

サイバーランド

3742年となった今、サイバーランド国 (Cyberland) がAPIOを開催する番がやってきた。この世界には、0 から $N - 1$ までの番号が付けられた N 個の国と、0 から $M - 1$ までの番号が付けられた M 本の双方向に移動可能な道路が存在する。各道路は二つの異なる国 $x[i]$ と $y[i]$ を繋ぎ、その道路を通るには一定の時間 $c[i]$ がかかる。あなたの国以外のすべての参加者は、APIOのためにサイバーランドに集まっている。あなたは国 0 に住んでおり、サイバーランドは国 H である。国で一番賢いあなたの助けが、再び至急必要とされている。具体的には、あなたの国からサイバーランドに到着するのに必要な最短時間を求めるよう要求されている。

いくつかの国は、今までにかかった時間をなかったことにする能力を持つ。また、いくつかの国は今までにかかった時間を 2 で割った値にする能力 (二分能力) をもつ。各国は何度でも訪れられる。国を訪れる度に、その国が持つ特殊能力を使うかどうか選ぶことができる。ただし、特殊能力を使えるのは 1 回の訪問で多くとも 1 回だけである (つまり、特殊能力を複数回使うには複数回訪れる必要がある)。その上、サイバーランド化学財団 (Cyberland Chemistry Foundation) に逮捕されかねないので、二分能力は最大 K 回までしか使えない。APIOの素晴らしいコンテストはあなたがサイバーランドに到着し次第すぐに始まるので、サイバーランドに着いたらどこにも移動できなくなる。

配列 arr が与えられ、 arr_i が国 i の持つ特殊能力を表す。特殊能力は以下の 3 種類である:

- $arr_i = 0$: この国は今までにかかった時間を 0 にできる。
- $arr_i = 1$: この国で今までにかかった時間は変えられない。
- $arr_i = 2$: この国は今までにかかった時間を 2 で割った値にできる。

$arr_0 = arr_H = 1$ が成立することが保証されている。言い換えると、サイバーランドとあなたの国は特殊能力を持っていない。

あなたの国はAPIOに一瞬たりとも遅刻したくないので、サイバーランドに到着するまでの最短時間を求める必要がある。サイバーランドに到着できない場合は -1 と答えよ。

実装の詳細

あなたは、以下の関数を実装する必要がある:

```
double solve(int N, int M, int K, int H, std::vector<int> x, std::vector<int> y, std::vector<int> c, std::vector<int> arr);
```

- N : 国の個数。
- M : 双方向道路の本数。

- K : 二分能力の使用回数の最大値.
- H : サイバーランドの国番号.
- x, y, c : 長さ M の3つの配列. 組 $(x[i], y[i], c[i])$ は, i 番目の無向辺が国 $x[i]$ と国 $y[i]$ を結び, $c[i]$ の時間コストを持つことを表す.
- arr : 長さ N の配列. $arr[i]$ は国 i の持つ特殊能力を表す.
- この関数は, あなたの国からサイバーランドに到着できる場合はその最短時間を, できない場合は -1 を返さなければならない.
- この関数は複数回呼び出されうる.

あなたの戻り値を ans_1 , 戻り値の基準値を ans_2 において, $\frac{|ans_1 - ans_2|}{\max\{ans_2, 1\}} \leq 10^{-6}$ が満たされる時のみ, あなたの戻り値を正解とみなす.

注: この関数は複数回呼び出されうるので, 前回の呼び出しのデータの残りが今回の呼び出しに与える影響に注意する必要がある.

入出力例

入出力例 1

以下のような関数呼び出しを考える:

```
solve(3, 2, 30, 2, {1, 2}, {2, 0}, {12, 4}, {1, 2, 1});
```

サイバーランドに到着した後は移動できなくなるので, サイバーランドまでの経路は $0 \rightarrow 2$ しかない. 経過時間の計算は以下になる.

国番号	経過時間
0	0
2	$0 + 4 \rightarrow 4(\text{和}) \rightarrow 4(\text{特殊能力})$

したがって, 関数は 4 を返さなければならない.

入出力例 2

以下のような関数呼び出しを考える:

```
solve(4, 4, 30, 3, {0, 0, 1, 2}, {1, 2, 3, 3}, {5, 4, 2, 4}, {1, 0, 2, 1});
```

あなたの国からサイバーランドに行く経路には $0 \rightarrow 1 \rightarrow 3$ と $0 \rightarrow 2 \rightarrow 3$ の2つがある.

$0 \rightarrow 1 \rightarrow 3$ の経路では, 経過時間の計算は以下になる.

国番号	経過時間
0	0
1	$0 + 5 \rightarrow 5(\text{和}) \rightarrow 0(\text{特殊能力})$
3	$0 + 2 \rightarrow 2(\text{和}) \rightarrow 2(\text{特殊能力})$

$0 \rightarrow 2 \rightarrow 3$ の経路では, 経過時間の計算は以下のようになる.

国番号	経過時間
0	0
2	$0 + 4 \rightarrow 4(\text{和}) \rightarrow 2(\text{特殊能力})$
3	$2 + 4 \rightarrow 6(\text{和}) \rightarrow 6(\text{特殊能力})$

したがって, 関数は 2 を返さなければならない.

制約

- $2 \leq N \leq 10^5$ かつ $\sum N \leq 10^5$.
- $0 \leq M \leq \min\{10^5, \frac{N(N-1)}{2}\}$ かつ $\sum M \leq 10^5$.
- $1 \leq K \leq 10^6$.
- $1 \leq H < N$.
- $0 \leq x[i], y[i] < N$ かつ $x[i] \neq y[i]$.
- $1 \leq c[i] \leq 10^9$.
- $arr[i] \in \{0, 1, 2\}$.
- どの国のペアも, 多くとも一本の道路でしか繋がっていない.

小課題

1. (5 点): $N \leq 3, K \leq 30$.
2. (8 点): $M = N - 1, K \leq 30, arr[i] = 1, M$ 本の辺を通じて任意の国から任意の別の国に移動できる.
3. (13 点): $M = N - 1, K \leq 30, arr[i] \in \{0, 1\}, M$ 本の辺を通じて任意の国から任意の別の国に移動できる.
4. (19 点): $M = N - 1, K \leq 30, x[i] = i, y[i] = i + 1$.
5. (7 点): $K \leq 30, arr[i] = 1$.
6. (16 点): $K \leq 30, arr[i] \in \{0, 1\}$.
7. (29 点): $K \leq 30$.
8. (3 点): 追加の制約はない.

採点プログラムのサンプル

採点プログラムのサンプルは以下の形式で入力を読み込む.

- 1 行目: T

続く T 個のテストケースそれぞれについて:

- 1 行目: $N\ M\ K$
- 2 行目: H
- 3 行目: $arr[0]\ arr[1]\ arr[2]\ \dots\ arr[N-1]$
- $4+i$ 行目 ($0 \leq i \leq M-1$): $x[i]\ y[i]\ z[i]$

採点プログラムのサンプルは以下の形式であなたの答えを出力する.

各テストケースについて:

- 1 行目: `solve` の戻り値