Experiment6

StudentName: Abith
UID:22BCS10634
Branch:BE-CSE
Section/Group:22BCS_NTPP_603'A'
Semester:6th
DateofPerformance:24-02-25
SubjectName:APLab-2
SubjectCode:22CSP-351

1. Aim:ClimbingStairs

2. Objective:

Youareclimbingastaircase.Ittakes n stepstoreachthetop.

Eachtimeyoucaneitherclimb1 or2 steps.Inhowmanydistinctwayscanyouclimb tothetop?

3. Implementation/Code:

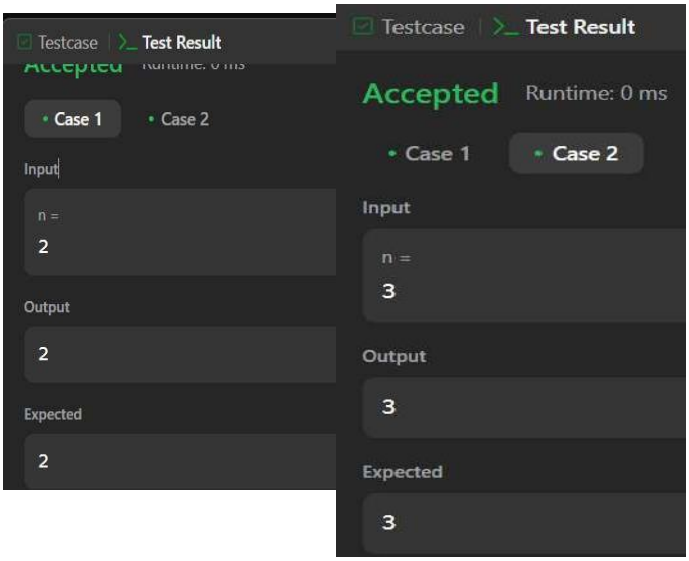
```
classSolution{
   publicintclimbStairs(intn){ if
      (n == 0) return 1;
      if(n == 1)return 1;

      int[]dp=newint[n+1];
      dp[0] = 1;
      dp[1]=1;

      for (int i = 2; i <= n; i++) {
         dp[i]=dp[i-1]+dp[i-2];
      }

      returndp[n];
   }
}
```

4. Output



5. LearningOutcome:

- Understanddynamicprogrammingtosolverecurrence-basedproblems.
- Learnhowtouseanarraytostoreintermediateresultsforoptimization.
- RecognizetheFibonacci-likenatureofthe"climbingstairs"problem.
- Implementiterativesolutionstoreduceredundantcalculations.
- Improveproblem-solvingskillswithbottom-updynamicprogramming

Question2.

1. Aim: MaximumSubarray

2. Objective:

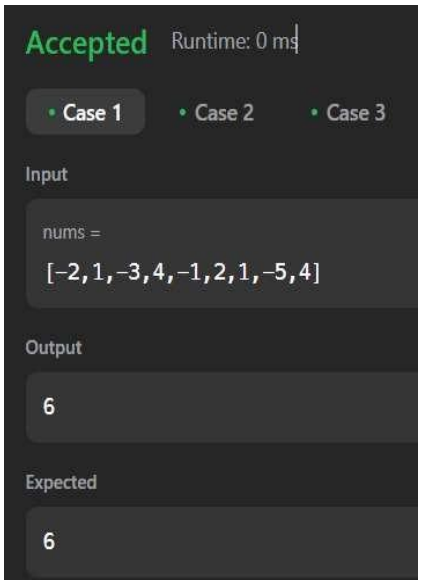
Givenanintegerarraynums,findthe subarray withthelargestsum,and return itssum .

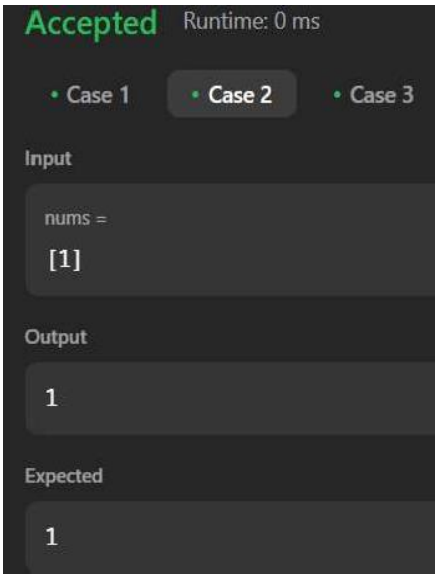
3. Implementation/Code:

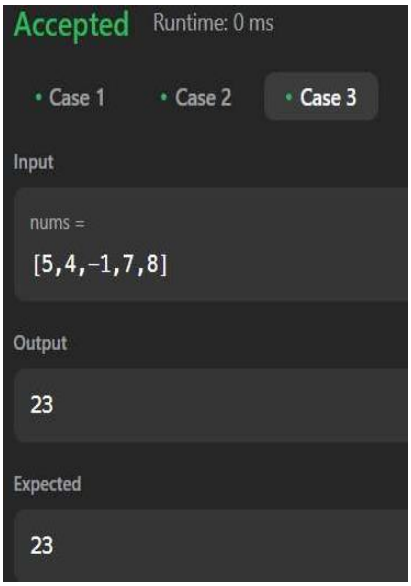
```
classSolution{
    publicintmaxSubArray(int[]nums){ int
        maxSum = nums[0];
        intcurrentSum=nums[0];
```

```
        for(inti=1;i<nums.length;i++){
            currentSum=Math.max(nums[i],currentSum+nums[i]);
            maxSum=Math.max(maxSum,currentSum);
        }
        returnmaxSum;
    }
}
```

4. Output







5. LearningOutcome:

- Understanddynamicprogrammingtosolverecurrence-basedproblems.
- Learnhowtouseanarraytostoreintermediateresultsforoptimization.
- RecognizetheFibonacci-likenatureofthe"climbingstairs"problem.
- Implementiterativesolutionstoreduceredundantcalculations.
- Improveproblem-solvingskillswithbottom-updynamicprogramming