free・コーディングテスト

グラフ, 散布図

自動的に生成された説明

いろは市には路線Aと路線Bの 2 つの路線が走っています。都心はA7であり、A1, A13, B1は住宅地のあるターミナル駅です。いろは市では、都心か住宅地のあるターミナル駅のみが、電車の始点か終点になりえます。

住宅地のあるターミナル駅には車庫があり、これらの駅には電車を夜間に何台でも停めておくことができます。しかし、都心には車庫がありません。そのため都心では、都心始発の電車を確保するために、前日のA線上り/下りB線下りのそれぞれの最終電車（合計 3 台のみ）を、プラットフォームに停めておく運用をしています。 このような運用を実現するために、下記のように電車が運行されています。

あなたは、いろは市の時刻表を作るプログラムを作成するように頼まれました。

**ザ・ミッション**

いろは市の路線Aと路線Bの、各駅の時刻表を出力するCLIアプリケーションを作成してください。

**共通**

* 以下のルールによって、電車の動きが一意に確定することが保証される
* 都心(A7)と住宅地のあるターミナル駅(A1, A13, B1)のみが、電車の始点・終点となる
* 住宅地のあるターミナル駅(A1, A13, B1)からは、23  23 時(23:00を含む)以降出発する電車はない
* 翌朝の一番電車の始発運転に利用するために、住宅地のあるターミナル駅(A1, A13, B1)から出発したそれぞれの最終電車は、必ずA7行である
* 路線の向きは以下のように定められる
* 同じ路線内ならば、路線のあとに書かれた数字が大きくなる方が「上り」である
* 同じ路線内ならば、路線のあとに書かれた数字が小さくなる方が「下り」である
* 全ての路線Bの駅にとって、A7は「上り」方向である
* 例: A4にいる人にとって、上りの電車とは、A4からA5へと向かう電車のことである

**路線A**

* A1発の電車
* 5:55発を始発として、5  5 分毎に出発する
* 1  1 本おきにA7行とA13行となる（最終電車はA7行）
* 始発はA7行である
* A7発の電車
* 6:06発を始発として、A1行の電車が 10  10 分毎に出発する
* 6:10に、A13行の電車が 1  1 本だけ出発する
* 一日を通じて、A1発A7行の電車の数だけ、電車が出発する
* A13発の電車
* 5:52発を始発として 10  10 分毎に出発する
* 最終電車以外の電車は、A1行である
* 最終電車は、A7行である

**路線B**

* B1発の電車
* 6:00発を始発として、6  6 分毎にA7行の電車が出発する
* A7発の電車
* 6:11発を始発として、6  6 分毎にB1行の電車が出発する
* 一日を通じて、B1発A7行の電車の数だけ、電車が出発する

**実装方法**

**CLIアプリケーションについて**

入力値を標準入力から受け取り、結果を標準出力に出力するCLIアプリケーションとして解答を実装してください。 詳細は末尾の「CLI アプリ作成用テンプレート」を参照ください。

**入力ルール**

以下のフォーマットの標準入力が与えられます。

R S DIR HH

制約は以下です。

1. R: 路線名
   * AもしくはB
2. S: 駅名
   * 路線名がAならば、A1, A2, ... A13のどれか
   * 路線名がBならば、A7, B1, B2, ..., B5のどれか
3. DIR: 方向
   * UもしくはD
   * Uは上り方向を表す
   * Dは下り方向を表す
4. HH: 時刻
   * 0  0 でパディングされた 2  2 文字の時間
   * 05, 06, ..., 22, 23のどれか

**出力ルール**

もし、 HH 時に運行される電車が

**1**

1 つもない場合は、一行で No train と標準出力に出力してください。

そうでない場合、1行でHH時の時刻表を、以下のフォーマットで標準出力に出力してください。

標準出力

HH: MM[ MM...]

* HH: MM[ MM...]のフォーマットで、出発時間の分を空白区切で出力する
* MMは昇順で出力する
* HH, 各MMは 0  0 でパディングする
* 最後のMMのあとにスペースを入れてはいけない（例えば、13: 03 23を13: 03 23と出力してはならない）