

魔法城

塔什干市長希望重新規劃著名的「魔法城」遊樂園。給定一個正整數 K ，你的任務是設計一座符合以下要求的遊樂園：

- 選擇一個整數 N 作為景點的數量，並將這些設施標記為從 0 到 $N - 1$ 的整數。
- 在**不同的**景點之間添加雙向走道。它也允許兩個景點之間存在多於一條的走道。**但並非必須能夠透過走道在每一對景點之間穿梭。**
- 對於每個 i 滿足 $0 \leq i < N$ ，請為景點 i 指派類型 $T[i]$ 。共有 $2K$ 種類型，編號範圍為 0 至 $2K - 1$ 。每種類型都應被指派給至少一個景點。不同的景點可能具有相同的類型。

市場調查顯示：

- 每位遊客偏好參觀恰好三個景點，且不會連續參觀兩個同類型的景點。
- 遊客不喜歡多條走道盡頭的景點。

為了滿足**所有**潛在遊客的需求，市長對你的設計提出了兩項條件。

我們稱一個有序三元組**類型** (t_1, t_2, t_3) ，其中 $0 \leq t_1, t_2, t_3 < 2K$ ，當 $t_1 \neq t_2$ 且 $t_2 \neq t_3$ 時是**有趣的**。注意 t_1 可能等於 t_3 。因此，有 $2K \cdot (2K - 1)^2$ 有趣的三元組。

條件 1：對於每個有趣的三元組 (t_1, t_2, t_3) ，必須存在三個景點 a_1 、 a_2 、 a_3 （其中 $0 \leq a_1, a_2, a_3 < N$ ），使得：

- a_1, a_2, a_3 的類型與 t_1, t_2, t_3 的類型相符。也就是說， $T[a_1] = t_1$ ， $T[a_2] = t_2$ ，且 $T[a_3] = t_3$ 。
- 在 a_1 和 a_2 之間有一條人行道。
- 在 a_2 和 a_3 之間有一條人行道。

a_1 和 a_3 之間是否有通道並不重要。另請注意，當 $t_1 = t_3$ 時，景點 a_1 和 a_3 可能相同。

條件 2：每個景點最多可以成為 K 條走道的終點。

這項任務包含50分**部分評分**的**僅輸出**子任務。每個子任務對應一個特定的 K 值，你必須設計一個樂園，使其滿足該 K 值對應的所有上述條件。您的分數取決於解決方案中所包含的景點數量：景點數量越少，分數越高或相同。

實作細節

提交解決方案有兩種方法，每個子任務可以使用其中任何一種方法：

- 過程呼叫**

- **輸出檔案**

要透過**過程呼叫**提交解決方案，您應該實現以下過程：

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

- K ：景點類型數量的一半，也是每個景點可以作為終點的走道的最大數量。
- 對於每個子任務，此過程恰好只調用一次。

該過程應返回一對 (T, E) 來描述一個遊樂園。假設 M 是你的遊樂園走道數量。

- T ：長度為 N 的數組用於描述景點的類型。
- E ：長度為 M 的數組用於描述走道。對於每個 $0 \leq j < M$ ， $E[j] = (U[j], V[j])$ 代表連接不同景點 $U[j]$ 與 $V[j]$ 之間的雙向步道。

若要透過 **輸出檔案** 提交您的解法，請建立並提交一個符合以下格式的文字檔案：

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

請注意，您的解法必須滿足以下限制條件，才被視為**有效**：

- $N \leq 2000$
- $0 \leq T[i] < 2K$ ，且 $0 \leq i < N$ ，而且每種類型都應被分配至至少一項景點。
- $0 \leq U[j], V[j] < N$ 且 $U[j] \neq V[j]$ ，且 $0 \leq j < M$ 。
- 條件 1 和 2 必須滿足。

限制條件

- $1 \leq K \leq 50$

計分

共有 50 個子任務，每個整數 K （範圍從 1 到 50）對應一個子任務。對於子任務 i ($1 \leq i \leq 50$)， K 的值為 i 。

根據下表，每個子任務都有一個分數 S 以及一個目標吸引力數 P 。

子任務	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	$12 \cdot K$

對於每個子任務，若您的解法未能描述一個有效的遊樂園，則該解法的得分將為 0 (在 CMS 中顯示為 `Output isn't correct`)。

否則，您的分數將根據 N 以及參數 S 和 P 如下計算：

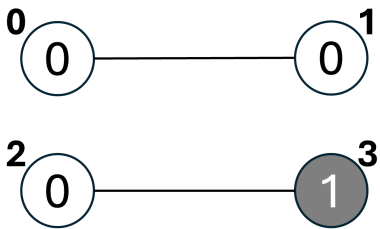
條件	分數
$N \leq P$	S
$P < N \leq 2P$	$(0.4 + 0.3 \cdot \frac{2P-N}{P}) \cdot S$
$2P < N \leq 2000$	$(0.1 + 0.3 \cdot \frac{2P}{N}) \cdot S$

樣例

請考慮以下呼叫：

```
construct(1)
```

在此範例中， $K = 1$ ，因此共有 $2K = 2$ 種遊樂設施類型。下圖顯示了一個有效的解法，其中包含 $N = 4$ 個遊樂設施與 $M = 2$ 條步道。遊樂設施 0, 1, 2 屬於 0 類型，而遊樂設施 3 則屬於 1 類型。



有兩個有趣的三元組：

- 對於類型三元組 $(0, 1, 0)$ ，我們可以選擇 $(a_1, a_2, a_3) = (2, 3, 2)$ 。

- 對於類型三元組 $(1, 0, 1)$ ，我們可以選擇 $(a_1, a_2, a_3) = (3, 2, 3)$ 。

這表示滿足條件 1。

該程序可以返回元組 $([0, 0, 0, 1], [(0, 1), (2, 3)])$ 。請注意，此範例中提供的解法對於 $K = 1$ 而言，可能並非最優解。

樣例評測器

輸入格式:

K

輸出格式:

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

請注意，樣例評分器的輸出結果符合輸出檔案所需的格式。