



Ques . Cities are arranged in linear order.. 1 to n. Each city contains some money. If you take some money from city x, then you are not allowed to take money from city x+1. In other words, just next city won't allow you to take any money. What is the maximum amount you can make, starting from city 1.

$1 \leq n \leq 10^5$, $1 \leq \text{money for each city} \leq 10^9$

Topdown Down Approach

// Assuming cities are one indexed. The approach is, we will try to find answer for every x to n. We will try to find answer for x-1 using answers for all the states from x to n.

```
int price[100010],n;
long long solve(int x){
    if (x>=n+1){ // note that n is global, and can be used here.
        The above becomes our base condition, in this example
        return 0;
    }
    if (dp[x]!=-1){
        return dp[x];
    }
    int ans=max(solve(x+1),price[x]+solve(x+2));
    dp[x]=ans;
    return ans;
}
```

```
int main(){
    scanf("%d",&n); // global n
    int i;
    for (i=1;i<=n;i++){
        scanf("%d",&price[i]); // read the values.
    }
    long long ans=solve(1);
    cout<<ans<<endl;
}
```

Bottom up Approach

```
ans[0]=0;
ans[1]=price[1];
for (i=2;i<=n;i++){
    ans[i]=max(ans[i-1],price[i]+ans[i-2]);
}
//ans[n] is your final answer
```