

Cyberland

The year 3742 has arrived, and now it is Cyberland's turn to host the APIO. In this world, there are N countries indexed from 0 to $N - 1$, along with M bidirectional roads (allowing travel in both directions) indexed from 0 to $M - 1$. The i -th road ($0 \leq i < M$) connects two different countries, $x[i]$ and $y[i]$, and requires a certain amount of time $c[i]$ to pass the road. All participants have gathered in Cyberland for the APIO, except for your country. You are living in country 0, and Cyberland is country H . As the cleverest person in your country, your assistance is urgently needed once again. To be more specific, you are asked to determine the minimum time required to reach Cyberland from your country.

Some countries can clear your total passing time. Also, some countries can divide your total passing time by 2 (divide-by-2 ability). You can visit a country repeatedly. Every time you visit a country, **you may choose whether to use the special ability in the country**. But you can use the special ability **at most once in a single visit** (which means that special ability can be used multiple times by visiting the country multiple times). Moreover, you can only use the divide-by-2 ability **at most K times** in case of being caught by Cyberland Chemistry Foundation. **Once you reached Cyberland, you cannot move anywhere** because the great APIO contest will be held soon.

An array arr is given, where arr_i ($0 \leq i < N$) shows the special abilities of country i . There are 3 types of special abilities:

- $arr_i = 0$, means this country makes the passing time 0.
- $arr_i = 1$, means the passing time remains unchanged at this country.
- $arr_i = 2$, means this country divides the passing time by 2.

It is guaranteed that $arr_0 = arr_H = 1$ holds. In other words, Cyberland and your country do not have any special abilities.

Your country does not want to miss any moment of APIO, so you need to find the minimum time to reach Cyberland. If you cannot reach to Cyberland, your answer should be -1 .

Implementation Details

You need to implement the following function:

```
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 : The number of countries.
- M : The number of bidirectional roads.
- K : The limit of divide-by-2 ability usage.
- H : The index of the country Cyberland.
- x, y, c : three arrays with a length of M . the tuple $(x[i], y[i], c[i])$ represents the i -th undirected edge which connects country $x[i]$ and $y[i]$, with time cost $c[i]$.
- arr : an array with a length of N . $arr[i]$ represents the special ability of country i .
- This procedure should return the minimum time to reach Cyberland from your country if you can reach Cyberland, and -1 if you cannot do so.
- This procedure can be called more than once.

Assume that return value of contestant's is ans_1 , and the return value of standard's is ans_2 , your return value is considered correct if and only if $\frac{|ans_1 - ans_2|}{\max\{ans_2, 1\}} \leq 10^{-6}$.

Note: Since the procedure can be called more than once, contestants need to pay attention to the impact of the remaining data of the previous call on the current call.

Examples

Example 1

Consider the following call:

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

The only path to Cyberland is $0 \rightarrow 2$, because you can not move to anywhere after reaching Cyberland. The calculation of passing time is shown as below.

country number	passing time
0	0
2	$0 + 4 \rightarrow 4$ (sum) $\rightarrow 4$ (special ability)

Therefore, the procedure should return 4.

Example 2

Consider the following call:

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

There are two paths from your country to Cyberland. They are: $0 \rightarrow 1 \rightarrow 3$ and $0 \rightarrow 2 \rightarrow 3$.

If your path is $0 \rightarrow 1 \rightarrow 3$, the calculation of passing time is shown as below.

country number	passing time
0	0
1	0 + 5 → 5 (sum) → 0 (special ability)
3	0 + 2 → 2 (sum) → 2 (special ability)

If your path is $0 \rightarrow 2 \rightarrow 3$, the calculation of passing time is shown as below.

country number	passing time
0	0
2	0 + 4 → 4 (sum) → 2 (special ability)
3	2 + 4 → 6 (sum) → 6 (special ability)

Therefore, the procedure should return 2.

Constraints

- $2 \leq N \leq 10^5$, and $\sum N \leq 10^5$.
- $0 \leq M \leq \min\{10^5, \frac{N(N-1)}{2}\}$, and $\sum M \leq 10^5$.
- $1 \leq K \leq 10^6$.
- $1 \leq H < N$.
- $0 \leq x[i], y[i] < N$, and $x[i] \neq y[i]$.
- $1 \leq c[i] \leq 10^9$.
- $arr[i] \in \{0, 1, 2\}$.
- It is guaranteed that every pair of countries is connected by at most one road.

Subtasks

1. (5 points): $N \leq 3, K \leq 30$.
2. (8 points): $M = N - 1, K \leq 30, arr[i] = 1$, you can travel from any countries to another via the M edges.
3. (13 points): $M = N - 1, K \leq 30, arr[i] \in \{0, 1\}$, you can travel from any countries to another via the M edges.
4. (19 points): $M = N - 1, K \leq 30, x[i] = i, y[i] = i + 1$.
5. (7 points): $K \leq 30, arr[i] = 1$.
6. (16 points): $K \leq 30, arr[i] \in \{0, 1\}$.
7. (29 points): $K \leq 30$.
8. (3 points): No additional constraints.

Sample Grader

The sample grader reads the input in the following format:

- line 1: T

For each of the following T test cases:

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

The sample grader prints your answers in the following format:

For each test cases:

- line 1: the return value of `solve`