

**ĐỀ RA**

**Câu 1: (3,0 điểm) Đếm ký tự trong văn bản**

**DEMKT.PAS**

Cho một văn bản gồm  $N$  dòng. Các ký tự được lấy từ tập các chữ cái và chữ số.  
**Yêu cầu:** Tìm số lượng ký tự của dòng ngắn nhất, số lượng ký tự của dòng dài nhất và số lượng ký tự của văn bản.

**Dữ liệu vào:** Cho trong file văn bản DEMKT.INP, có cấu trúc như sau:

- Dòng 1: Ghi số nguyên dương  $N$  là số dòng của văn bản ( $1 \leq N \leq 100$ ).
- $N$  dòng tiếp theo: Mỗi dòng ghi một xâu gồm  $L$  ký tự ( $0 < L < 255$ ).

**Dữ liệu ra:** Ghi ra file văn bản DEMKT.OUT, theo cấu trúc như sau:

- Dòng 1: Ghi 3 số nguyên dương  $x$   $y$   $z$ . Trong đó:  $x$  là số lượng ký tự của dòng ngắn nhất;  $y$  là số lượng ký tự của dòng dài nhất,  $z$  là số lượng ký tự của văn bản. Các số được ghi cách nhau ít nhất một dấu cách.

**Ví dụ:**

DEMKT.INP	DEMKT.OUT
3	5 8 20
ThiHSG12	
Nam2008	
Vong2	

**Câu 2: (3,5 điểm) Sắp xếp dãy số**

**SAPDAY.PAS**

Cho dãy số gồm  $N$  phần tử có giá trị nguyên dương  $a_1, a_2, \dots, a_n$  sao cho các phần tử nếu có giá trị giống nhau thì giá trị đó xuất hiện trong dãy không quá 255 lần. Chẳng hạn: Trong dãy số 2 5 2 4 3 8 5 2

Những phần tử có giá trị bằng 2 xuất hiện trong dãy số không quá 255 lần.

Những phần tử có giá trị bằng 5 xuất hiện trong dãy số không quá 255 lần.

**Yêu cầu:** Hãy sắp xếp dãy số theo thứ tự không giảm.

**Dữ liệu vào:** Cho trong file văn bản SAPDAY.INP, có cấu trúc như sau:

- Dòng 1: Ghi số nguyên dương  $N$  ( $1 \leq N \leq 32000$ ).
- Dòng 2: Ghi  $N$  số nguyên dương  $x_i$  là giá trị của các phần tử trong dãy. Các số được ghi cách nhau ít nhất một dấu cách ( $0 < x_i \leq 32000; 1 \leq i \leq N$ ).

**Dữ liệu ra:** Ghi ra file văn bản SAPDAY.OUT, theo cấu trúc như sau:

- Dòng 1: Ghi  $N$  số nguyên dương là giá trị của  $N$  phần tử trong dãy sau khi đã sắp xếp. Các số được ghi cách nhau ít nhất một dấu cách.

**Ví dụ:**

SAPDAY.INP	SAPDAY.OUT
8	2 2 2 3 4 5 5 8
2 5 2 4 3 8 5 2	

Giới hạn thời gian thực hiện chương trình không quá 0,5 giây đối với mọi trường hợp của dữ liệu vào.

**Câu 3: (3,5 điểm) Tiếp nhiên liệu cho tàu trên biển****TIEPNL.PAS**

Trên một vùng biển có N đảo. Các đảo được đánh số từ 1 đến N. Đảo i có tọa độ  $(x_i, y_i)$ . Trên mỗi đảo có một trạm tiếp nhiên liệu.

Một tàu thủy với một bình chứa đầy nhiên liệu chỉ có khả năng đi được một quãng đường tối đa là L km. Vì vậy, để di chuyển giữa hai đảo bất kỳ, có thể thuyền trưởng phải cho tàu ghé vào một số đảo nào đó để tiếp đầy nhiên liệu.

**Yêu cầu:** Hãy tìm một hành trình của tàu thủy xuất phát từ đảo P đến đảo Q sao cho số lần ghé vào các đảo để tiếp nhiên liệu là ít nhất (biết rằng tàu thủy được tiếp đầy nhiên liệu ở đảo P; Các trạm tiếp nhiên liệu luôn có đủ để tiếp đầy bình chứa của tàu thủy).

**Dữ liệu vào:** Cho trong file văn bản TIEPNL.INP có cấu trúc như sau:

- *Dòng 1:* Ghi 4 số nguyên dương N L P Q. Trong đó: N là số lượng đảo trên biển; L là độ dài quãng đường tối đa mà tàu thủy có thể đi khi bình chứa đầy nhiên liệu; P là đảo xuất phát, Q là đảo kết thúc của hành trình. Các số được ghi cách nhau ít nhất một dấu cách ( $2 < N \leq 100$ ;  $1 \leq L \leq 1000$ ;  $1 \leq P, Q \leq N$ ).

- *N dòng tiếp theo:* Trên mỗi dòng ghi hai số nguyên  $x_i y_i$  là tọa độ của đảo thứ i. Các số được ghi cách nhau ít nhất một dấu cách ( $-30000 \leq x_i y_i \leq 30000$ ).

**Dữ liệu ra:** Ghi ra file văn bản TIEPNL.OUT theo cấu trúc như sau:

Nếu không có hành trình để tàu thủy đi từ đảo P đến đảo Q thì

- *Dòng 1:* Ghi số -1

Nếu có hành trình để tàu thủy đi từ đảo P đến đảo Q thì

- *Dòng 1:* Ghi số nguyên dương M là số lần ghé đảo ít nhất để tiếp nhiên liệu (kể cả đảo P và đảo Q).

- *Dòng 2:* Ghi M số nguyên dương là chỉ số của M đảo mà tàu thủy phải ghé vào để tiếp nhiên liệu theo thứ tự từ đảo P đến đảo Q. Các số ghi cách nhau ít nhất một dấu cách.

**Ví dụ:**

TIEPNL.INP	TIEPNL.OUT
5 3 1 5	3
1 1	1 3 5
6 2	
4 1	
3 6	
4 4	

**==HẾT==**

**HƯỚNG DẪN CHẤM THI CHỌN HỌC SINH GIỎI LỚP 12**  
**NĂM HỌC 2008 - 2009**  
**MÔN TIN HỌC - VÒNG II**

**I/ Phương pháp:**

- Giám khảo tạo bộ dữ liệu vào, tính toán kết quả. Thực hiện chương trình của học sinh và so sánh kết quả.
- Giám khảo có thể sử dụng chương trình gợi ý dưới đây để tính toán kết quả của dữ liệu vào.
- Chương trình học sinh đúng một bộ test, giám khảo cho 0,5 điểm. Như vậy, nếu một bài có 3 điểm thì giám khảo phải tạo 6 bộ test.
- Với mỗi bài, giám khảo phải tạo được một bộ test có không gian dữ liệu lớn.
- Đối với bài 3 có thể có nhiều phương án ghé đảo với số lần ít nhất. Nếu học sinh đưa ra phương án khác với đáp án nhưng vẫn đúng thì giám khảo cho điểm tối đa.

**II/ Chương trình gợi ý:**

**Câu 1: Đếm ký tự trong văn bản**

```
Program Dem_ky_tu;
Const fi='DEMKT.inp';
      fo='DEMKT.out';
      MaxN=100;
Type mmc = Array[1..MaxN] of String;
Var A:mmc;
     N,Min,Max:Byte;Sum:Word;
Procedure Doc;
  Var f:Text;
      i:Byte;
  Begin
    Assign(f,fi);
    Reset(f);
    Readln(f,N);
    For i:= 1 to N do
      Readln(f,A[i]);
    Close(f);
  End;
Procedure Xuly;
  Var i:Byte;
  Begin
    Min:=255;
    Max:=0;
    Sum:=0;
    For i:=1 to N do
      Begin
        Sum := Sum + Length(A[i]);
        If Min > Length(A[i]) then
          Min := Length(A[i]);
        If Max < Length(A[i]) then
          Max := Length(A[i]);
      End;
  End;
Procedure Ghi;
  Var f:Text;
  Begin
    Assign(f,fo);
    Rewrite(f);
    Writeln(f,Min,' ', Max,' ', Sum);
    Close(f);
  End;
```

```
BEGIN Doc; Xuly; Ghi; END.
```

**Câu 2: Sắp xếp dãy số**

```
Program Sap_day;
Const fi='SAPDAY.inp';
      fo='SAPDAY.out';
```

```

MaxN=32000;
Type mmc = Array[1..MaxN] of Byte;
Var A:mmc;
N:word;
Procedure Doc;
Var f:Text;
i:word; so:Word;
Begin
Fillchar(A,sizeof(A),0);
Assign(f,fi);
Reset(f);
Readln(f,N);
For i:=1 to N do
Begin
Read(f,so);
A[so]:= A[so] + 1;
End;
Close(f);
End;
Procedure Ghi;
Var i,j:Word;
f:Text;
Begin
Assign(f,fo);
ReWrite(f);
For i:=1 to MaxN do
For j:=1 to A[i] do Write(f,i,' ');
Close(f);
End;
BEGIN Doc; Ghi; END.

```

### **Câu 3: Tiếp nhiên liệu cho tàu trên biển**

```

Program Tiep_Nhien_Lieu;
Const fi='TIEPNL.in0';
fo='TIEPNL.ou0';
MaxN=200;
Type mhc = Array[1..MaxN,0..MaxN] of Byte;
mmc = Array[1..MaxN] of Byte;
Var N,P,Q,Left,Right,Dem:Byte;L:longint;
A:mhc;
Tr,D,Queue:mmc;
Procedure Doc;
Var i,j:Byte;
f:Text;
T:Array[1..MaxN] of Record x,y:Integer; End;
t1,t2,t3:real;
Begin
Fillchar(Tr,sizeof(Tr),0);
Fillchar(D,sizeof(D),0);
Assign(f,fi);
Reset(f);
Readln(f,N,L,P,Q);
For i:=1 to N do
Begin
Readln(f,T[i].x,T[i].y);
End;
For i:=1 to N - 1 do
For j:=i+1 to N do
Begin
t1:=(T[j].x-T[i].x);
t2:=(T[j].y-T[i].y);
If sqrt(t1*t1+t2*t2) <= 1 then
Begin
A[i,j]:=1;
A[j,i]:=1;
end
Else
Begin
A[i,j]:=0;
A[j,i]:=0;
End;
End;
Close(f);

```

```

End;
Procedure Q_Push(so:Byte);
Begin
  If Right mod MaxN + 1 = Left then
    Writeln('Queue day')
  Else
    Begin
      Right:= Right mod MaxN + 1;
      Queue[Right]:=so;
      If Left = 0 then Left := 1;
    End;
  End;
End;
Function Q_Pop:Byte;
Begin
  If Left = 0 then
    Writeln('Queue rong')
  Else
    Begin
      Q_Pop:=Queue[Left];
      Left:=Left mod MaxN + 1;
      If Right mod MaxN + 1 = Left then
        Begin
          Left:=0;
          Right:=0;
        End;
    End;
  End;
End;
Procedure Xuly;
Var i,j:Byte;
Begin
  Left:=0;
  Right:=0;
  Q_Push(P);
  D[P]:=1;
  Dem:=1;
  While Left > 0 do
    Begin
      i:=Q_Pop;
      For j:=1 to n do
        If (A[i,j] = 1) and (D[j]=0) then
          Begin
            Q_Push(j);
            D[j]:=1;
            Tr[j]:=i;
            inc(Dem);
          End;
        End;
    End;
End;
Procedure Ghi;
Var f:Text;
    Luu:mmc;
    Top,i:Byte;
Begin
  Top:=0;
  Assign(f,fo);
  Rewrite(f);
  If Dem < N then
    Writeln(f,-1)
  Else
    Begin
      While Q<>0 do
        Begin
          inc(Top);
          Luu[Top]:=Q;
          Q:=Tr[Q];
        End;
      Writeln(f,Top);
      For i:=Top Downto 1 do
        Write(f,Luu[i],' ');
      End;
      Close(f);
    End;
End;
BEGIN Doc; Xuly; Ghi; END.

```