

Bài toán dãy con đơn điệu tăng dài nhất

Cho dãy số nguyên $A = a_1, a_2, \dots, a_n$. ($n \leq 10000, -10000 \leq a_i \leq 10000$). **Một dãy con của A là một cách chọn ra trong A một số phần tử giữ nguyên thứ tự.** Như vậy A có 2^n dãy con.

Yêu cầu: Tìm dãy con đơn điệu tăng của A có độ dài lớn nhất.

Ví dụ: $A = (1, 2, 3, 4, 9, 10, 5, 6, 7, 8)$. Dãy con đơn điệu tăng dài nhất là: $(1, 2, 3, 4, 5, 6, 7, 8)$.

Dữ liệu (**Input**) vào từ file văn bản INCSEQ.INP

- Dòng 1: Chứa số n
- Dòng 2: Chứa n số a_1, a_2, \dots, a_n cách nhau ít nhất một dấu cách

Kết quả (**Output**) ghi ra file văn bản INCSEQ.OUT

- Dòng 1: Ghi độ dài dãy con tìm được
- Các dòng tiếp: ghi dãy con tìm được và chỉ số những phần tử được chọn vào dãy con đó.

INCSEQ.INP	INCSEQ.OUT
11	8
1 2 3 8 9 4 5 6	$a[1] = 1$
20 9 10	$a[2] = 2$
	$a[3] = 3$
	$a[6] = 4$
	$a[7] = 5$
	$a[8] = 6$
	$a[10] = 9$
	$a[11] = 10$

Cách giải:

Bổ sung vào A hai phần tử: $a_0 = -\infty$ và $a_{n+1} = +\infty$. **Khi đó dãy con đơn điệu tăng dài nhất chắc chắn sẽ bắt đầu từ a_0 và kết thúc ở a_{n+1} .**

Với $\forall i: 0 \leq i \leq n + 1$. Ta sẽ tính $L[i]$ = độ dài dãy con đơn điệu tăng dài nhất bắt đầu tại a_i .

1. Cơ sở quy hoạch động (bài toán nhỏ nhất):

$L[n+1]$ = Độ dài dãy con đơn điệu tăng dài nhất bắt đầu tại $a_{n+1} = +\infty$. Dãy con này chỉ gồm mỗi một phần tử $(+\infty)$ nên $L[n+1]=1$.

2. Công thức truy hồi:

Giả sử với i từ n đến 0 , ta cần tính $L[i]$: độ dài dãy con tăng dài nhất bắt đầu tại a_i . $L[i]$ được tính trong điều kiện $L[i + 1]$, $L[i + 2]$, ..., $L[n + 1]$ đã biết:

Dãy con đơn điệu tăng dài nhất bắt đầu từ a_i sẽ được thành lập bằng cách lấy a_i ghép vào đầu một trong số những dãy con đơn điệu tăng dài nhất bắt đầu tại vị trí a_j đứng sau a_i . Ta sẽ chọn dãy nào để ghép a_i vào đầu? Tất nhiên là chỉ được ghép a_i vào đầu những dãy con bắt đầu tại a_j nào đó lớn hơn a_i (để đảm bảo tính tăng) và dĩ nhiên ta sẽ chọn dãy dài nhất để ghép a_i vào đầu (để đảm bảo tính dài nhất). Vậy $L[i]$ được tính như sau: **Xét tất cả các chỉ số j trong khoảng từ $i + 1$ đến $n + 1$ mà $a_j > a_i$, chọn ra chỉ số j_{\max} có $L[j_{\max}]$ lớn nhất. Đặt $L[i] := L[j_{\max}] + 1$.**

3. Truy vết

Tại bước xây dựng dãy L , mỗi khi tính $L[i] = L[j_{\max}] + 1$, ta đặt $T[i] = j_{\max}$. Để lưu lại rằng: Dãy con dài nhất bắt đầu tại a_i sẽ có phần tử thứ hai kế tiếp là $a_{j_{\max}}$.

Sau khi tính xong hay dãy L và T , ta bắt đầu từ 0 .

$T[0]$ là phần tử đầu tiên được chọn,

$T[T[0]]$ là phần tử thứ hai được chọn,

$T[T[T[0]]]$ là phần tử thứ ba được chọn ... Quá trình truy vết có thể diễn tả như sau:

```
i := T[0];
while i <> n + 1 do {Chừng nào chưa duyệt đến số  $a_{n+1}=+\infty$  ở cuối}
begin
  <Thông báo chọn  $a_i$ >
  i := T[i];
end;
```

Ví dụ: với $A = (5, 2, 3, 4, 9, 10, 5, 6, 7, 8)$. Hai dãy L và T sau khi tính sẽ là:

i	0	1	2	3	4	5	6	7	8	9	10	11
a_i	$-\infty$	5	2	3	4	9	10	5	6	7	8	$+\infty$
L[i]	9	5	8	7	6	3	2	5	4	3	2	1
T[i]	2	8	3	4	7	6	11	8	9	10	11	

Tracing → → → → → → → → → → → →

PROG03_1.PAS * Tìm dãy con đơn điệu tăng dài nhất

```

program LongestSubSequence;
const
  max = 10000;
var
  a, L, T: array[0..max + 1] of Integer;
  n: Word;
procedure Enter; {Nhập dữ liệu từ thiết bị nhập chuẩn theo đúng khuôn dạng Input}
var
  i: Word;
begin
  ReadLn(n);
  for i := 1 to n do Read(a[i]);
end;
procedure Optimize; {Quy hoạch động}
var
  i, j, jmax: Word;
begin
  a[0] := -32768; a[n + 1] := 32767; {Thêm hai phần tử canh hai đầu dãy a}
  L[n + 1] := 1; {Điền cơ sở quy hoạch động vào bảng phương án}
  for i := n downto 0 do {Tính bảng phương án}
    begin
      {Chọn trong các chỉ số j đứng sau i thoả mãn  $a_j > a_i$  ra chỉ số jmax có L[jmax] lớn nhất}
      jmax := n + 1;
      for j := i + 1 to n + 1 do
        if (a[j] > a[i]) and (L[j] > L[jmax]) then jmax := j;
      L[i] := L[jmax] + 1; {Lưu độ dài dãy con tăng dài nhất bắt đầu tại ai}
      T[i] := jmax; {Lưu vết: phần tử đứng liền sau ai trong dãy con tăng dài nhất đó}
    end;
end;

```

```

là ajmax}
end;
WriteLn(L[0] - 2); {Chiều dài dãy con tăng dài nhất}
i := T[0]; {Bắt đầu truy vết tìm nghiệm}
while i <> n + 1 do
begin
  WriteLn('a[', i, '] = ', a[i]);
  i := T[i];
end;
end;
begin
  {Định nghĩa lại thiết bị nhập/xuất chuẩn}
  Assign(Input, 'INCSEQ.INP'); Reset(Input);
  Assign(Output, 'INCSEQ.OUT'); Rewrite(Output);
  Enter;
  Optimize;
  Close(Input); Close(Output);
end.

```

Cài đặt bằng ngôn ngữ C++

```

#include <iostream>
#include <fstream>
using namespace std;

#define Input "INCSEQ.INP"
#define Output "INCSEQ.OUT"

int main()
{
  int A[10002], L[10002], T[10002], n;

  ifstream fi(Input);
  fi >> n;
  for (int i=1; i<=n; i++)
    fi >> A[i];
  A[0] = -(1<<31);
  A[n+1] = (1<<31)-1;
  fi.close();

```

```

for (int i=0; i<=n; i++) L[i]=0;
L[n+1]=1;
for (int i=n; i>=0; i--)
{
    int jmax=i;
    for (int j=i+1; j<=n+1; j++)
        if (A[i]<A[j] && L[j]>L[jmax])
            jmax=j;
    L[i]=L[jmax]+1;
    T[i]=jmax;
}

ofstream fo(Output);
fo<<L[0]-1<<endl;
int i=0;
while (T[i]<n+1)
{
    fo<<"A["<<T[i]<<"]="<<A[T[i]]<<endl;
    i=T[i];
}
fo.close();
}

```