コ　　ア　　ラ

解説者:笠浦一海
（ガチクズ勢）

問題

- コアラが座標Kの点から座標Mの点までジャンプして進む。
- 一回のジャンプで距離Dまで移動できる。

問題

- 一回のジャンプで体力をA消費。
- 座標T_iの点に行くと体力がB_i回復。

問題

- 体力の初期値は0。
- 最終的な体力の最大値は？

考察

- 後戻りしない。
- 「座標Xにいるときの体力の最大値」をDPで順番に求めればいけそう。
- 愚直にやるとO(M*D)。

→明示的な部分点はない

考察

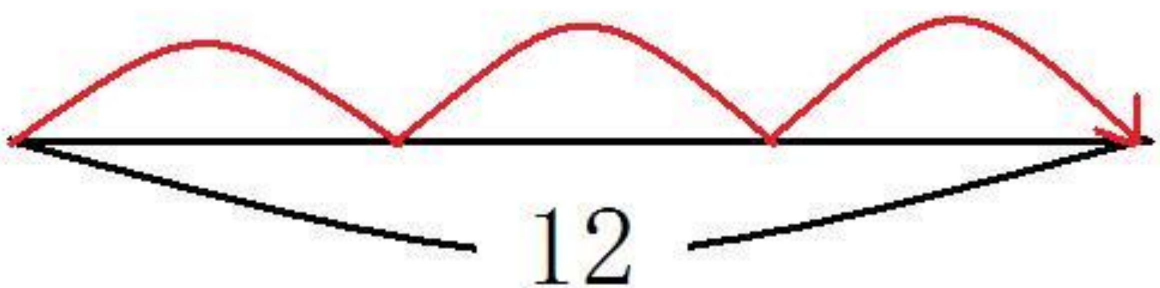
- 家があるところだけ考える。

動的計画法(20点解法)

- T_0=K,T_{N+1}=M,B_0=B_{N+1}=0とする。
- DP[i]=(T_iまで来たときの体力の最大値)。
- DP[i]をDP[0]～DP[i-1]までの値を使って求めればよい。

動的計画法(20点解法)

- T_kからT_iまで寄り道せずに移動するとき体力をceil((T_i-T_k)/D)*Aだけ消費。



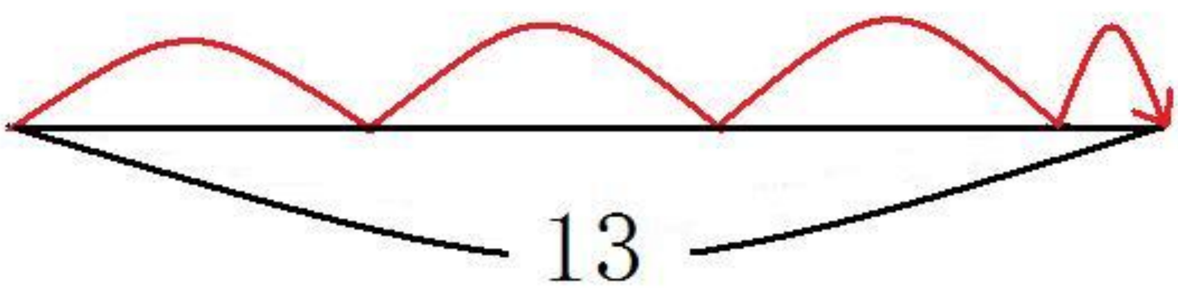
D=4

T_i-T_k=12

ジャンプ3回

動的計画法(20点解法)

- T_kからT_iまで寄り道せずに移動するとき体力をceil((T_i-T_k)/D)*Aだけ消費。



D=4
T_i-T_k=13
ジャンプ4回

動的計画法(20点解法)

- $DP[i]=\max\{DP[k]-\mathrm{ceil}((T_i-T_k)/D)*A : k<i\}$
  $+B[i]$
- O(N^2)→20点

さらに考察

- それまでの値を全部調べなくても DP[i]を求められるようにしたい。
- DPの式を眺めてみる。
- ceil((T_i-T_k)/D)が大事。

さらに考察

- D=1のとき。
- $DP[i]=\max\{DP[k]+T_k*A-T_i*A:k<i\}+B[i]$
- kに関する式とiに関する式に分離できる。
- $DP[k]+T_k*A$の最大値を記録しておけばオッケー

さらに考察

- Dが一般の値の時も同じようにできないだろうか？

さらに考察

- ceil((T_i-T_k)/D)は……。

さらに考察

- $T_i\%D>T_k\%D$のとき

  $floor(T_i/D)-floor(T_k/D)+1$
- $T_i\%D<=T_k\%D$のとき

  $floor(T_i/D)-floor(T_k/D)$
- これからはceil()はつかわないのでfloor()は省略します。

代入して

- $DP[i]=max($
  $max\{DP[k]-(T_i/D-T_k/D+1)*A$
  $:k<i,T_i\%D<T_k\%D\},$
  $max\{DP[k]-(T_i/D-T_k\%D)*A$
  $:k<i,T_i\%D>=T_k\%D\})$
  $+B[i]$

整理

- $P[i]=DP[i]+(T_i/D)*A$とおくと

$P[i]=\max(\max\{P[k]-A:k<i,T_i\%D<T_k\%D\},$

$\max\{P[k]:k<i,T_i\%D>=T_k\%D\})+B[i]$

- きれいな式になった。
- T_iをDで割った余りだけが重要。

３０点解法

- D<=100
- T_i%Dの値ごとにP[i]の最大値を記録。

30点解法

- $S[x]=\max\{P[i]:T_i\%D==x\}$として
- $P[i]=\max(\max\{S[x]:0<=x<T_i\%D\}-A,$
  $\max\{S[x]:T_i\%D<=x<D\})+B[i]$
- $S[T_i\%D]=\max(S[T_i\%D],P[i])$
- $O(N*D)$→30点

満点解法

- 数列Sに対する操作。

①区間の最大値をとる。②値の更新。

- RMQ(Range Minimum Query)
- segment tree で解ける。
- Dが大きいので愚直に実装するとMLE

満点解法

- 座標圧縮。
- T_0%D,T_1%D,…,T_{N+1}%Dをソートして座標圧縮。
- ①も②もO(log N)でできる。
- 時間計算量O(N log N)

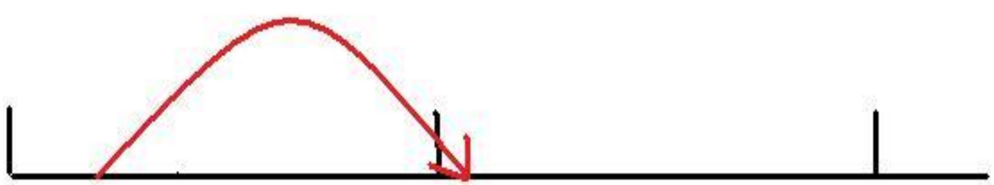
  空間計算量O(N) →満点！

数式を使わない考え方

- 道を距離Dごとに区切る。

数式を使わない考え方

- 一回のジャンプで次の区間に行けることもあれば、いけないこともある。



次の区間に行く例

次の区間に行かない例

数式を使わない考え方

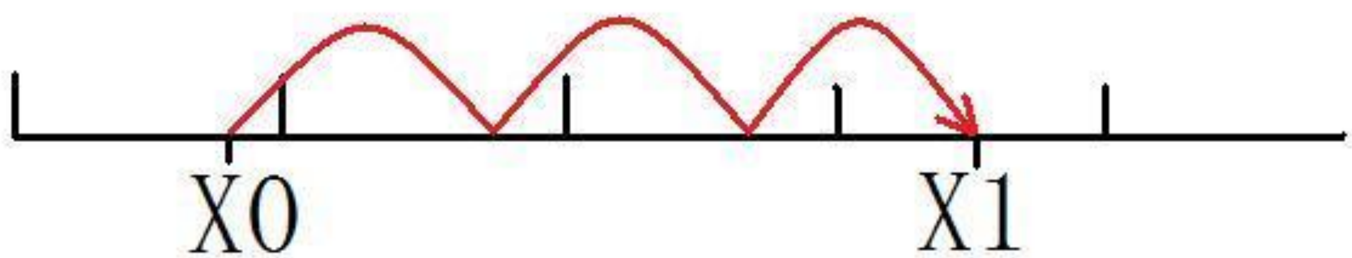
- 次の区間に行くジャンプを良いジャンプ、行けないジャンプを悪いジャンプを呼ぶ。
- ある地点まで行くときの良いジャンプの数は決まってる。
- 悪いジャンプの数だけ考えればよい。

数式を使わない考え方

- 座標X0の地点から座標X1の地点に行くとき……

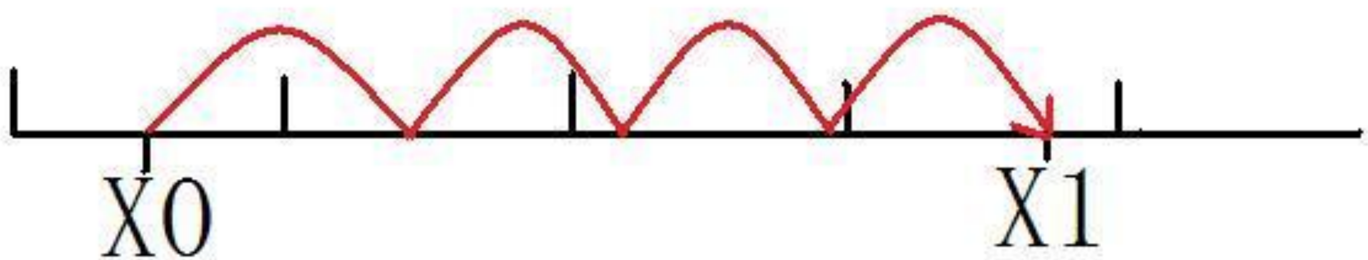
数式を使わない考え方

- X0%D>=X1%Dならば良いジャンプのみで行ける。



数式を使わない考え方

- X0%D<X1%Dのときは悪いジャンプが一回必要。



数式を使わない考え方

- 悪いジャンプによる疲労のみを考慮してDPを書き直すと、

数式を使わない考え方

- $P[i]=\max(\max\{P[k]-A:k<i,T_i\%D<T_k\%D\},$
  $\max\{P[k]:k<i,T_i\%D>=T_k\%D\})+B[i]$
- 同じ式が得られる。
- 良いジャンプによる疲労は最後に引けばよい。

得点分布

得点分布