

112 彰中校內學科能力競賽遴選賽題解

Gamic

2023/8/29

目錄

1 守護城牆

2 E 班的復仇

3 文字獄

目錄

1 守護城牆

2 E 班的復仇

3 文字獄

題目

給定 N 個巨人的身高 h_i 和距離 d_i

巨人每單位時間移動 1 的距離

擊殺一隻巨人需要 h_i 的時間

請問是否有種方法在巨人抵達前全部擊殺

守護城牆 sub1&sub2

- 每次都擊殺最近的巨人會是最好的方案嗎？

守護城牆 sub1&sub2

- 每次都擊殺最近的巨人會是最好的方案嗎？
- 試試看！

守護城牆 sub1&sub2

- 每次都擊殺最近的巨人會是最好的方案嗎？
- 試試看！
- 把 d_i 和 h_i 用 pair 合併起來，對 d_i 由小到大排序

守護城牆 sub1&sub2

- 每次都擊殺最近的巨人會是最好的方案嗎？
- 試試看！
- 把 d_i 和 h_i 用 pair 合併起來，對 d_i 由小到大排序
- 如果對於所有 $d_i \leq i$ 那就是 Yes(i 是從 1 開始)

守護城牆 sub1&sub2

- 每次都擊殺最近的巨人會是最好的方案嗎？
- 試試看！
- 把 d_i 和 h_i 用 pair 合併起來，對 d_i 由小到大排序
- 如果對於所有 $d_i \leq i$ 那就是 Yes(i 是從 1 開始)
- 時間複雜度 $O(N \log N)$

守護城牆

- 用一個變數紀錄到目前為止花了多少時間，若有某個時刻花的時間大於 d_i 就輸出 No
- 時間複雜度 $O(N \log N)$

目錄

1 守護城牆

2 E 班的復仇

3 文字獄

E 班的復仇

題目

給定三個 $1 \sim 9$ 的數字

數字之間可以合併

加上 $+ - *$ 運算子

製造出的式子可以有多少種不一樣的答案

E 班的復仇

- 可以分成合併後兩個數字和沒有合併三個數字兩種情況討論

E 班的復仇

- 可以分成合併後兩個數字和沒有合併三個數字兩種情況討論
- 遍歷所有 abc, acb, bac... 的排列

E 班的復仇

- 可以分成合併後兩個數字和沒有合併三個數字兩種情況討論
- 遍歷所有 abc, acb, bac... 的排列
- 在中間位置劃刀分成兩個數字，分別算 + - *

E 班的復仇

- 可以分成合併後兩個數字和沒有合併三個數字兩種情況討論
- 遍歷所有 $abc, acb, bac\dots$ 的排列
- 在中間位置劃刀分成兩個數字，分別算 $+ - *$
- 不用劃刀了，在三個數字中間枚舉 $(+, +), (+, -)\dots$ 九種情況

E 班的復仇

- 可以分成合併後兩個數字和沒有合併三個數字兩種情況討論
- 遍歷所有 $abc, acb, bac\dots$ 的排列
- 在中間位置劃刀分成兩個數字，分別算 $+ - *$
- 不用劃刀了，在三個數字中間枚舉 $(+, +), (+, -)\dots$ 九種情況
- 注意重複的答案不能重複算到

目錄

1 守護城牆

2 E 班的復仇

3 文字獄

題目

定義 A 陣列由所有不包含 7 的正整數由小到大排列所形成的陣列

$A = [1, 2, 3, 4, 5, 6, 8, 9, 10, 11, 12, 13, 14, 15, 16, 18, 19\dots]$

詢問第 N 項是多少 (1-base) ?

文字獄 sub1sub2

- 寫一個函式判斷輸入的數字有沒有包含 7

```
int main(){
    int q;
    cin >> q;
    while(q--){
        ll n, cnt = 1;
        cin >> n;
        for(int i = 1; i <= n; ++i, ++cnt){
            while(!ok(cnt)) ++cnt;
        }
        cout << cnt - 1 << endl;
    }
}
```

文字獄

- 如果把 A 改定義成由所有只含 1 或 0 的正整數由小到大排列所形成的陣列

文字獄

- 如果把 A 改定義成由所有只含 1 或 0 的正整數由小到大排列所形成的陣列
- 那麼第 N 項不就變成了二進位表示法下的 N 了嗎

文字獄

- 如果把 A 改定義成由所有只含 1 或 0 的正整數由小到大排列所形成的陣列
- 那麼第 i 項不就變成了二進位表示法下的 N 了嗎

$1 \rightarrow 1$
 $2 \rightarrow 10$
 $3 \rightarrow 11$
 $4 \rightarrow 100$

文字獄

- 因此把 N 轉換為九進制的表達法

文字獄

- 因此把 N 轉換為九進制的表達法
- 如果某個位數大於等於 7 的話把那個位數 +1

文字獄

- 因此把 N 轉換為九進制的表達法
- 如果某個位數大於等於 7 的話把那個位數 +1
- 時間複雜度 $O(T \log_9 N)$