Maximum sum non adjacent

15 April 2022 17:19

https://nados.io/question/maximum-sum-non-adjacent-elements

7TH APRIL - SS

While making dp, see the last element and think accordingly.
If we want to include current element, than add it to

arr	1	2	10	20	4
Include	1	2+0 =2	1+10 =11	22	15
Exclude	0	1	2	11	22

```
// #include<iostream>
// #include<bits/stdc++.h>
// using namespace std;
// int main()
// {
//     string str;
//     getline(cin,str);

//     int dp[str.size()+1][2];
//     dp[0][0]=0;
//     dp[0][1]=0;
//     string s1;
//     s1= str[0];
//     int n1=stoi(s1);
//     dp[1][0]=n1;
//     dp[1][1]=0;

//     for(int i=2;i<=str.size();i++)
//     {
//         string inc;
//         inc= str[i-1];
//         //cout<<inc<<endl;
//         int num_inc = stoi(inc);
//         num_inc+=dp[i-1][1];
//         dp[i][0]=num_inc;

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

//     cout<<dp[str.size()][0]<<endl;
// }
```

Base case

Include the present element

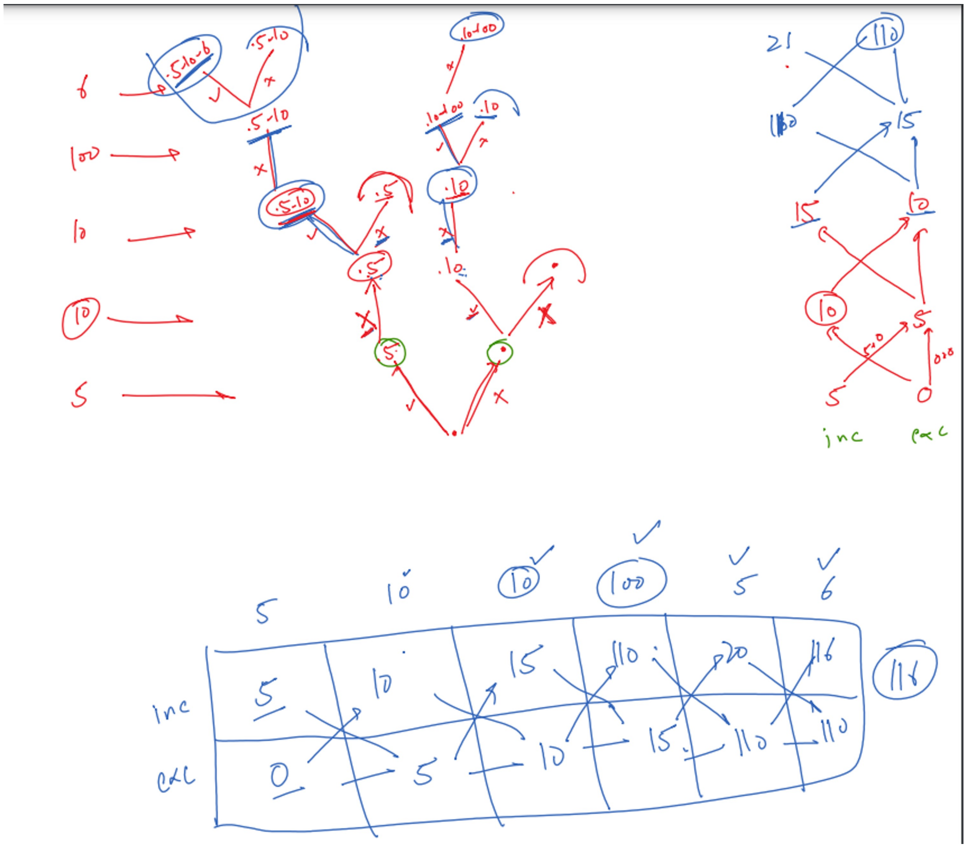
Exclude the present element.

```
#include<iostream>
#include<bits/stdc++.h>
using namespace std;
int main()
{
```

```
using namespace std;
int main()
{
    int n;
    cin>>n;

    int arr[n];
    for(int i=0;i<n;i++)
    {
        cin>>arr[i];
    }

    int dp[n+1][2];
    dp[0][0]=0;
    dp[0][1]=0;
    if(arr[0]>0){
    dp[1][0]=arr[0];
    dp[1][1]=0;
    }
    else
    {
        dp[1][0]=0;
    dp[1][1]=0;
    }
    for(int i=2;i<=n;i++)
    {
        if(arr[i-1]>0){
        int inc;
        inc= arr[i-1];

        int num_inc = inc;
        num_inc+=dp[i-1][1];
        dp[i][0]=num_inc;
        }
        else
        {
            int a = dp[i-1][0]>dp[i-1][1]?dp[i-1][0] : dp[i-1][1];
            dp[i][0]=a;
        }
        dp[i][1]=dp[i-1][0];
    }
    int a = dp[n][0]>dp[n][1]?dp[n][0] : dp[n][1];
    cout<<a<<endl;
    //for(int i=0;i<=n;i++)
    //{
      //  cout<<dp[i][0]<<"  "<<dp[i][1]<<endl;
    //}
}
```

DRY RUN

	5	10	10	100	5	6
inc	5 5	10 10	15 5 10	110 100 10	20 5 10 5	116 100 10 6
exc	0	5 5	10 10	15 5 10	110 100 10	110 100 10



	5	10	10	100	5	6	
inc	5	10	15	110	20	116	116
exc	0	5	10	15	110	110	