R-1 실습 문제

|  |
| --- |
| <설명>  한 마을에 모험가가 N명 있습니다. 모험가 길드에서는 N명의 모험가를 대상으로 '공포도'를 측정했는데, '공포도'가 높은 모험가는 쉽게 공포를 느껴 위험 상황에서 제대로 대처할 능력이 떨어집니다. 모험가 길드장인 동빈이는 모험가 그룹을 안전하게 구성하고자 공포도가 X인 모험가는 반드시 X명 이상으로 구성한 모험가 그룹에 참여해야 여행을 떠날 수 있도록 규정했습니다. 동빈이는 최대 몇 개의 모험가 그룹을 만들 수 있는지 궁금합니다.  동빈이를 위해 N명의 모험가에 대한 정보가 주어졌을 때, 여행을 떠날 수 있는 그룹 수의 최댓값을 구하는 프로그램을 작성하세요.  예를 들어, N = 5이고, 각 모험가의 공포도가 다음과 같다고 가정합시다.  2 3 1 2 2  이 경우 그룹 1에 공포도가 1,2,3인 모험가를 한 명씩 넣고, 그룹 2에 공포도가 2인 남은 두 명을 넣게 되면 총 2개의 그룹을 만들 수 있습니다. 또한 몇 명의 모험가는 마을에 그대로 남아 있어도 되기 때문에, 모든 모험가를 특정한 그룹에 넣을 필요는 없습니다. |
| <입력>  - 첫째 줄에 모험가의 수 N이 주어집니다. (1 <= N <= 100,000)  - 둘째 줄에 각 모험가의 공포도의 값을 N 이하의 자연수로 주어지며, 각 자연수는 공백으로 구분합니다. |
| <출력>  -여행을 떠날 수 있는 그룹 수의 최댓값을 출력합니다. |
| <예제>   |  |  | | --- | --- | | 입력  5  2 3 1 2 2 | 출력  2 | |

R-2 실습문제

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| <설명>  모든 원소가 양의 정수인 n크기 배열이 A가 있다고 가정하자. 배열 A의 마지막 원소는 모든 원소 보다 크다. 이때, 배열 B에 대한 정의는 아래와 같다.  배열 B의 i번째 원소는 배열 A의 i+1번째부터 n번째까지의 원소 중에서 배열 A의 i번째 원소보다 큰 원소 중, 제일 작은 원소 값의 인덱스 값으로 정의한다. 이를 식으로 표현하면  B[i] = min { k : i<k and A[i]<A[k] }  이 된다.  예를 들어, 배열 A= {2,4,3,5,1,6}이라고 가정하면 아래와 같이 표를 만들 수 있다.   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | Index i | 0 | 1 | 2 | 3 | 4 | 5 | | A[i] | 2 | 4 | 3 | 5 | 1 | 6 | | B[i] | 1 | 3 | 3 | 5 | 5 |  |   배열 B를 출력하는 함수를 구하시오. |
| <입력>  - 첫째 줄에 배열A의 크기가 주어집니다.  - 둘째 줄에 배열 A의 원소를 공백으로 구분하여 넣습니다. |
| <출력>  - 첫째 줄에 배열 B의 원소를 공백으로 구분하여 출력합니다. |
| <예제>   |  |  | | --- | --- | | 입력  6  2 4 3 5 1 6 | 출력  1 3 3 5 5 | |

R-3 과제문제

|  |
| --- |
| <설명>  초기 역량값 X와 배열 [A, B]의 집합인 TaskSet이 주어질 때, A는 작업을 시작할 수 있는 최소 역량값, B는 작업 후 얻게 되는 역량 증가값이다. 즉, X ≥ A 이면 작업이 자동 시작되고, 그로 인해 X는 X+B로 바뀌게 된다.  이 때, X=0부터 시작한다고 가정했을 때 주어진 TaskSet으로 최대 얼마까지 역량이 상승될 수 있는 지를 구하시오. 예를 들어, 배열 [11, 4], [0, 3], [3, 5]가 주어졌을 때, 시작 값 0부터 값을 최대 8까지 올릴 수 있다 (0+3+5).  <제한 사항>  가능한 작업이 존재하지 않을 때 -1를 출력.  TaskSet의 각 배열 원소 B의 값은 1 이상. |
| <입력>  - 첫째 줄에 TaskSet의 길이가 주어진다.  - 둘째 줄에 TaskSet의 원소를 공백으로 구분하여 넣습니다. |
| <출력>  - 첫째 줄에 상승된 X의 최대값이 출력됩니다. |
| <예제>   |  |  | | --- | --- | | 입력 1  4  0 1 5 4 1 2 10 2 | 출력 1  3 | | 입력 2  4  2 3 2 2 0 2 6 4 | 출력 2  11 | | 입력 3  3  14 4 4 4 1 2 | 출력 3  -1 | |

R-4 과제문제

|  |
| --- |
| <설명>  초기 평소 식욕이 왕성한 무지는 자신의 재능을 뽐내고 싶어 졌고 고민 끝에 카카오 TV 라이브로 방송을 하기로 마음먹었다. 그냥 먹방을 하면 다른 방송과 차별성이 없기 때문에 무지는 아래와 같이 독특한 방식을 생각해냈다.  회전판에 먹어야 할 N 개의 음식이 있다. 각 음식에는 1부터 N 까지 번호가 붙어있으며, 각 음식을 섭취하는데 일정 시간이 소요된다. 무지는 다음과 같은 방법으로 음식을 섭취한다.  1. 무지는 1번 음식부터 먹기 시작하며, 회전판은 번호가 증가하는 순서대로 음식을 무지 앞으로 가져다 놓는다.  2. 마지막 번호의 음식을 섭취한 후에는 회전판에 의해 다시 1번 음식이 무지 앞으로 온다.  3. 무지는 음식 하나를 1초 동안 섭취한 후 남은 음식은 그대로 두고, 다음 음식을 섭취한다.  다음 음식이란, 아직 남은 음식 중 다음으로 섭취해야 할 가장 가까운 번호의 음식을 말한다.  회전판이 다음 음식을 무지 앞으로 가져오는데 걸리는 시간은 없다고 가정한다.  4. 무지가 먹방을 시작한 지 K 초 후에 네트워크 장애로 인해 방송이 잠시 중단되었다.  무지는 네트워크 정상화 후 다시 방송을 이어갈 때, 몇 번 음식부터 섭취해야 하는지를 알고자 한다. 각 음식을 모두 먹는데 필요한 시간이 담겨있는 배열 food\_times, 네트워크 장애가 발생한 시간 K 초가 매개변수로 주어질 때 몇 번 음식부터 다시 섭취하면 되는지 return 하도록 solution 함수를 완성하라.  입출력 예 설명  입출력 예 #1  0~1초 동안에 1번 음식을 섭취한다. 남은 시간은 [2,1,2] 이다.  1~2초 동안 2번 음식을 섭취한다. 남은 시간은 [2,0,2] 이다.  2~3초 동안 3번 음식을 섭취한다. 남은 시간은 [2,0,1] 이다.  3~4초 동안 1번 음식을 섭취한다. 남은 시간은 [1,0,1] 이다.  4~5초 동안 (2번 음식은 다 먹었으므로) 3번 음식을 섭취한다. 남은 시간은 [1,0,0] 이다.  5초에서 네트워크 장애가 발생했다. 1번 음식을 섭취해야 할 때 중단되었으므로, 장애 복구 후에 1번 음식부터 다시 먹기 시작하면 된다.  제한사항  -food\_times 는 각 음식을 모두 먹는데 필요한 시간이 음식의 번호 순서대로 들어있는 배열이다.  -k 는 방송이 중단된 시간을 나타낸다.  -만약 더 섭취해야 할 음식이 없다면 -1을 반환하면 된다.  정확성 테스트 제한 사항  -food\_times 의 길이는 1 이상 2,000 이하이다.  -food\_times 의 원소는 1 이상 1,000 이하의 자연수이다.  -k는 1 이상 2,000,000 이하의 자연수이다.  효율성 테스트 제한 사항  -food\_times 의 길이는 1 이상 200,000 이하이다.  -food\_times 의 원소는 1 이상 100,000,000 이하의 자연수이다.  -k는 1 이상 2 x 10^13 이하의 자연수이다. |
| <입력>  - 첫째 줄에 k값을 입력한다.  - 둘째 줄에 food\_times을 입력합니다. |
| <출력>  - 첫째 줄에 섭취해야하는 음식의 번호를 입력하세요 |
| <예제>   |  |  | | --- | --- | | 입력 1  5  3 12 | 출력 1  1 | |