentar je s trochou snahy zarovnaný. To se
  udela tak, ze se napise \(\#1) a dalsi radky budou
  o tuto hodnotu (v pt) odsazeny. Nevhodne pro vlo-
  zene komentare, protoze pouzijeme-li tuto moznost,
  musime nechat sestavit odstavec. Navic je potreba
  zamaskovat konec radku znackou %. }

var  $i$  : integer;
begin
  NSD := 1;
  while ( $i \leq M$ ) and ( $i \leq N$ ) do begin { Dokud  $i$  není větší než  $M$  nebo  $N$  delej: }
    if ( $N \bmod i = 0$ ) and ( $M \bmod i = 0$ ) then { Zmen hodnotu NSD: } NSD :=  $i$ ;
     $i := i + 1$ ;
  end;
end;

{ Je implementováno nahrazování dvojic <=, <> a >= }
function Show( $S$  : string,  $l$  : Word);
begin
  write('string_mode:~!@#%~&*()-={}\: "<>?',  $l$ , 'Dobre, _ze?',  $S$ );
  if ( $(a \leq b)$  or ( $a \neq b$ ) or ( $a \geq b$ )) then write('Funguje _to. ');
   $Show := 0$ ;
end;

{ Chcete-li v komentáři sázet skupinu, pak vězte, že hranaté závorky mají kategorie 1 a 2, takže
  je samostatně nepoužívejte pro jiné účely než pro otevírání a zavírání skupiny. }

{  $N$ -krát vypíše 'Hello, world'. Podtržitko je součástí identifikátoru. Svislá čára | uzavře se-
 znam příkazů Pascalu a další svislá čára | ho zase obnoví. }
procedure Hello_world( $N$  : integer);
var  $i$  : integer;
begin
  for  $i = 1$  to  $N$  ||Příkazy TEXu fungují:  $\lim_{n \rightarrow \infty} 1/n = 0$ .|| do writeln('Hello, _world!');
end;

```

begin

Obcas se hodi dostat sem nejaky obycejny text

a treba ho i nejak zarovnat.

*⟨Nekdo rad uklada kusy zdrojoveho textu mimo a to podle vzoru WEBu.⟩**writeln('Zadejte_cisla:'); read(M, N);**writeln('Nejvetsi_spolecny_delitel_cisel', M, 'a', N, 'je', NSD(M, N));***end.***⟨Nekdo rad uklada kusy zdrojoveho textu mimo a to podle vzoru WEBu.⟩ ≡**write('Stringy_funguji_podle_ocekavani: # \$ % ^ & \ | ');*

A nyní nějaký pořádný program. Jedná se o mé nefunkční (proč nefunkční, to nevím) řešení problému zadaného v celostátním kole programovací olympiády 2007.

```

program policie;

type
  ukvrch = ^vrchol;
  uksez = ^seznam;
  seznam = record           { seznam = ukazatel na sousedni vrchol a uk. na dalsi pole v sez. }
    num : Longint;
    next : uksez;
  end;

  vrchol = record           { cislo vrcholu, ukazatel na prvnio souseda }
    num : Longint; mafie : Boolean; stupen : Longint; prvni : uksez;
    exist : Boolean; prechod : Longint;
  end;

var N, P, Police : Longint;
    A : array [1..100000] of ukvrch;

procedure Nacti_Strom;
var I, x, y : Longint; dalsi : uksez; F : text;
begin
  assign(F, 'policie.in');
  reset(F); read(F, N); read(F, P);
  for I := 1 to N - 1 do begin
    read(F, x); read(F, y);
    if A[x] = nil then begin           { jestlize jsme se jeste s vrcholem nesetkali → udelej ho }
      new(A[x]); A[x].num := x;
      new(A[x].prvni);
    end;
    dalsi := A[x].prvni;
    while dalsi.next ≠ nil do dalsi := dalsi.next;
    new(dalsi.next); (dalsi.next).num := y;
    if A[y] = nil then begin         { to same opacne }
      new(A[y]); A[y].num := y;
      new(A[y].prvni);
    end;
    dalsi := A[y].prvni;
    while dalsi.next ≠ nil do dalsi := dalsi.next;
    new(dalsi.next); (dalsi.next).num := x;
  end;
  for I := 1 to N do begin A[I].mafie := False; end;
  for I := 1 to P do begin read(F, x); A[x].mafie := True; end;
  close(F);
  for I := 1 to N do begin x := 0;
    dalsi := A[I].prvni; dalsi := dalsi.next;
    while dalsi ≠ nil do begin dalsi := dalsi.next; Inc(x); end;
    A[I].stupen := x;
  end;
  for I := 1 to N do A[I].exist := True;
end;

function Najdi_souseda(var dom : ukvrch) : ukvrch;
var dalsi : uksez;

```

```

begin Najdi_souseda := nil;
  dalsi := dom^.prvni; dalsi := dalsi^.next;
  while dalsi ≠ nil do begin
    if A[dalsi^.num]^.exist then begin Najdi_souseda := A[dalsi^.num]; break; end;
    dalsi := dalsi^.next;
  end;
end;

procedure Posli(x : integer; var soused : ukvrch);
begin
  Inc(soused^.stupen, -1); Inc(soused^.prechod, x);
end;

function Zpracuj : Boolean;
var I : longint; soused : ukvrch;
begin
  Zpracuj := False;
  for I := 1 to N do if A[I]^.exist then begin    { pro vsechhy vrcholy, co existuji: }
    if A[I]^.stupen = 1 then begin                { pro kazdy vrchol se stupnem 1: }
      soused := Najdi_souseda(A[I]);              { najde posl. aktivniho souseda }
      if soused ≠ nil then begin
        if A[I]^.prechod > 1 then begin Inc(Police); Posli(0, soused); end;
        if (A[I]^.prechod = 1) then begin
          if (A[I]^.mafie) then begin Inc(Police); Posli(0, soused); end
          else begin Posli(1, soused); end;
        end;
        if (A[I]^.prechod = 0) then
          if (A[I]^.mafie) then Posli(1, soused) else Posli(0, soused);
        Zpracuj := True;
        A[I]^.exist := False;
      end
    end;
    if A[I]^.stupen = 0 then begin
      if A[I]^.prechod > 1 then Inc(Police);
      if (A[I]^.prechod = 1) and (A[I]^.mafie) then Inc(Police);
      A[I]^.exist := False;
    end;
  end;
end;

procedure Vypis;
var F : text;
begin
  assign(F, 'policie.out');
  rewrite(F);
  write(F, Police);
  close(F);
end;

begin
  Nacti_Strom;
  while Zpracuj do begin end;
  writeln(Police);
  Vypis;
  readln;
end.

```