

연속행렬곱셈-Memoization

여러 개의 행렬을 곱할 때 각 행렬 원소의 곱셈의 총 횟수를 최소한으로 하는 행렬의 곱셈을 계산하는 프로그램을 작성하시오. 단, 원소의 곱셈의 최소 횟수만 계산하면 된다. 즉, 강의 노트에서 제시한 원소 곱셈의 최소 횟수를 정의하는 점화식을 Memoization 기법을 이용한 동적계획법 프로그램으로 구현하시오.

예를 들어, 4개의 행렬 M_1, M_2, M_3, M_4 가 주어졌고, 각 행렬의 크기가 각각 $20 \times 2, 2 \times 30, 30 \times 12, 12 \times 8$ 이라고 하자.

$$\begin{array}{ccccccc} M_1 & \times & M_2 & \times & M_3 & \times & M_4 \\ 20 \times 2 & & 2 \times 30 & & 30 \times 12 & & 12 \times 8 \end{array}$$

위와 같이 이 네 개의 행렬의 곱 ($M_1 M_2 M_3 M_4$) 을 계산할 때, 이 행렬을 곱하는 여러 개의 수식에 대하여, 그 수식에서 곱하게되는 행렬 원소의 곱셈의 총 횟수는 다음과 같다.

$$\begin{array}{ll} (((M_1 M_2) M_3) M_4) & : 20 \times 2 \times 30 + 20 \times 30 \times 12 + 20 \times 12 \times 8 = 10,320 \\ ((M_1 (M_2 M_3)) M_4) & : 2 \times 30 \times 12 + 20 \times 2 \times 12 + 20 \times 12 \times 8 = 3,120 \\ ((M_1 M_2) (M_3 M_4)) & : 20 \times 2 \times 30 + 30 \times 12 \times 8 + 20 \times 30 \times 8 = 8,880 \\ (M_1 ((M_2 M_3) M_4)) & : 2 \times 30 \times 12 + 2 \times 12 \times 8 + 20 \times 2 \times 8 = 1,232 \\ ((M_1 (M_2 (M_3 M_4))) & : 30 \times 12 \times 8 + 2 \times 30 \times 8 + 20 \times 2 \times 8 = 3,680 \end{array}$$

따라서 위 4 개의 행렬을 곱할 때, 각 행렬원소의 곱셈의 총 횟수의 최소 횟수는 1,232이다.

<< 주의 >>

다음과 같은 경우 혹은 그 외의 편법을 통한 프로그램을 제출하는 경우에는 이 문제의 점수뿐만 아니라 다른 모든 과제의 점수를 0 점으로 처리함.

1. 위 해결 프로그램을 Bottom-Up 방법의 동적계획법 등의 다른 방법으로 구현하여 제출하는 경우.
2. 채점 프로그램 시스템에 Bottom-Up 방법의 동적계획법 등의 다른 방법으로 구현한 프로그램을 한 번이라도 제출하는 경우.

입력

입력은 표준입력(standard input)을 사용한다. 입력은 t 개의 테스트 케이스로 주어진다. 입력 파일의 첫 번째 줄에 테스트 케이스의 개수를 나타내는 정수 t 가 주어진다. 두 번째 줄부터 두 줄에 한 개씩의 테스트 케이스가 입력된다. 각 테스트 케이스에 해당되는 첫 번째 줄에는 곱해야 할 행렬의 개수를 나타내는 정수 n ($1 < n \leq 100$)이 먼저 주어지고, 그 다음 줄에는 $n + 1$ 개의 정수 d_0, d_1, \dots, d_n ($1 \leq$

$d_i \leq 100, i = 0, \dots, n$ 이 주어진다. 이 정수들은 k ($1 \leq k \leq n$) 번째 행렬 M_k 의 크기 $d_{k-1} \times d_k$ 를 나타낸다. 이 정수들 사이에는 한 개의 공백이 있으며, 잘못된 데이터가 입력되는 경우는 없다.

출력

출력은 표준출력(standard output)을 사용한다. 입력되는 테스트 케이스의 순서대로 다음 줄에 이어서 각 테스트 케이스의 결과를 출력한다. 각 테스트 케이스에 해당하는 출력의 첫 줄에 입력되는 행렬을 곱할 때, 행렬 원소의 곱셈의 최소횟수를 나타내는 정수를 출력한다.

입력과 출력의 예

입력	출력
3	1232
4	24
20 2 30 12 8	2814
2	
2 3 4	
25	
4 5 4 4 3 8 10 7 9 8 3 10 11 4 9 11 7 6 5 3 4 5 5 6 3 3	