

Magic City

Ο δήμαρχος της Τασκένδης θέλει να επανασχεδιάσει το διάσημο πάρκο ψυχαγωγίας Magic City. Σας δίνεται ένας θετικός ακέραιος K και σας ζητείται να σχεδιάσετε ένα πάρκο ως εξής:

- Επιλέγετε έναν ακέραιο N , το πλήθος των αξιοθεάτων. Τα αξιοθέατα είναι αριθμημένα από 0 έως και $N - 1$.
- Προσθέτετε αμφίδρομους πεζόδρομους μεταξύ ζευγών **διακριτών** αξιοθεάτων. Επιτρέπεται να υπάρχουν περισσότεροι πεζόδρομοι μεταξύ του ίδιου ζεύγους αξιοθεάτων. **Δεν απαιτείται να μπορεί κανείς να ταξιδέψει μεταξύ οποιουδήποτε ζεύγους αξιοθεάτων χρησιμοποιώντας τους πεζόδρομους.**
- Για κάθε i τέτοιο ώστε $0 \leq i < N$, αναθέτετε στο αξιοθέατο i έναν τύπο $T[i]$. Υπάρχουν $2K$ τύποι αριθμημένοι από 0 έως και $2K - 1$. Κάθε τύπος πρέπει να ανατεθεί σε τουλάχιστον ένα αξιοθέατο. Διαφορετικά αξιοθέατα ενδέχεται να έχουν τον ίδιο τύπο.

Η έρευνα αγοράς αποκάλυψε ότι:

1. Κάθε επισκέπτης προτιμά να επισκέπεται τρία αξιοθέατα χωρίς να επισκεφτεί δύο αξιοθέατα του ίδιου τύπου το ένα μετά το άλλο.
2. Στους επισκέπτες δεν αρέσουν τα αξιοθέατα που είναι άκρα πολλών πεζοδρόμων.

Προκειμένου να ικανοποιήσει **όλους** τους πιθανούς επισκέπτες, ο δήμαρχος επιβάλλει δύο απαραίτητες συνθήκες για το σχέδιο σας.

Θα λέμε μία διατεταγμένη τριάδα **τύπων** (t_1, t_2, t_3) με $0 \leq t_1, t_2, t_3 < 2K$ **ενδιαφέρουσα** αν $t_1 \neq t_2$ και $t_2 \neq t_3$. Να σημειωθεί ότι το t_1 ενδέχεται να ισούται με το t_3 . Επομένως, υπάρχουν $2K \cdot (2K - 1)^2$ ενδιαφέρουσες τριάδες.

Συνθήκη 1: Για κάθε ενδιαφέρουσα τριάδα (t_1, t_2, t_3) , πρέπει να υπάρχουν τρία αξιοθέατα a_1, a_2, a_3 (με $0 \leq a_1, a_2, a_3 < N$) τέτοια ώστε:

- Οι τύποι των a_1, a_2, a_3 να είναι t_1, t_2, t_3 , αντίστοιχα. Δηλαδή, $T[a_1] = t_1$, $T[a_2] = t_2$, και $T[a_3] = t_3$.
- Να υπάρχει πεζόδρομος μεταξύ των a_1 και a_2 .
- Να υπάρχει πεζόδρομος μεταξύ των a_2 και a_3 .

Δεν έχει σημασία το κατά πόσο υπάρχει ή δεν υπάρχει πεζόδρομος μεταξύ των a_1 και a_3 . Επιπλέον, να σημειωθεί πως όταν $t_1 = t_3$, τα αξιοθέατα a_1 και a_3 μπορούν να είναι τα ίδια.

Συνθήκη 2: Κάθε αξιοθέατο μπορεί να είναι άκρο **το πολύ** K πεζοδρόμων.

Αυτό το θέμα αποτελείται από 50 **output-only** υποπροβλήματα με **μερική βαθμολόγηση**. Κάθε υποπρόβλημα αντιστοιχεί σε μια συγκεκριμένη τιμή του K και πρέπει να σχεδιάσετε ένα πάρκο που

να ικανοποιεί όλες τις παραπάνω συνθήκες για αυτήν την τιμή του K . Η βαθμολογία σας εξαρτάται από το πλήθος αξιοθεάτων στη λύση σας: λιγότερα αξιοθέατα οδηγούν σε υψηλότερη ή ίση βαθμολογία.

Λεπτομέρειες υλοποίησης

Υπάρχουν δύο τρόποι να υποβάλετε τη λύση σας και μπορείτε να χρησιμοποιείτε οποιονδήποτε από τους δύο για κάθε υποπρόβλημα:

- **Κλήση συνάρτησης**
- **Αρχείο εξόδου**

Για να υποβάλετε της λύσης σας μέσω μιας **κλήσης συνάρτησης**, πρέπει να υλοποιήσετε την παρακάτω συνάρτηση.

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

- K : το μισό του πλήθους των τύπων αξιοθεάτων και επίσης το μέγιστο επιτρεπόμενο πλήθος πεζόδρομων που μπορούν να είναι άκρα ενός αξιοθέατου.
- Η συνάρτηση αυτή καλείται ακριβώς μία φορά για κάθε υποπρόβλημα.

Η συνάρτηση πρέπει να επιστρέφει ένα ζεύγος (T, E) που να περιγράφει το πάρκο ψυχαγωγίας. Έστω M το πλήθος πεζόδρομων στο πάρκο σας.

- T : ένας πίνακας μήκους N που να περιγράφει τους τύπους των αξιοθεάτων.
- E : ένας πίνακας μήκους M που να περιγράφει τους πεζόδρομους. Για κάθε $0 \leq j < M$, το στοιχείο $E[j] = (U[j], V[j])$ αναπαριστά έναν αμφίδρομο πεζόδρομο μεταξύ δύο διαφορετικών αξιοθεάτων $U[j]$ και $V[j]$.

Για να υποβάλετε τη λύση σας μέσω **αρχείου εξόδου**, δημιουργήστε και υποβάλετε ένα αρχείο κειμένου με την παρακάτω μορφή:

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

Να σημειωθεί ότι η λύση σας πρέπει να ικανοποιεί τους παρακάτω περιορισμούς για να θεωρηθεί **έγκυρη**:

- $N \leq 2000$
- $0 \leq T[i] < 2K$ για κάθε $0 \leq i < N$, και κάθε τύπος πρέπει να αντιστοιχισθεί σε τουλάχιστον ένα αξιοθέατο.
- $0 \leq U[j], V[j] < N$ και $U[j] \neq V[j]$ για κάθε $0 \leq j < M$.

- Οι συνθήκες 1 και 2 πρέπει να ικανοποιούνται.

Περιορισμοί

- $1 \leq K \leq 50$

Βαθμολόγηση

Υπάρχουν 50 υποπροβλήματα, ένα για κάθε ακέραιο K από 1 έως και 50. Για το i -οστό υποπρόβλημα (όπου $1 \leq i \leq 50$), η τιμή του K είναι i .

Κάθε υποπρόβλημα έχει μία βαθμολογία S και ένα επιθυμητό πλήθος αξιοθεάτων P , σύμφωνα με τον παρακάτω πίνακα.

Υποπροβλήματα	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	$12 \cdot K$

Για κάθε ένα υποπρόβλημα, αν η λύση σας δεν περιγράφει ένα έγκυρο πάρκο ψυχαγωγίας, τότε η βαθμολογία της λύσης σας θα είναι 0 (ένδειξη `Output isn't correct` στο CMS).

Διαφορετικά, η βαθμολογία σας υπολογίζεται με βάση το N και τις παραμέτρους S και P ως εξής:

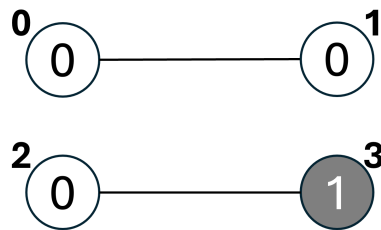
Συνθήκη	Βαθμολογία
$N \leq P$	S
$P < N \leq 2P$	$(0.4 + 0.3 \cdot \frac{2P-N}{P}) \cdot S$
$2P < N \leq 2000$	$(0.1 + 0.3 \cdot \frac{2P}{N}) \cdot S$

Παράδειγμα

Θεωρήστε την παρακάτω κλήση:

```
construct(1)
```

Σε αυτό το παράδειγμα, $K = 1$, οπότε υπάρχουν $2K = 2$ τύποι αξιοθεάτων. Η εικόνα παρακάτω δείχνει μία έγκυρη λύση με $N = 4$ αξιοθεάτα και $M = 2$ πεζόδρομους. Τα αξιοθεάτα 0, 1, 2 είναι τύπου 0, και το αξιοθεάτο 3 είναι τύπου 1.



Υπάρχουν δύο ενδιαφέρουσες τριάδες:

- Για την τριάδα τύπων $(0, 1, 0)$, μπορούμε να επιλέξουμε $(a_1, a_2, a_3) = (2, 3, 2)$.
- Για την τριάδα τύπων $(1, 0, 1)$, μπορούμε να επιλέξουμε $(a_1, a_2, a_3) = (3, 2, 3)$.

Αυτό σημαίνει ότι η συνθήκη 1 ικανοποιείται.

Η συνάρτηση μπορεί να επιστρέψει το ζεύγος $([0, 0, 0, 1], [(0, 1), (2, 3)])$. Να σημειωθεί ότι η λύση που παρέχεται για αυτό το παράδειγμα ενδέχεται να μην είναι βέλτιστη για $K = 1$.

Ενδεικτικός βαθμολογητής

Μορφή εισόδου:

K

Μορφή εξόδου:

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

Να σημειωθεί ότι η έξοδος του υποδειγματικού βαθμολογητή ταιριάζει με την απαιτούμενη μορφή του αρχείου εξόδου.