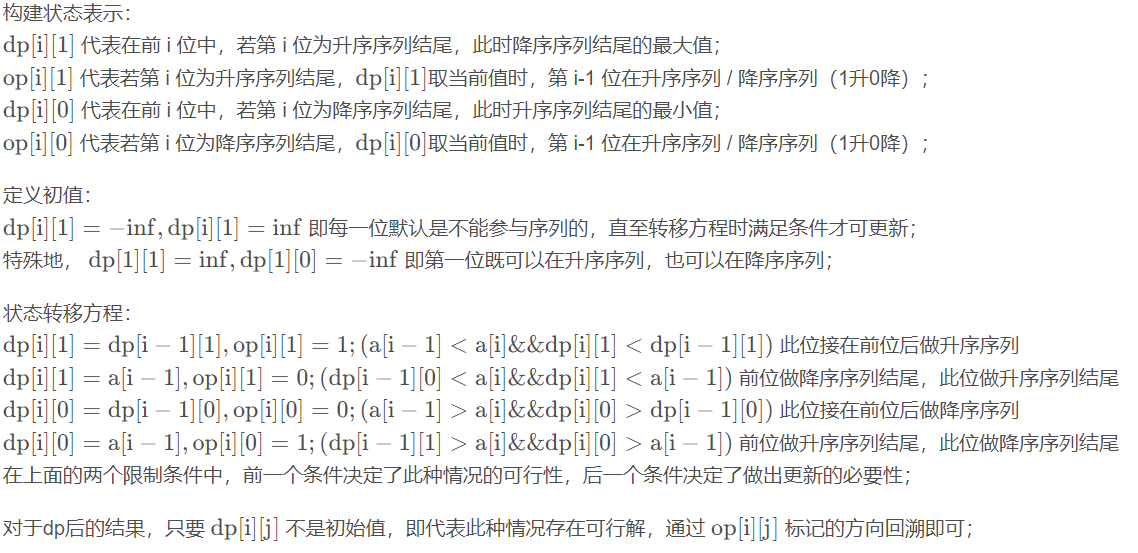
给一个序列，问是否能将这个序列完全划分成一个上升子序列和下降子序列#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

const int inf = 0x7f7f7f7f;

int a[200005];

int dp[200005][2];

int op[200005][2];

int opt[200005];

int main()

{

int n;

scanf("%d", &n);

for (int i = 1; i <= n; i++)

scanf("%d", &a[i]);

dp[1][1] = inf, dp[1][0] = -inf;

for (int i = 2; i <= n; i++)

{

dp[i][1] = -inf, dp[i][0] = inf;

if (a[i - 1] < a[i] && dp[i][1] < dp[i - 1][1])

{

dp[i][1] = dp[i - 1][1];

op[i][1] = 1;

}

if (dp[i - 1][0] < a[i] && dp[i][1] < a[i - 1])

{

dp[i][1] = a[i - 1];

op[i][1] = 0;

}

if (a[i - 1] > a[i] && dp[i][0] > dp[i - 1][0])

{

dp[i][0] = dp[i - 1][0];

op[i][0] = 0;

}

if (dp[i - 1][1] > a[i] && dp[i][0] > a[i - 1])

{

dp[i][0] = a[i - 1];

op[i][0] = 1;

}

}

if (dp[n][1] != -inf)

{

printf("YES\n");

for (int i = n, optmp = 1; i >= 1; i--)

{

opt[i] = optmp;

optmp = op[i][optmp];

}

for (int i = 1; i <= n; i++)

{

printf("%d ", opt[i] != 1);

}

}

else if (dp[n][0] != inf)

{

printf("YES\n");

for (int i = n, optmp = 0; i >= 1; i--)

{

opt[i] = optmp;

optmp = op[i][optmp];

}

for (int i = 1; i <= n; i++)

{

printf("%d ", opt[i] != 1);

}

}

else

printf("NO");

return 0;

}