Name: Roll Number:

ESO207: Data Structures and Algorithms

Programming Assignment 1

Due Date: 24th August 11:59pm, 2023

Total Number of Pages: 4 Total Points 100

Note :

- The questions have to be answered through a contest in Hackerrank. The contest has 3 challenges, each corresponding to a question. You have to submit your code through the contest. (Link will be circulated soon)
- Additionally you must upload your solutions on Moodle as well. You need to upload 3 files corresponding to the 3 programs.
- Your codes will be checked for possible plagiarism of any sorts. If we find such cases, then we will possibly award an F grade.
- Allowed Languages for challenge code submission : C, C++
- Allowed libraries : stdio.h for C and iostream for C++

Name: Rollno:

Question 1. (30 points) **Problem 1 - MaxCraft**

You have an array a of size n.
For every i in 1 to n-
Among all subarrays starting at index i you have to report the sum of maximum sum subarray.

- **Input**

  First line of input consists of number of test cases T. Every test case has two lines first line contains a single integer N , next line has N integers $A_1, A_2, \ldots, A_N$

- **Output**

  For each test case, output contains one line having N integers, i^{th} integer represents sum of maximum sum subarray starting at index i.

- **Constraints**

  - $1 \leq T \leq 10^4$
  - $1 \leq N \leq 10^5$
  - $-10^9 \leq A[i] \leq 10^9$

  sum of N over all test cases is less than $2 * 10^5$

- **Sample Input**

```
2
5
1 2 3 -5 3
7
1 2 3 4 5 6 -9
```

- **Sample Output**

```
6 5 3 -2 3
21 20 18 15 11 6 -9
```

Name: Rollno:

Question 2. (30 points) **Problem 2 - Bob the Destroyer**

Bob likes to destroy buildings.
There are N buildings in a city, i^{th} building has a_i floors. Bob can do the following operation.
Choose a building, and destroy its uppermost floor with cost h. Where h is building's height before removing the floor
You can do this operation any number of times but total cost should be less than or equal to K
Since you don't like tall buildings you want to decrease their heights.
You have to minimise the maximum height of buildings and report this height (height of tallest among them after operations).

- **Input**

  First line of input consists of number of test cases T. Every test case has two lines first line contains a two integers N and K , next line has N integers $A_1, A_2, \ldots, A_N$

- **Output**

  For each test case, output a line containing answer for that test case.

- **Constraints**

  - $1 \leq T \leq 10^3$
  - $1 \leq N \leq 10^5$
  - $1 \leq K \leq 10^{15}$
  - $0 \leq A[i] \leq 10^6$

  sum of N over all test cases is less than $2 * 10^5$

- **Sample Input**

```
3
5 23
1 3 2 4 5
5 3000000
0 1000000 2 5 99999
3 9
3 3 3
```

- **Sample Output**

```
2
999997
2
```

Name: Rollno:

Question 3. (40 points) **Problem 3 - k-dice ways**

You have a $k-dice$.
A $k-dice$ is a dice which have k faces and each face have value written from 1 to k.
Eg. A $6-dice$ is the normal dice we use while playing games.
For a given N, you have to calculate the number of ways you can throw this dice so that we get sum equal to N.
Since number of ways can be large you have to calculate ways mod 998244353
Refer to samples for better understanding.

- **Input**

  First line of input consists of number of test cases T. Every test case has one line which contains two integers N and K

- **Output**

  For each test case, output a line containing answer for that test case.

- **Constraints**

  - $1 \leq T \leq 10^2$
  - $1 \leq N \leq 10^2$ $(20\% points)$
  - $1 \leq N \leq 10^{18}$ $(100\% points)$
  - $2 \leq k \leq 20$

  You will get 20% points of whole problem if solved for $n <= 100$. You will recieve full points of the problem if solved for $n <= 10^{18}$

- **Sample Input**

```
3
5 3
7 4
8 10
```

- **Sample Output**

```
13
56
128
```

- **Explanation**

```
In first test case where n = 5 and k = 3
1+1+1+1+1=5
1+1+1+2=5
1+1+2+1=5
1+2+1+1=5
2+1+1+1=5
2+2+1=5
2+1+2=5
1+2+2=5
3+1+1=5
1+3+1=5
1+1+3=5
3+2=5
2+3=5
So there are 13 ways.
```