

ZASTOSOWANIE KOMPUTEROWYCH SYSTEMÓW GRAFICZNYCH DO LOGICZNEJ MINIMALIZACJI DRZEW DECYZYJNYCH

Marian A. PARTYKA, Adam NIESŁONY, Politechnika Opolska

Minimalizacja dwuwartościowych funkcji logicznych może prowadzić do drzewa logicznego, którego struktura nie jest ciągła. W takiej sytuacji nie jest możliwe zastosowanie klasyfikatorów drzewiastych. Dlatego należy najpierw wykonać równoważne przekształcenia informacyjne, aby minimalna postać dwuwartościowej funkcji logicznej była opisana ciągłą strukturą.

Poniżej przedstawiono fragment wydruku komputerowego minimalizacji drzew decyzyjnych.

```
{ $A+, B-, D+, E+, F-, G-, I+, L+, N-, O-P-, Q-, R-, S+, T-, V+, X+, Y+ }
{ $M 16384, 0, 655360 }
Uses Crt;
const MaxDIWiersza=8; MaxDIKolumny=50;
type TypDanych=record
  D:array[1..MaxDIKolumny] of array[1..MaxDIWiersza] of Byte;
  Klasa:array[1..MaxDIKolumny] of Byte;
  Upr:array[1..MaxDIKolumny] of Byte;
  Odz:array[1..MaxDIKolumny] of Boolean;
  DIW, DIK:Integer; end;
  TypNazwy=array[0..MaxDIWiersza-1] of Byte;
var Dane: TypDanych; Nazwy: TypNazwy; St:String;
    T: Text; NrUprosz:Integer;
function NewFileExt(BaseFile, NewExt:String):String;
  var i:Byte
begin
  for i:=1 to 8 do
    begin if (BaseFile[i]='.') or (BaseFile[i]=' ') then Break; end;
    Delete(BaseFile, i, Length(BaseFile)-i+1);
    NewFileExt:=BaseFile+'.'+NewExt[1]+NewExt[2]+NewExt[3];
  end;
function FileExists(FileName:String):Boolean;
  { Boolean function that returns True if the file exists;
    otherwise, it returns False. Closes the file if it exists. }
  var F:File;
  begin
    {$I-} Assign(F, FileName); FileMode:=0; Reset(F); Close(F); {$I+}
    FileExists:=(IORResult=0) and (FileName<>''); end {File Exists}
Procedure InitProgram;
  var DaneStr:String; i, j, Code:Integer;
  begin ClrScr; Writeln; Writeln('*** Minimalizacja indywidualnych
  czstkowych funkcji logicznych ***');
  Istnieje możliwość wprowadzenia dalszych uogólnień z punktu widzenia logik
  wielowartościowych i informatyki.
```