Simple Yet Tricky

This question is implemented by stack data structure. Here stack will keep the record of the heights of the students, and then for every height greater than previous height the stack will be emptied.

For more reference - https://www.geeksforgeeks.org/the-stock-span-problem/

Code

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

using namespace std;

typedef unsigned long long ull;
typedef long long ll;
typedef long double ld;

#define fr(i, n) for (ll i = 0; i < n; i++)
#define fr1(i, n) for (ll i = 1; i <= n; i++)
#define pb(x) push_back(x)
#define l(s) s.size()
#define as(a) sort(a, a + n)
#define ds(a) sort(a, a + n, greater<int>())
#define vs(v) sort(v.begin(), v.end());
#define inf 1e18
#define nl cout << endl;
#define F first
#define S second
typedef vector<ll> vi;
typedef pair<ll, ll> pi;

int main()
{
    ios_base::sync_with_stdio(false);
    cin.tie(NULL);

    ll t;
    cin >> t;
    while (t--)
    {
        ll n;
        cin >> n;
        vi arr(n );
        fr(i, n)
```

```
        {
            cin >> arr[i];
        }
        stack<ll> st;
        st.push(0);
        vi result(n, 0);
        result[0] = 1;

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

            while (!st.empty() && arr[st.top()] <= arr[i]){
                st.pop();
            }

            result[i] = (st.empty()) ? (i + 1) : (i - st.top());
            st.push(i);
        }

        for (auto it : result)
        {
            cout << it << " ";
        }
        nl
    }

    return 0;
}

// dev
```