Maximum product of increasing subsequence

> LIS变种——求乘积最大的LIS

```
#include <bits/stdc++.h>

using namespace std;


typedef long long ll;

int maxProductOfLIS(vector<int> &arr, int &N) {
    vector<ll> dp(N);
    ll maxProduct = dp[0];
    for (int i = 0; i < N; ++i) dp[i] = arr[i];
    for (int i = 1; i < N; ++i) {
        for (int j = 0; j < i; ++j) {
            if (arr[i] > arr[j] && dp[i] < dp[j] * arr[i])
                dp[i] = dp[j] * arr[i];
        }
        maxProduct = max(maxProduct, dp[i]);
    }
    return maxProduct;
}


int main() {
    int T;
    scanf("%d", &T);
    while (T--) {
        int N, num;
        vector<int> arr;
        scanf("%d", &N);
        for (int i = 0; i < N; ++i) {
            scanf("%d", &num);
            arr.push_back(num);
        }
        printf("%d\n", maxProductOfLIS(arr, N));
    }
    return 0;
}
```