

マジックシティ (Magic City)

タシケントの市長は、有名な遊園地であるマジックシティを再設計したいと考えている。あなたには正の整数 K が与えられる。あなたの課題は、以下のように遊園地を設計することである：

- アトラクションの個数を表す整数 N を決める。ここで、アトラクションには 0 から $N - 1$ までの整数の番号が付けられる。
- 異なる 2 個のアトラクションの間を双方向に結ぶ歩道を追加する。ここで、同じアトラクションの組の間に、 2 個以上の歩道が存在しても良い。任意のアトラクションの組の間を、何本かの歩道を用いて移動可能であることは、要求されていない。
- $0 \leq i < N$ を満たす各 i について、アトラクション i にタイプ $T[i]$ を割り当てる。 0 から $2K - 1$ までの番号が付けられた $2K$ 種類のタイプがあり、それぞれのタイプは少なくとも 1 個のアトラクションに割り当てられている必要がある。異なるアトラクションに同じタイプが割り当てられていても良い。

市場調査では、次の事実が判明した：

1. それぞれの客は、同じタイプのアトラクションを連続して訪れることなく、 3 個のアトラクションを訪れたいと思っている。
2. 客は、多くの歩道の端点となっているアトラクションを好まない。

あり得るすべての客を満足させるため、市長はあなたの設計に 2 個の条件を課した。

ここで、 $t_1 \neq t_2$ かつ $t_2 \neq t_3$ であるとき、 $0 \leq t_1, t_2, t_3 < 2K$ を満たす順序のついたタイプの三つ組 (t_1, t_2, t_3) が興味深いと呼ぶことにする。 t_1 は t_3 と等しくても良いことに注意せよ。すなわち、 $2K \cdot (2K - 1)^2$ 個の興味深い三つ組が存在する。

条件 1： それぞれの興味深い三つ組 (t_1, t_2, t_3) について、次の条件を満たすアトラクション a_1, a_2, a_3 ($0 \leq a_1, a_2, a_3 < N$) が存在する：

- a_1, a_2, a_3 のタイプは t_1, t_2, t_3 である。すなわち、 $T[a_1] = t_1, T[a_2] = t_2, T[a_3] = t_3$ である。
- a_1 と a_2 を結ぶ歩道が存在する。
- a_2 と a_3 を結ぶ歩道が存在する。

ここで、 a_1 と a_3 を結ぶ歩道が存在するかどうかは無関係である。また、 $t_1 = t_3$ であるとき、アトラクション a_1 と a_3 は等しくても良いことに注意せよ。

条件 2： それぞれのアトラクションは、高々 K 本の歩道の端点である。

この問題は部分点の付いた 50 個の出力のみの小課題からなる。各小課題は具体的な K の値と対応しており、あなたはその K の値に対し、上記のすべての条件を満たす遊園地を設計しなければならない。

あなたの得点は、あなたの解のアトラクションの個数に依存する。アトラクションの数がより少なければ、より高い、または等しい得点を得ることができる。

実装の詳細

あなたの解を提出するには2つの方法がある。あなたはそれぞれの小課題でいずれかの方法を用いて良い：

- 関数呼び出し
- 出力ファイル

関数呼び出しによってあなたの解を提出する場合、あなたは以下の関数を実装する必要がある。

```
std::pair<std::vector<int>, std::vector<std::pair<int, int>>>  
construct(int K)
```

- K ：アトラクションのタイプの種類数の半分である。また、それぞれのアトラクションについて、それが端点である歩道の本数として許される最大の値でもある。
- この関数は各小課題でちょうど1回呼び出される。

この関数は遊園地を表現するペア (T, E) を返す必要がある。 M をあなたの遊園地における歩道の本数とする。

- T ：アトラクションのタイプを表す長さ N の配列である。
- E ：歩道を表す長さ M の配列である。各 $0 \leq j < M$ に対し、 $E[j] = (U[j], V[j])$ は、異なるアトラクション $U[j]$ と $V[j]$ を双方向に結ぶ歩道を表す。

出力ファイルによってあなたの解を提出する場合、以下の形式でテキストファイルを作成し、提出する：

```
N M  
T[0] T[1] ... T[N-1]  
U[0] V[0]  
U[1] V[1]  
...  
U[M-1] V[M-1]
```

ここで、あなたの解が**正当である**と判定されるためには、以下の制約を満たさなければならない。

- $N \leq 2000$
- それぞれの $0 \leq i < N$ に対し $0 \leq T[i] < 2K$ であり、すべてのタイプが少なくとも1個のアトラクションに割り当てられている。
- それぞれの $0 \leq j < M$ に対し $0 \leq U[j], V[j] < N$ かつ $U[j] \neq V[j]$ 。
- 条件1と2を満たす。

制約

- $1 \leq K \leq 50$

配点

1 から 50 までの整数 K ごとに 1 個, 合計 50 個の小課題がある. 小課題 i ($1 \leq i \leq 50$) において, K の値は i である.

各小課題は以下の表に示される配点 S と遊園地のアトラクションの目標の個数 P を持つ.

小課題	S	P
1	1	2
2	8	12
3	9	24
4	9	40
5	9	50
6 - 10	4	$12 \cdot K$
11 - 12	3	$12 \cdot K$
13 - 50	1	$12 \cdot K$

それぞれの小課題において, あなたの解が正当な遊園地を表さない場合, あなたの解の得点は 0 になる (CMS では `Output isn't correct` と表示される) .

そうでない場合, あなたの得点は N とパラメータ S, P に基づき, 以下のように計算される.

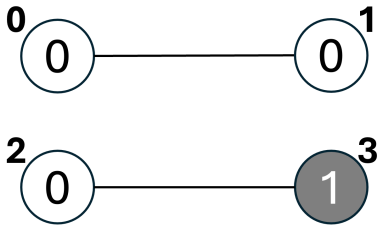
条件	得点
$N \leq P$	S
$P < N \leq 2P$	$(0.4 + 0.3 \cdot \frac{2P-N}{P}) \cdot S$
$2P < N \leq 2000$	$(0.1 + 0.3 \cdot \frac{2P}{N}) \cdot S$

例

以下の呼び出しを考える:

```
construct(1)
```

この例では $K = 1$ であるため、アトラクションのタイプは $2K = 2$ 種類ある。以下の図は $N = 4$ 個のアトラクションと $M = 2$ 本の歩道からなる正当な解を表している。アトラクション 0, 1, 2 はタイプ 0 であり、アトラクション 3 はタイプ 1 である。



2 個の興味深い三つ組が存在する：

- 三つ組 $(0, 1, 0)$ に対しては、 $(a_1, a_2, a_3) = (2, 3, 2)$ とできる。
- 三つ組 $(1, 0, 1)$ に対しては、 $(a_1, a_2, a_3) = (3, 2, 3)$ とできる。

すなわち、条件 1 が満たされている。

この関数はペア $([0, 0, 0, 1], [(0, 1), (2, 3)])$ を返すことができる。この例で示された解は $K = 1$ に対し最適であるとは限らないことに注意せよ。

採点プログラムのサンプル

入力形式：

K

出力形式

```
N M
T[0] T[1] ... T[N-1]
U[0] V[0]
U[1] V[1]
...
U[M-1] V[M-1]
```

採点プログラムのサンプルの出力は、出力ファイルに求められる形式と同一である。