27. вариант

Developed by Deniska

1) Начертите неориентированный связанный граф, содержащий 5 вершин 10 рёбер.

2) В текстовом файле подготовьте список рёбер данного графа.

3) Напишите процедуру ввода данного списка и формирования его в виде динамической структуры линейного односвязного списка.

4) Напишите процедуру вывода данного списка на экран.

5) Напишите процедуру проверки, является ли данный граф связанным.

2

1

3

5

4

5 10

1 2

1 3

1 4

1 5

2 3

2 4

2 5

3 4

3 5

4 5

КОД ПРОГРАММЫ

program Graph;

{$APPTYPE CONSOLE}

uses

SysUtils;

Type

uk = ^spisok;

spisok = record

x, y: integer;

Next: uk;

end;

dovsh=array[1..10,1..10]of byte;

Var

Top, Bottom: uk;

i,j,n,m,kol:integer;

{ Добавление в спискок }

Procedure put\_in(Var B,T:uk; x,y:integer);

Var

u:uk;

Begin

new(u);

u^.x:=x;

u^.y:=y;

u^.Next:=Nil;

if B=nil then

B:=u

else

T^.next:=u;

T:=u;

End;

// ввод

Procedure Vvod(filename:string);

Var

f:textfile;

x,y:integer;

Begin

Bottom:=NIl;

Top:=Nil;

assignfile(f,filename); reset(f);

read(f,n,m);

for i:=1 to m do

Begin

read(f,x,y);

put\_in(Bottom, Top, x,y);

end;

closefile(f);

End;

// вывод

Procedure Vivod(u:uk);

begin

while u<>nil do

begin

writeln(u^.x,' ',u^.y);

u:=u^.next;

end;

end;

// создадим байтовый список смежностей

Procedure create\_spisok\_smeznostei(Var a:dovsh; u:uk);

Var

i,j:integer;

Begin

for i:=1 to n do

for j:=1 to n do

a[i,j]:=0;

while u<>nil do

begin

a[u^.x,u^.y]:=1;

a[u^.y,u^.x]:=1;

u:=u^.Next;

end;

End;

// обход в ширину

Procedure obhod\_v\_shirinu;

Var

a:dovsh;

visited:array[1..10]of boolean;

q:array[1..10]of integer;

x,pis,pil:integer;

Begin

create\_spisok\_smeznostei(a,Bottom);

for i:=1 to n do

begin

q[i]:=0;

visited[i]:=True;

end;

x:=Bottom^.x;

pis:=1;

pil:=1;

visited[x]:=False; // Дальше посетим превую вершину

q[pil]:=x; // Её мы запомнили в очереди посещенных вершин

while pis<=pil do // Находится ли в очереди ещё вершины для посещения ?

begin

for i:=1 to n do

if (a[x,i]<>0)and visited[i] then // Исследована ли вершина ранее ?

begin

inc(pil);

q[pil]:=i; // Сдвинем индекс новых вершин в очереди на один

visited[i]:=False; // Каждую посещенную вершину мы тоже обозначаем в массиве visited

end;

inc(pis); // А дальше переходим к следующей посещенной вершине

x:=q[pis];

end;

for i:=1 to n do

if visited[i] then

begin

write('graph ne sviazan :-( sorry');

exit;

end;

write('graph sviazan! ;-) uhhuhuhu');

End;

begin

Vvod('input.txt'); // ввод

Vivod(Bottom); // вывод

obhod\_v\_shirinu;

readln;

readln;

end.