



I: Tokyo Olympics Center

†JAG† 春コンテスト 2014

原案: 楠本
解答: 楠本, 汐田
解説: 楠本

問題概要

- 合宿後, みんなが泊まった後の部屋点検を任された!!
- 右の図のような地図が与えられる
 - A,B,C,...が宿泊棟の各ユニットの床, ‘.’が壁を表す.
 - 周り4マスのうち3マスが‘.’であるセルは部屋である.
 - そうでないセルは床である.
- 最初, セル(s,t)に K 人手伝ってくれる人(スタッフ)がいるので分担して全ての部屋をチェックしたい.
 - 部屋移動に T_{move} 時間, 部屋チェックに T_{check} 時間かかる.

```
.....  
.....AAABBBBBBB.....  
...A.AA.A...B.B..B.  
..AAAAAAAABBBBBBBB..  
...A..A.A.....B....  
.....A.....BBBB..  
...A.AA..C.C...B..  
...AAAAACCCCCBBBB..  
...A..A...C.C...B..  
.....
```



問題概要 (cont.)

- 以下のように分担する.
 - それぞれのユニットをスタッフのうちの誰かに割り振る.
 - 各スタッフはそれぞれ割り当てられたユニットをどう回るか決める.
 - 一度回る順番を決めたら、そのとおりに回らないといけない

A→E→C



D→F



B



問題概要 (cont.)

- 自分の分担されたユニットをチェックし終わったスタッフは元の位置 (s,t) に戻らなければならない.
- 全部屋をチェックし終わるのにかかる最短時間を計算せよ.
- 制約:
 - 地図の横幅, 縦幅 ≤ 50
 - ユニットの種類数 ≤ 12
 - 各ユニット内の部屋の個数 ≤ 12

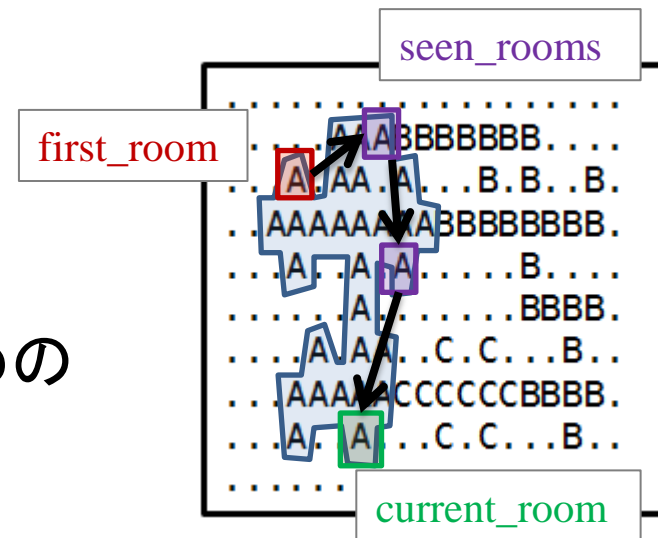


解法

- まず各セル同士の距離を前もって求めておく
 - 各セルを始点にしてBFSすればよい
 - 計算時間は $O(\text{部屋数}^2)$
- あと、部屋の位置をあらかじめ列挙しておく
- bit DPをいっぱいやる
 - 「ユニット内の部屋巡り」「ユニット間の移動」「スタッフへの割り振り」がそれぞれ層構造みたいになっていて、巡回セールスマン問題型のbit DPを各層ごとにやるイメージ

ユニット内の部屋巡り

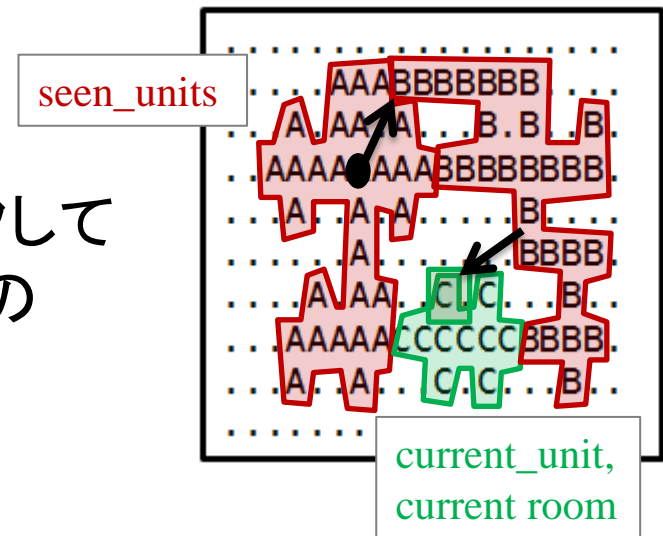
- $dp1[unit_id][seen_rooms][first_room][current_room]$
:= ユニット $unit_id$ 内の部屋を, $first_room$ を始点として $seen_rooms$ (部屋の集合) を見た上で $current_room$ に至る
ときの最短時間 をまず計算.
- よくある bit DP
- これを計算できると,
1つのユニットを全部調べるための
最短時間がわかる.



ユニット間の移動

- $dp2[seen_units][current_unit][current_room]$
:= スタート位置 (s,t) からスタートしてユニットの集合 $seen_units$ 内の全部屋をチェックし、ユニット $current_unit$ の部屋 $current_room$ に至るときの最短時間.
 - $dp1$ の計算結果を使って計算する.

- $dp3[seen_units]$
:= $seen_units$ 内の全部屋をチェックして元のスタート位置 (s,t) に戻るための最短時間
 - これは $dp2$ から即座に求まる.



スタッフへの割り振り

- $dp4[k][assigned_units]$
:= k 人のスタッフに $assigned_units$ (ユニットの集合) を割り振るときの最短時間
- 擬似コードは右みたいな感じになる
- foreachは右下の実装がオススメ

```
for k = 1→K
  for as = 0→2^#units
    rest = (2^#units-1) xor as
    foreach <<s:subset of rest>>
      dp4[k][as] = min(dp4[k][as],
                       max(dp4[i-1][as], dp3[s]))
```

```
for (int s=rest; ; s=(s-1)&s) {
  // <TODO>
  if (s == 0) break;
}
```

k, assigned_units

A→E→C

D→F



.....

bit DP の計算量は?

- 1つのユニットの部屋の個数の上界を R , ユニットの個数を U とする.
- 漸近的な計算量は以下のようになる.
 - dp1: $O(UR^3 2^R)$
 - dp2&dp3: $O(U^2 R^2 2^U)$
 - dp4: $O(K 3^U)$
- $U, R \leq 12$ だと全体的に少し大きいですが, 定数倍の枝刈りが効くしループ内の処理が軽いので実際には高速に動く.
(あとTLEも5秒あって安心?)

結果

- 提出数 : 5
- 解いた人 :
 - kawatea(194:01) (First AC)
 - wakaba (198:00)
 - uwi(235:58)



※<http://nyc.niye.go.jp/facilities/d2-6.html>

- ジャッジ解:
 - 楠本 : 4980bytes (手抜き実装:4554bytes)
 - 汐田 : 5460bytes
 - 保坂 : 5455bytes
 - 実装がそこそこ重いです

余談

- この問題は夏合宿2013で部屋点検を頑張って行った後、スタッフの地道な大変さを表すために合宿から帰宅した直後に作った.
- マナーを守って楽しく合宿!!!