Lecture 29 Stack

1. <https://leetcode.com/problems/daily-temperatures/>

class Solution {

public:

vector<int> dailyTemperatures(vector<int>& T) {

int n = T.size();

stack<int> s;

vector<int> ans(n, 0);

for(int i=0; i<n; i++)

{

while(!s.empty() && T[i] > T[s.top()])

{

ans[s.top()] = i - s.top();

s.pop();

}

s.push(i);

}

return ans;

}

};

1. <https://practice.geeksforgeeks.org/problems/stock-span-problem-1587115621/1#>

class Solution{

public:

vector <int> calculateSpan(int price[], int n)

{

// Your code here

vector<int> span(n);

stack<int> s;

s.push(0);

span[0] = 1;

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

{

while(!s.empty() && price[s.top()] <= price[i])

{

s.pop();

}

span[i] = (s.empty()) ? (i+1) : (i-s.top());

s.push(i);

}

return span;

}

};

1. <https://practice.geeksforgeeks.org/problems/the-celebrity-problem/1#>

class Solution {

public:

int celebrity(vector<vector<int> >& M, int n) {

// code here

stack<int> s;

for(int i=0; i<n; i++)

{

s.push(i);

}

while(s.size()>1)

{

int i=s.top();

s.pop();

int j = s.top();

s.pop();

if(M[i][j] == 1)

{

s.push(j);

}

else

{

s.push(i);

}

}

int cand = s.top();

s.pop();

for(int i=0; i<n; i++)

{

if(i!=cand)

{

if(M[cand][i]==1 || M[i][cand]==0)

{

return -1;

}

}

}

return cand;

}

};