

ICPC 模擬国内予選 2016

D : ゲームバランス

原案 : 井上

問題文 : 飯塚

解答 : 井上・栗田・水野・飯塚・田中・矢野

解説 : 飯塚

問題概要 (1/2)

- ゲームの戦闘とレベルアップに関係するパラメータ x を決めてゲームバランスを調整したい
- 敵モンスターは N 種類いる。 i 種類目の強さは s_i
- 主人公キャラクターが一番強い敵モンスターを倒すとゲームクリア
- ゲームクリアまでに必要な戦闘の回数を M 回以上にするような最大の x を求めよ

問題概要 (2/2)

- 戦闘の仕様
 - 主人公のレベルが L のとき, 倒せるのは $L+X$ 未満の敵モンスターだけ
 - 同じ種類のモンスターと何度でも戦える
- レベルアップの仕様
 - 主人公の最初のレベルは 1
 - レベルが L のとき強さ s_j の敵を倒すと主人公のレベルが $L + \max(1, X - |L - s_j|)$ になる

サンプル

- $N = 3, M = 4, S = [1, 5, 9]$
 - $X = 3$ の場合
 - 最初の状態では $L = 1$
 - s_1 を倒すと $L = 1 + \max(1, 3 - |1 - 1|) = 4$
 - s_2 を倒すと $L = 4 + \max(1, 3 - |5 - 4|) = 6$
 - s_3 はまだ倒せないなので s_2 を倒す。
 $L = 6 + \max(1, 3 - |5 - 6|) = 8$
 - s_3 が倒せるようになったので倒す
 - 4回の戦闘でゲームクリア
 - これは M の制約を満たす

サンプル

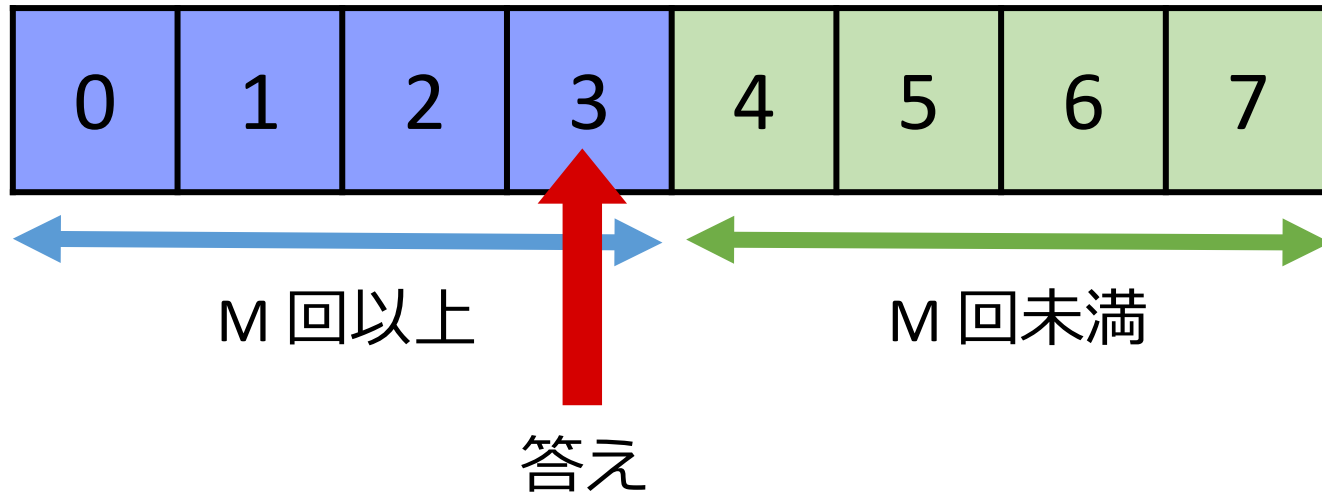
- $N = 3, M = 4, S = [1, 5, 9]$
 - $X = 4$ の場合
 - 最初の状態では $L = 1$
 - s_1 を倒すと $L = 1 + \max(1, 4 - |1 - 1|) = 5$
 - s_2 を倒すと $L = 5 + \max(1, 4 - |5 - 4|) = 8$
 - s_3 が倒せるようになったので倒す
 - 3回の戦闘でゲームクリア
 - これは M の制約を満たさない
 - よって M の制約を満たす X の最大値は 3

特殊な場合

- $N = 1, M = 2, S = [1]$
- $X = 1$ の場合
 - 最初の戦闘でゲームクリアできてしまうので,
M の制約を満たせない
- $X = 0$ の場合
 - ゲームをクリアできない
- このような場合は -1 を答える

想定解法

ゲームクリアに必要な戦闘回数が
M 回以上となる x の範囲を二分探索で求める



二分探索を使うために確認が必要なこと

1. 戦闘回数は x の増加に対して単調減少か？
2. x を決まっているときクリアまでに必要な最小の戦闘回数はいかに計算すればよいか？ ← 先にこっちを説明

X が決まっているときの最小の戦闘回数

問題：X の値が決められていたとして、ゲームクリアまでに必要な戦闘回数を最小化する戦略は？

答え：貪欲に、主人公のレベルが一番上がる敵モンスターを倒し続ける

証明：一番強い敵モンスターを倒せるレベルになるための最適な戦略であることを示せば良い。

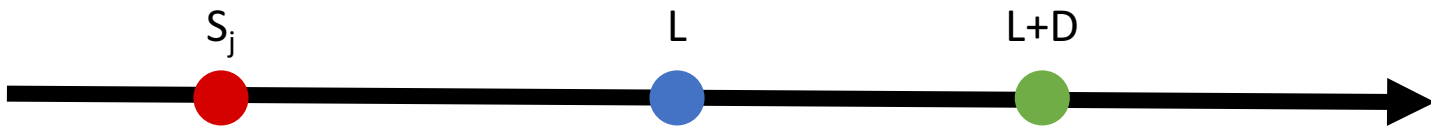
主人公のレベルを L にする選択肢と $L+D$ ($D>0$) にする選択肢があるとき、 $L+D$ の選択肢が常に良いことを示す（次ページに続く）

Xが決まっているときの最小の戦闘回数

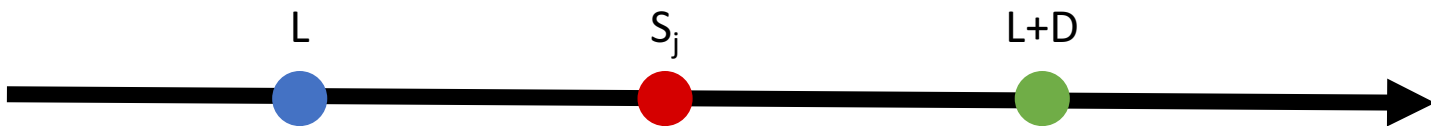
(証明つづき)

次に倒そうとしている敵モンスターの強さを s_j とする

i. $s_j \leq L$ のとき



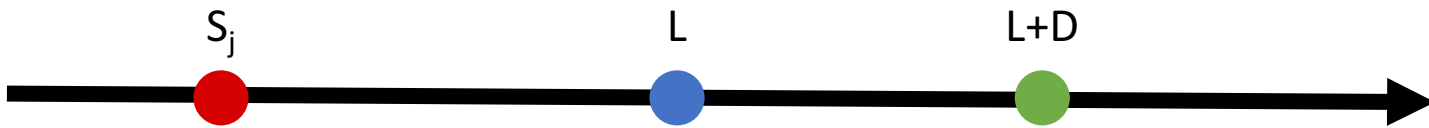
ii. $L < s_j < L+D$ のとき



iii. $L+D \leq s_j$ のとき

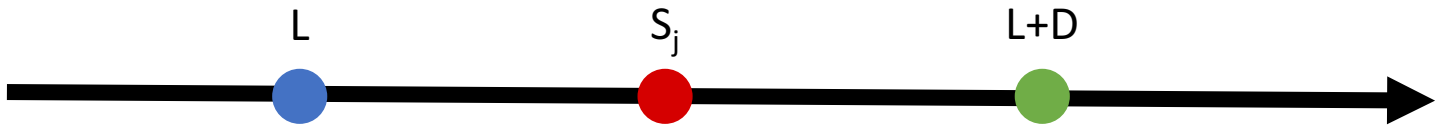


i. $S_j \leq L$ のとき



- レベル L で倒すと $L + \max(1, X + S_j - L)$ になる
- レベル $L+D$ で倒すと $L + D + \max(1, X + S_j - L - D)$ になる
- a. $X + S_j - L \leq 1$ のとき
レベル L で倒すと $L+1$ に, レベル $L+D$ で倒すと $L+D+1$ に
➡ **$L+D$ の勝ち!**
- b. $X + S_j - L > 1$ かつ $X + S_j - L - D \leq 1$ のとき
レベル L で倒すと $X + S_j$ に, レベル $L+D$ で倒すと $L+D+1$ に
 $X + S_j - L - D \leq 1$ より $X + S_j \leq L + D + 1$
➡ **$L+D$ の勝ち!** or **引き分け**
- c. $X + S_j - L - D > 1$ のとき
レベル L で倒すと $X + S_j$ に, レベル $L+D$ で倒すと $X + S_j$ に
➡ **引き分け**

ii. $L < S_j < L+D$ のとき



- レベル L で倒すと $L + \max(1, X - S_j + L)$ になる
- レベル $L+D$ で倒すと $L + D + \max(1, X + S_j - L - D)$ になる
- a. $X - S_j + L \leq 1$ かつ $X + S_j - L - D \leq 1$ のとき
レベル L で倒すと $L+1$ に, レベル $L+D$ で倒すと $L+D+1$ に
➡ **$L+D$ の勝ち!**
- b. $X - S_j + L \leq 1$ かつ $X + S_j - L - D > 1$ のとき
レベル L で倒すと $L+1$ に, レベル $L+D$ で倒すと $X + S_j > L+D+1$
➡ **$L+D$ の勝ち!**
- c. $X - S_j + L > 1$ かつ $X + S_j - L - D \leq 1$ のとき
レベル L で倒すと $X + 2L - S_j$ に, レベル $L+D$ で倒すと $L+D+1$ に
 $(X + 2L - S_j) - (L - D - 1) = X + L - S_j - D - 1 < X + S_j - L - D - 1 \leq 1$
➡ **$L+D$ の勝ち!** or **引き分け**
- d. $X - S_j + L > 1$ かつ $X + S_j - L - D > 1$ のとき
レベル L で倒すと $X + 2L - S_j$ に, レベル $L+D$ で倒すと $X + S_j$ に
 $(X + 2L - S_j) - (X + S_j) = 2(L - S_j) < 0$ ➡ **$L+D$ の勝ち!**

iii. $L+D \leq s_j$ のとき



- Lよりも $L+D$ のほうが s_j に近いのでレベルアップ後のレベルが高い → **$L+D$ の勝ち!**

i, ii, iii まとめ

- i, ii, iii より「レベルLよりもレベルL+Dのほうが損になる」ということはない,ということが分かった
 - 最小の戦闘回数でゲームクリアするためにはレベルが一番上がる敵モンスターを倒し続けるのが最適
- ➡ パラメータ x が決まっているときの
ゲームクリアまでの最小の戦闘回数が分かった

戦闘回数は X の増加に対して単調減少か？

- 先ほどの議論より, レベルが高くて損することはない
- X が増加すると, レベルアップ後のレベルが高くなる
- よって X が大きいほうが戦闘回数が少なくなる

- 二分探索で動かす x の範囲は $[1, S_N] \dots O(\log S_N)$
- x を決めて戦闘が M 回以上か判定する処理 $\dots O(S_N)$
 - 最もレベルが上がる敵モンスターの強さは、主人公のレベルが上がるごとに広義単調増加なのでポイントを動かす方法によって合計で $O(N)$
 - 1回の戦闘でレベルが少なくとも1は上がるので、同じモンスターを何度も倒すのは高々 $O(S_N)$ 回
 - S は distinct なので結局 $O(S_N)$
- よって合計で $O(S_N \log S_N)$

注意したほうが良いケース

- ゲームクリアできない場合
 - $s_1 \geq X+1$ の場合はゲームクリアできない
 - 倒せる敵ならば、最低でもレベルが1上がるので s_1 が倒せるかどうかのみ考えればよい
- 全ての種類の敵を倒す必要はない場合
 - s_N を倒した瞬間にゲームクリアなので、全種類の敵を倒す必要はない
 - s_j を倒したあとに s_{j+1} を倒すよりも s_{j+2} を倒したほうが良いような場合もある
 - 前ページの「ポイントを動かす方法」の実装にバグが入りやすい

ジャッジ解

- 井上 (C++): 104行, 2682 bytes
- 栗田 (C++): 48行, 1087 bytes
- 水野 (C++): 55行, 1273 bytes
- 飯塚 (C++): 63行, 1452 bytes
- 田中 (Java): 53行, 1150 bytes
- 矢野 (C++): 61行, 1508 bytes

- ACチーム数
 - 41チーム
- 最初に正解したチーム
 - moyat (0:54:34)