

DSPC 2024

by

Official Solutions



출제

- ✓ iansong
- ✓ kes0716
- ✓ byunsanghoon1025
- ✓ htg1616



검수

- ✓ h0ngjun7
- ✓ lky7674
- ✓ tony9402
- ✓ utilforever
- ✓ kwoncycle
- ✓ vklddjvkdnj



문제	의도한 난이도	출제자
A 단원평가	Easy	iansong
B 평점 변환	Medium	kes0716
C Best Chance	Medium	byunsanghoon1025, iansong
D 증가와 감소	Medium	htg1616
E 주기 함수 (Easy)	Medium	iansong
F 교실 배정	Medium	htg1616
G chance!	Medium	byunsanghoon1025
H 문자열 수정	Hard	iansong
I 구조대	Hard	iansong
J 주기함수(Hard)	Very Hard	iansong
Ex 가우스 법칙	Challenging	iansong



A. 단원평가

implementation, arithmetic

출제진 의도 – **Easy**

- ✓ 제출 269번, 정답 198명 (정답률 74.349%)
- ✓ 처음 푼 사람: **dkim110807, 0분**
- ✓ 출제자: **iansong**



- ✓ 입력을 if 문을 사용하여 0이 어디에 있느냐에 따라 다르게 처리해 주면 됩니다.
- ✓ 마지막에 0이 들어오는 경우 $\sqrt{a + b}$ 를, 그렇지 않은 경우 마지막 수의 제곱에서 a, b 중 0이 아닌 수를 빼 주면 됩니다.



B. 평점 변환

implementation

출제진 의도 – **Easy**

- ✓ 제출 154번, 정답 82명 (정답률 63.636%)
- ✓ 처음 푼 사람: **dabbler1, 5분**
- ✓ 출제자: kes0716



- ✓ 주어진 문자열을 순서대로 탐색하면서, 맨 끝에 다다르거나 알파벳이 나올 때마다 하나의 평점을 지나간 것입니다.
- ✓ 하나의 평점을 지나간 후에는 그 평점을 읽고 해당 학기의 바뀐 평점을 출력한 후에, 평점을 저장해 두어야 나중에 이전 평점으로 사용할 수 있습니다.
- ✓ 'A', 'B', 'C'에 각각 2, 5, 8의 수를 부여하고, '+' '-'에 각각 1과 -1을 부여하면 평어가 1부터 9 까지의 수로 바뀌므로 구현이 간단해집니다.
- ✓ 문자열의 길이가 최대 $2N$ 임에 유의해야 합니다.



C. Best Chance

implement

출제진 의도 – **Easy**

- ✓ 제출 173번, 정답 85명 (정답률 58.382%)
- ✓ 처음 푼 사람: **dkim110807, 4분**
- ✓ 출제자: byunsanghoon1025, iansong



- ✓ 기회비용을 구하기 위해 모든 물건에 대해 자신을 제외한 물건의 이익을 구하면 시간복잡도가 $\mathcal{O}(N^2)$ 이기 때문에 시간초과가 발생합니다.
- ✓ 가장 큰 물건의 이익 값과 두번째로 물건의 이익 값만 구하면 기회비용을 구할 수 있습니다.
- ✓ 기회비용을 구하면 물건의 이익에서 기회비용과 가격을 빼서 순수익을 구할 수 있습니다
- ✓ 가장 큰 물건의 이익, 두번째로 큰 물건의 이익, 기회비용, 순수익을 구하는 것은 모두 $\mathcal{O}(N)$ 의 시간복잡도로 구할 수 있습니다.



D. 증가와 감소

bruteforcing, prefix_sum, precomputation

출제진 의도 - **Medium**

- ✓ 제출 118번, 정답 43명 (정답률 36.441%)
- ✓ 처음 푼 사람: **dabbler1, 13분**
- ✓ 출제자: htg1616



- ✓ 어떤 양의 정수가 증가 후 감소하는 수인지는 $\mathcal{O}(1)$ 의 시간복잡도로 판별 가능합니다.
- ✓ i 이하의 양의 정수 중 증가 후 감소하는 수의 개수 = $P[i]$ 인 누적합 배열 P 를 생성합니다.
- ✓ 누적합 배열을 생성한 이후에, 테스트 케이스들을 입력받으며 $P[b] - P[a - 1]$ 를 출력하면 시간 제한 안에 해결할 수 있습니다.



E. 주기 함수 (Easy)

math, calculus

출제진 의도 – **Medium**

- ✓ 제출 105번, 정답 42명 (정답률 40.000%)
- ✓ 처음 푼 사람: **nflight11, 4분**
- ✓ 출제자: **iansomg**

E. 주기 함수 (Easy)



- ✓ 주어진 구간 $[a, b]$ 안에 주기가 얼마나 들어 있는지를 셀 수 있습니다.
- ✓ $\frac{b-a}{p}$ 는 구하고자 하는 구간 안에 완전히 들어있는 주기의 개수를 의미합니다.
- ✓ 이 값에 한 주기의 적분값을 곱하여 불필요한 연산을 크게 줄일 수 있습니다.
- ✓ 나머지 부분은 누적합을 사용하거나 이정도는 단순히 더해 줘도 시간은 충분합니다.
- ✓ 앞서 주기를 전처리한 것에 의해 나머지 부분이 두 구간으로 분리될 수 있음을 유의하세요.



F. 교실 배정

math, combinatorics

출제진 의도 – **Medium**

- ✓ 제출 94번, 정답 40명 (정답률 42.553%)
- ✓ 처음 푼 사람: **p_a_r_k_j_u_n_e, 18분**
- ✓ 출제자: htg1616



- ✓ 교실의 번호를 $1, 2, 3, \dots, N$ 으로 둡시다.
- ✓ N 이 짝수라면 1번 교실과 짹지를 수 있는 교실은 $N - 1$ 개입니다.
- ✓ 1번 교실을 다른 교실과 짹짓고 나면 $N - 2$ 개의 교실이 남게 됩니다. 이는 귀납적으로 $N - 2$ 개의 교실을 짹짓는 상황으로 생각할 수 있습니다.
- ✓ 따라서 N 이 짝수일 때 교실을 짹짓는 방법의 수는 $1 \times 3 \times \dots \times 5 \times (N - 1) = (N - 1)!!$ 이 됩니다.
- ✓ N 이 홀수라면 빈 교실이 될 교실 하나를 선택한 후, $N - 1$ 개의 교실을 짹지으면 됩니다.
- ✓ 따라서 N 이 홀수일 때, 교실을 짹짓는 방법은 $N!!$ 이 됩니다.



G. Chance!

bfs, dp

출제진 의도 – **Medium**

- ✓ 제출 69번, 정답 37명 (정답률 53.623%)
- ✓ 처음 푼 사람: **p_a_r_k_j_u_n_e, 23분**
- ✓ 출제자: **byunsanghoon1025**



- ✓ $\times 10$ 을 안 쓰고도 b 에 빨리 도착할 수 있기 때문에 최소 마법 사용 횟수를 구하기 위해서는 bfs로 탐색해야 합니다.
- ✓ bfs로 a 에서 b 까지 몇 번 마법을 사용해야 하는지 탐색합니다.
- ✓ a 에서 어떤 정수까지 도달하는데 필요한 마법 사용 횟수를 저장하는 배열을 $2(b + 1)$ 크기의 배열로 만듭니다.
- ✓ chance! 마법은 1 번만 사용할 수 있기 때문에 chance! 마법 사용 전과 사용 후를 따로 나눠서 마법 사용 횟수를 저장합니다.



H. 문자열 수정

constructive

출제진 의도 – **Hard**

- ✓ 제출 56번, 정답 15명 (정답률 27.778%)
- ✓ 처음 푼 사람: **pretty_pi, 37분**
- ✓ 출제자: **iansong**



- ✓ 만약 어떤 문자의 개수가 길이의 절반을 넘어선다면 비둘기집 원리에 의해 어떤 방식으로든 완전히 새로운 문자열을 만들 수 없습니다.
- ✓ 이외의 경우에는 서로 다른 문자열을 하나씩 짹지어서 바꿔줄 수 있으므로 $\left\lfloor \frac{N+1}{2} \right\rfloor$ 번 교환해서 완전히 새로운 문자열을 만들 수 있습니다.
- ✓ 교환하는 방법은 예시를 통해 생각해봅시다.
 - 주어진 문자열이 aaabbbcccc라고 생각해봅시다.
 - 이때 a와 b를 두 번 이상 교환하면 c를 교환할 상대가 없어 실패합니다.
 - 따라서 현재 가장 많이 남아있는 글자끼리 교환해주는 전략이 필요합니다.
- ✓ 힙이나 `std::set`, `deque` 등의 자료구조로 편하게 구현할 수 있습니다.



I. 구조대

greedy, sweeping

출제진 의도 – **Hard**

- ✓ 제출 34번, 정답 11명 (정답률 32.353%)
- ✓ 처음 푼 사람: **dabbler1, 55분**
- ✓ 출제자: **iansong**



- ✓ 어떤 팀이 두개의 구간 $[l_1, r_1], [l_2, r_2]$ 을 가진다고 가정해 봅시다. ($r_1 \leq l_2$)
- ✓ 두 구간과 모두 겹치는 선분을 찾는 문제는 $[r_1 - 1, l_2]$ 를 포함하는 선분을 찾는 문제와 동치입니다.
- ✓ 하나의 팀에 대해 인접한 구간을 이와같이 변환해주면 원래 문제는 선분을 잘 이동해서 선분에 포함되는 구간에 최대한 많은 팀이 있도록 하면 됩니다.
- ✓ 동일한 팀의 변환된 선분 여러 개가 선분 안에 들어갈 수 있으니 이를 중복하여 세지 않도록 주의하며 sweeping으로 해결할 수 있습니다.



J. 주기함수(Hard)

math, KMP, calculus

출제진 의도 – **Very Hard**

- ✓ 제출 52번, 정답 16명 (정답률 30.769%)
- ✓ 처음 푼 사람: **nflight11, 11분**
- ✓ 출제자: **iansong**



- ✓ 함수의 주기가 입력 구간의 절반 이하임이 보장되므로 적어도 두 번의 주기가 입력 안에 나타납니다.
- ✓ 따라서 입력 배열에서 반복되는 패턴 중 아무 길이를 가진 패턴을 하나 잡아오면 됩니다.
- ✓ 주의해야할 점은 한 주기의 적분값이 1 2 3 1 2 3 4 5와 같이 내부에도 반복되는 패턴이 있을 수 있습니다.
- ✓ 따라서 $\frac{L}{2}$ 부터 시작하여 반복되는 가장 긴 패턴을 찾는 것이 안전합니다.
- ✓ 패턴 찾기는 KMP나 접미사 배열 등의 문자열 알고리즘을 적용할 수 있습니다.
- ✓ 주기를 구했다면 Easy와 동일하게 구현하면 됩니다.



K. 가우스법칙

geometry, convex_hull, point_in_convex_polygon, binary_search, sweeping

출제진 의도 – **Challenging**

- ✓ 제출 38번, 정답 3명 (정답률 7.895%)
- ✓ 처음 푼 사람: **dkim110807, 153분**
- ✓ 출제자: **iansong**

K. 가우스법칙



- ✓ 점을 제거하는 과정에서는 가우스 면 내부에 있는 점전하만 확인해주면 됩니다.
- ✓ 이분 탐색을 활용하여 점전하가 가우스 면 내부에 있는지 $\mathcal{O}(\log N)$ 으로 구할 수 있습니다.
- ✓ 점을 하나를 제거하면 제거한 점과 양 옆의 점을 꼭짓점으로 하는 삼각형 모양의 영역이 변동됩니다.
- ✓ 볼록 껍질 내부의 점전하는 최대 두개의 삼각형 영역에 들어갈 수 있습니다.
- ✓ $Q \leq 10^5, N \leq 10^5$ 이므로 점전하가 어떤 삼각형 영역에 들어갈 지 $\mathcal{O}(N)$ 이하로 구해야 합니다.
- ✓ 기준점을 잡고 각도 정렬한 후 이분 탐색을 활용하여 $\mathcal{O}(\log N)$ 으로 구할 수 있습니다.
- ✓ 새로 생긴 선분을 내부로 포함해야 할지 잘 선택해야 합니다.



- ✓ 하지만 N 이 작을때 예외 처리를 해주어야 합니다.
- ✓ 예를 들어 $N = 4$ 인 경우 사변형의 두 대각선의 교점에 있다면 어떤 점을 빼더라도 이 점전하는 새로운 가우스 면 내부에 포함되는 일이 없습니다.
- ✓ 이러한 예외는 N 이 작을 때 생기므로 완전 탐색으로 구현해주면 됩니다.
- ✓ 이렇게 각 영역이 포함하는 전하량의 총합을 구할 수 있습니다.
- ✓ 절댓값을 가장 크게 해야 하므로 초기 가우스 면 내부 전하량의 총합의 부호에 따라 어떤 영역을 빼야할 지 결정할 수 있습니다.