

[首页](#)[题库](#)[训练](#)[比赛](#)[作业](#)[讨论](#)[评测记录](#)[排名](#)

wz2023028 ▾


2 条题解

0

 2023宗文3班洪茉致 (wz2023028) LV 1

发表您的题解

2

 2023宗文2班冯子羿 (wz2023005) LV 7 @ 1 周前

How To Do

本题分享一下乱搞做法。开结点记录当前位置及之前的最长长度的最长序列。
使用简单 dp 即可。

建议在 [一本通题库](#), [OpenJudge](#) 和 [Luogu/CF10D LCIS](#) 上尝试做本题因为没有 SPJ

这里提供一组数据：

INPUT

```
9
2 3 7 4 6 7 0 8 2
8
3 2 9 5 6 7 3 8
```

Copy

OUTPUT

```
4
3 6 7 8
```

Copy

DEBUG INFO 讲解用

```
[LINK] i:1, j:2, length:1
[INFO] list:2
```

Cop 🔍

[首页](#)[题库](#)[训练](#)[比赛](#)[作业](#)[讨论](#)[评测记录](#)[排名](#)

```

[INFO] list:2
[LINK] i:2, j:7, length:2
[INFO] list:2 3
[FIND] i:3, j:1, length:1
[INFO] list:3
[LINK] i:3, j:6, length:2
[INFO] list:3 7
[FIND] i:3, j:7, length:2
[INFO] list:2 3
[FIND] i:4, j:1, length:1
[INFO] list:3
[FIND] i:4, j:7, length:2
[INFO] list:2 3
[FIND] i:5, j:1, length:1
[INFO] list:3
[LINK] i:5, j:5, length:2
[INFO] list:3 6
[FIND] i:5, j:7, length:2
[INFO] list:2 3
[FIND] i:6, j:1, length:1
[INFO] list:3
[FIND] i:6, j:5, length:2
[INFO] list:3 6
[LINK] i:6, j:6, length:3
[INFO] list:3 6 7
[FIND] i:8, j:1, length:1
[INFO] list:3
[FIND] i:8, j:5, length:2
[INFO] list:3 6
[FIND] i:8, j:6, length:3
[INFO] list:3 6 7
[LINK] i:8, j:8, length:4
[INFO] list:3 6 7 8
[LINK] i:9, j:2, length:1
[INFO] list:2

```

wz2023028 ▾

Code

无储存

```

#include <iostream>
#define reg register
using namespace std;

int dp[510];
int a[510], b[510];
int ans = -114514;
int n1, n2;

int main()
{
    scanf("%d", &n1);
    for (int i = 1; i <= n1; ++i)
        scanf("%d", &a[i]);
    scanf("%d", &n2);

```

Copy



[首页](#)[题库](#)[训练](#)[比赛](#)[作业](#)[讨论](#)[评测记录](#)[排名](#)

```

for (reg int i = 1; i <= n1; ++i)
{
    int p = 0;
    for (reg int j = 1; j <= n2; ++j)
    {
        // 寻找[上升]的[更长]的序列
        if (a[i] > b[j])
            p = max(p, dp[j]);

        // 我诶诶这就是说狗尾续貂是么
        if (a[i] == b[j])
            dp[j] = p + 1;
    }
}

// 寻找答案
for (reg int i = 1; i <= n2; ++i)
    ans = max(ans, dp[i]);
printf("%d\n", ans);
return 0;
}

```

wz2023028 ▾

存储内容

[Copy](#)

```

#include <iostream>

#include <vector>

#define longint long long
#define reg register
#define qwq puts("fzy qwq ~");
#define pb push_back

#define FINDDEBUG printf("[FIND] i:%d, j:%d, length:%d \n[INFO] list
#define LINKDEBUG printf("[LINK] i:%d, j:%d, length:%d \n[INFO] list

using namespace std;

struct Node
{
    int len;
    vector<int> now;
} dp[510];

int n1, n2, a[510], b[510];

int main()
{
    // freopen("subseq.in", "r", stdin);
    // freopen("subseq.out", "w", stdout);

    scanf("%d", &n1);
    for (int i = 1; i <= n1; ++i)
        scanf("%d", &a[i]);
    scanf("%d", &n2);

```



[首页](#)[题库](#)[训练](#)[比赛](#)[作业](#)[讨论](#)[评测记录](#)[排名](#)

```

// start to do the dynamic programing
// no god please no i do not want to do this
for (reg int i = 1; i <= n1; ++i)
{
    Node p; p.len = 0;
    for (reg int j = 1; j <= n2; ++j)
    {
        // what the f**king equaltion ahhhhhhhhhhhhhhhhhhhh
        // it takes me two thousand years to remember this s**t
        // O(n^2) yyds! go to the hell O(nlogn).

        // 寻找[上升]的[更长]的序列
        // 你是一个一个一个公式啊啊啊啊啊
        if (a[i] > b[j] && dp[j].len > p.len)
        {
            p = dp[j];

            // FINDEBUG
        }

        // 我愣愣这就是说狗尾续貂是么
        if (a[i] == b[j])
        {
            dp[j].len = p.len + 1;
            dp[j].now = p.now;
            dp[j].now.pb(a[i]);

            // LINKDEBUG
        }
    }
}

// 寻找答案
Node ans = dp[1];
for (reg int i = 2; i <= n2; ++i)
{
    if (dp[i].len > ans.len)
        ans = dp[i];
}
printf("%d\n", ans.len);
for (auto x : ans.now)
    printf("%d ", x);
return 0;
}

```

wz2023028 ▾

 2023宗文1班陈骏翔 (wz2023002) @ 1 周前

%%%

0



admin LV 4 SU @ 1 周前



[首页](#)[题库](#)[训练](#)[比赛](#)[作业](#)[讨论](#)[评测记录](#)[排名](#)

wz2023028 ▾

```
#include<cstdio>
#include<cstring>
#include<algorithm>
#include<string>
#define INF 0x3f3f3f3f
#define PI acos(-1.0)
#define N 501
#define MOD 100001
#define E 1e-12
using namespace std;
struct Node{
    int len;
    int ans[N];
}order[N],now;
int a[N],b[N];
int main(){
    freopen("subseq.in","r",stdin);
    freopen("subseq.out","w",stdout);
    int n,m;
    cin>>n;
    for(int i=1;i<=n;++i)
        cin>>a[i];
    cin>>m;
    for(int i=1;i<=m;++i)
        cin>>b[i];

    for(int i=1;i<=m;++i)
    {
        now.len=0;
        memset(now.ans,0,sizeof(now.ans));

        for(int j=1;j<=n;++j)
        {
            if(a[j]<b[i]&&order[j].len>now.len)
                now=order[j];

            if(a[j]==b[i])
            {
                order[j]=now;
                order[j].len++;
                order[j].ans[order[j].len]=a[j];
            }
        }
    }

    int flag=0;
    int maxx=-INF;
    for(int i=1;i<=n;++i)
        if(order[i].len>maxx)
        {
            maxx=order[i].len;
            flag=i;
        }

    cout<<order[flag].len<<endl;
    for(int i=1;i<=order[flag].len;++i)
        cout<<order[flag].ans[i]<<" ";
    cout<<endl;
```





首页

题库

训练

比赛

作业

讨论

评测记录

排名

wz2023028 ▾

1

最长公共子上升序列

查看题目

递交

讨论

题解

文件

信息

ID

216

时间

1000ms

内存

128MiB

难度

9

标签

动态规划 一本通基础

递交数

54

已通过

6

上传者

 admin





首页

题库

训练

比赛

作业

讨论

评测记录

排名

wz2023028 ▾

状态

评测队列

服务状态

开发

开源

API

支持

帮助

QQ 群

