第14回 日本情報オリンピック 本選

問題３ JOI公園(JOI Park) 解説

二階堂建人

問題概要

広場が N 個ある公園があり、広場どうしは M 本の道でつながっている。道 i は広場 Ai と 広場 Bi をつないでおり、長さは Di である。

まず、値 X を決め、広場 1 から距離 X 以内の広場どうしを全て地下道でつなぐ。これにはコストが C*X かかる。

次に、地下道でつながれている広場どうしをつないでいる道をなくす。

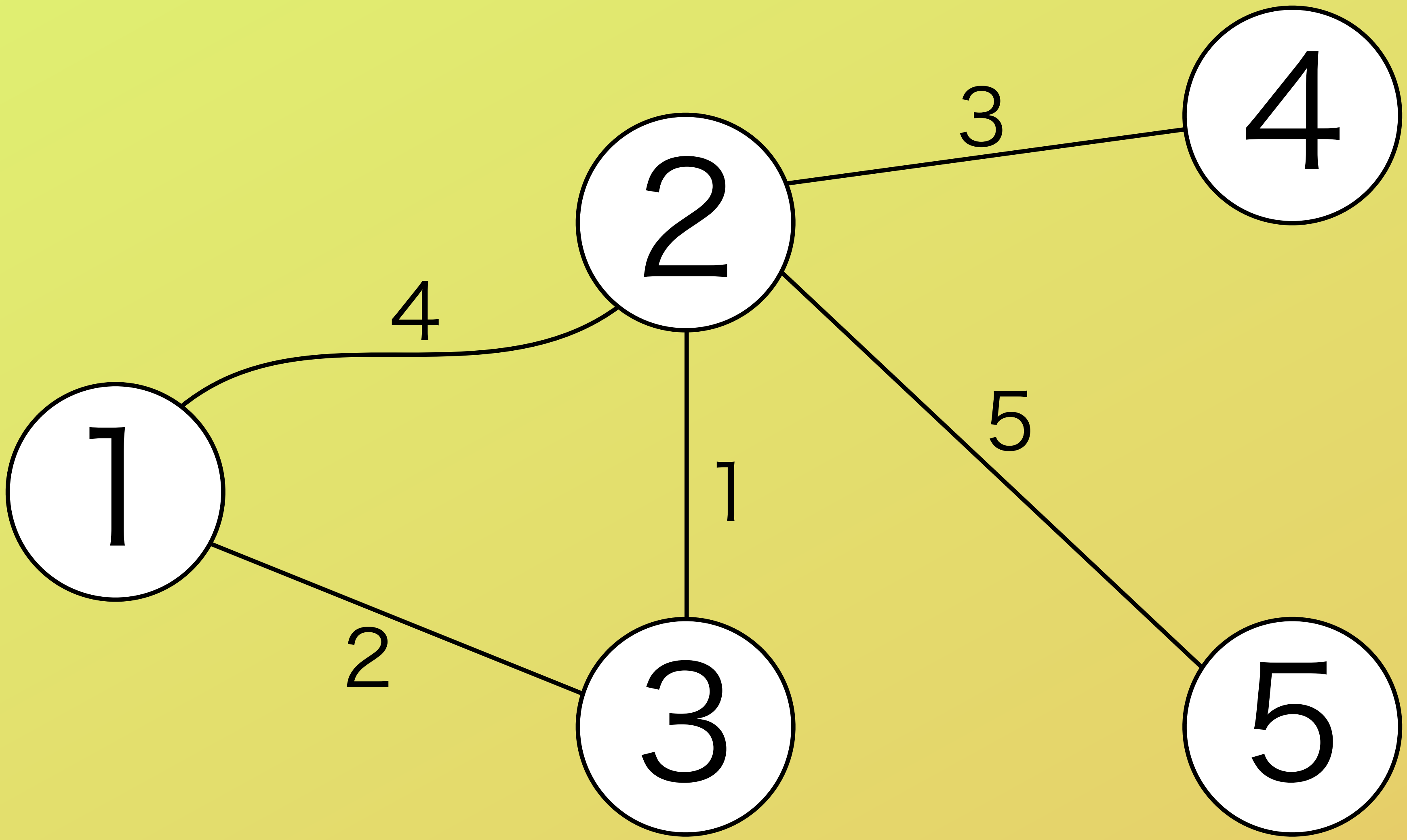
最後に、残った道を補修する。道 i を補修するときにかかるコストは Di である。

かかるコストの和の最小値を求めよ。

サンプル

入出力例 1

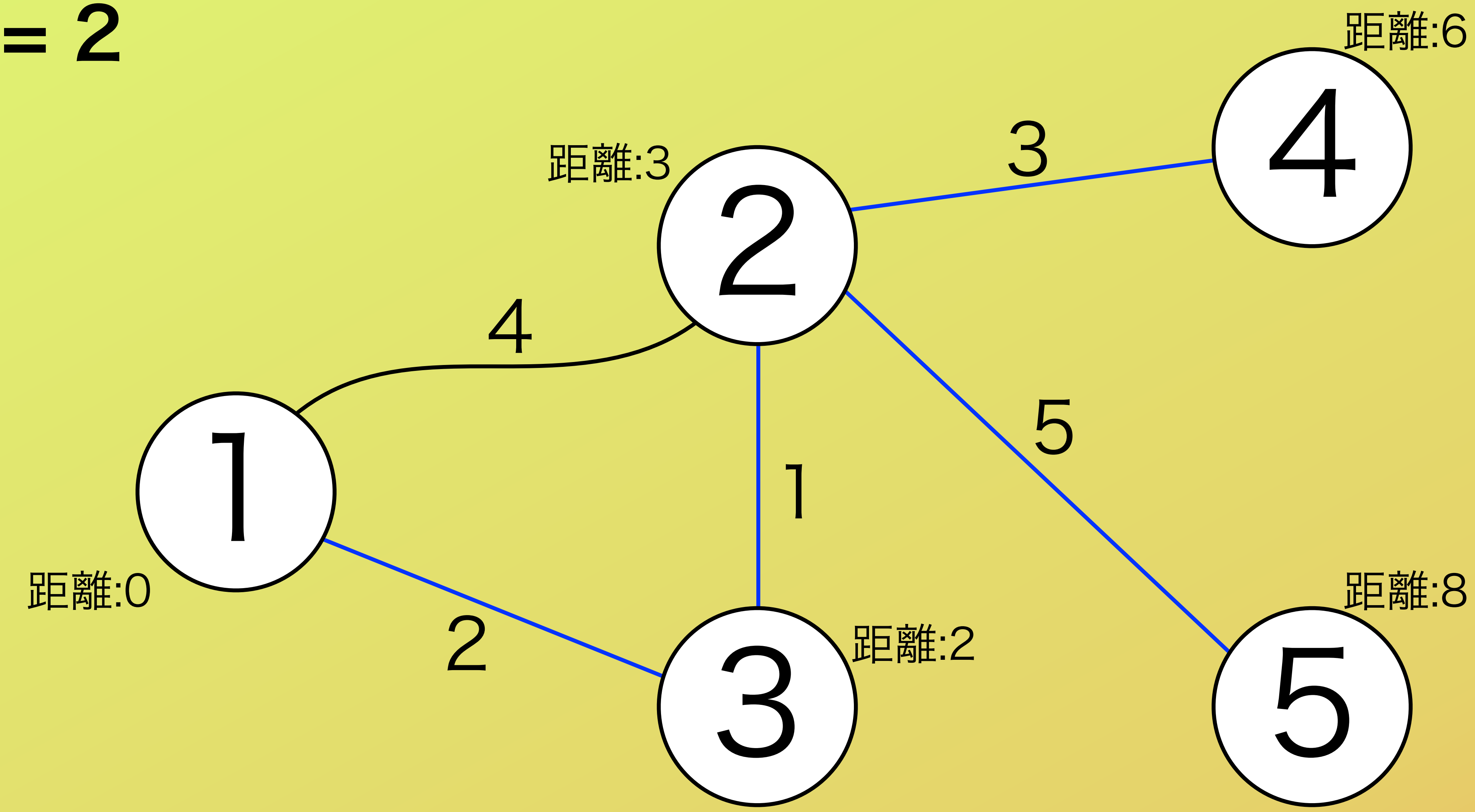
C = 2



サンプル

入出力例 1

C = 2

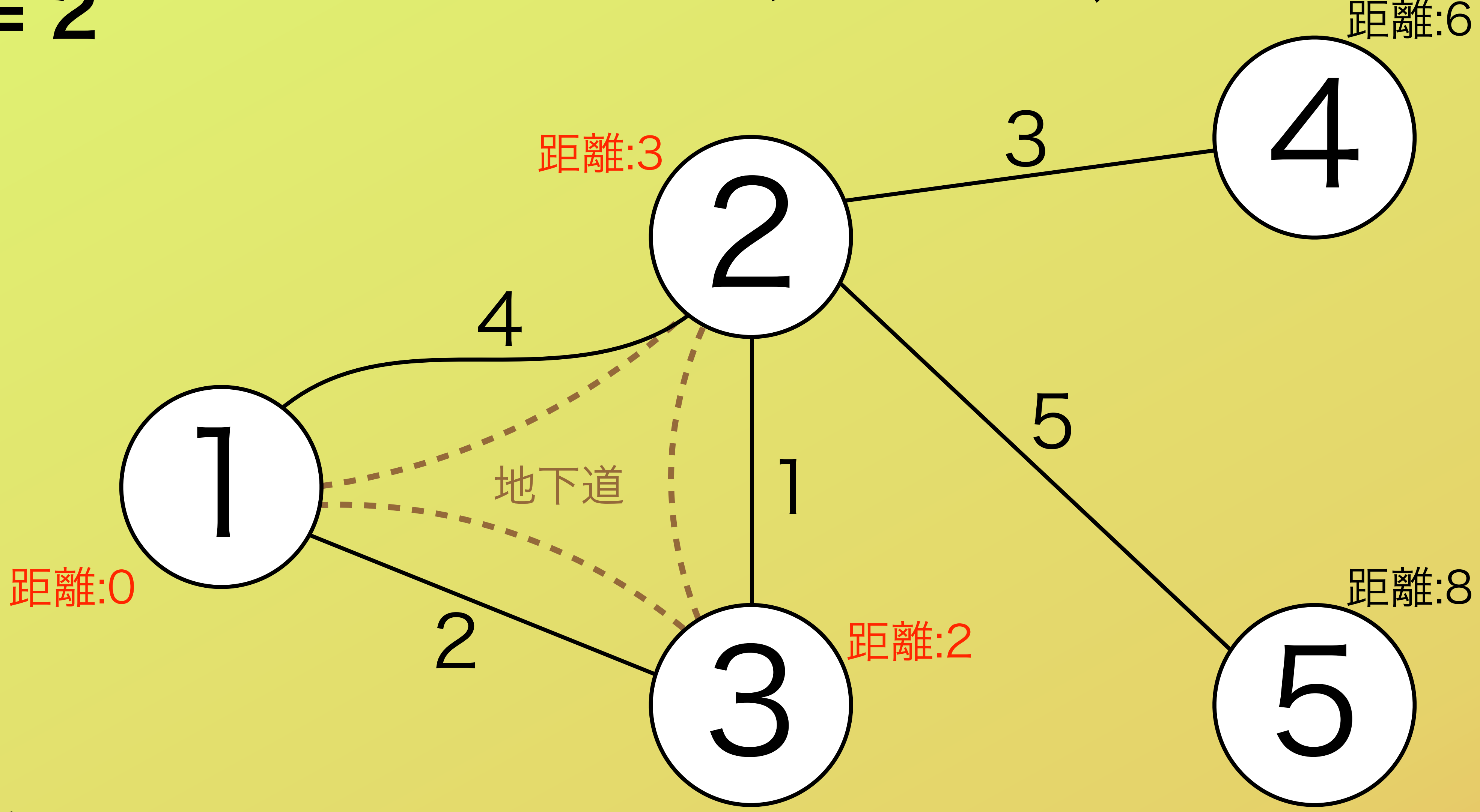


サンプル

入出力例 1

C = 2

Xを3にすると、



まず、広場1との距離が3以内の広場どうしを地下道でつなぐ。

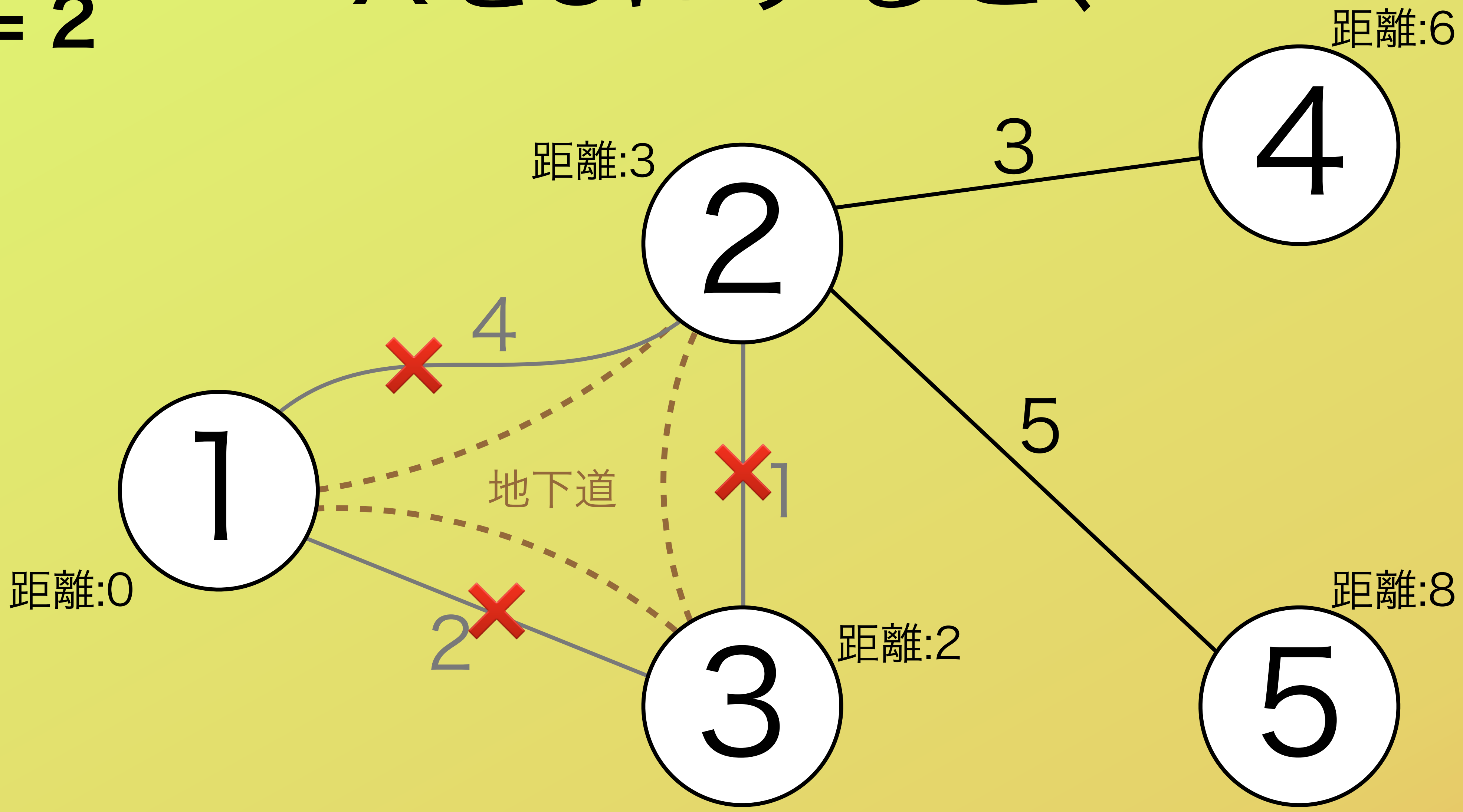
このとき、コストは C*X = 2*3 = 6 かかる。

サンプル

入出力例 1

C = 2

Xを3にすると、



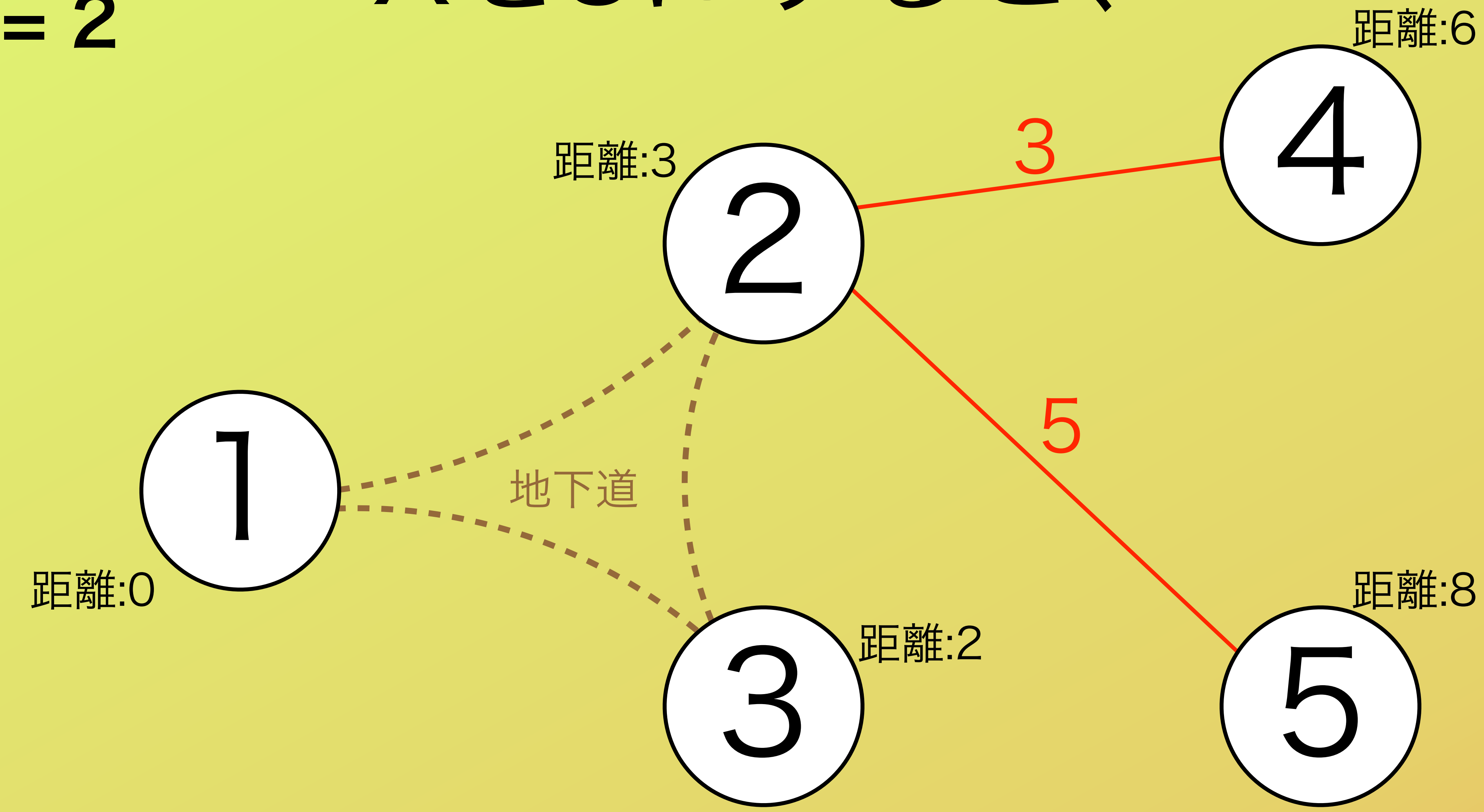
次に、地下道でつながれた広場どうしをつなぐ道をなくす。

サンプル

入出力例 1

C = 2

Xを3にすると、



最後に、残った道を補修する。

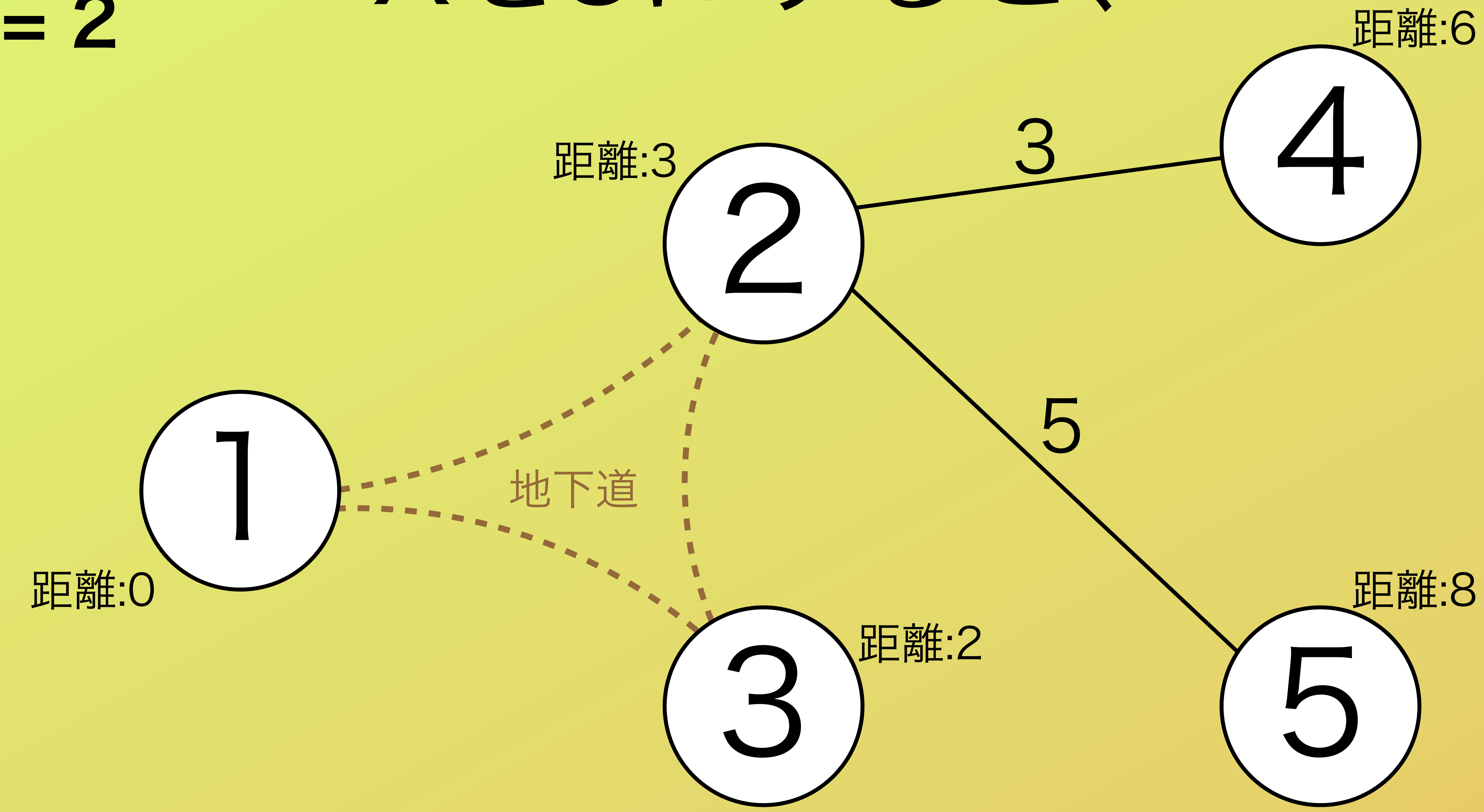
このとき、コストは補修する道の長さの和 = 3+5 = 8 かかる。

サンプル

入出力例 1

C = 2

Xを3にすると、



地下道のコストが 6、道の補修のコストが 8 かかったので、

コストの和は 14 となる。

距離の求め方

まずは、広場１からの距離を求める必要がある。

広場を頂点、道を辺としたグラフの

最短経路問題

を解けばよい。

距離の求め方

最短経路問題を解くアルゴリズムには、

・Dijkstra 法（ダイクストラ）

・Bellman-Ford 法（ベルマンフォード）

・Warshall-Floyd 法（ワーシャルフロイド）

等がある。

これらのアルゴリズムは書籍・WEB共に文献がたくさんあるので、もし知らなかった人は各自で調べましょう。

書籍の例：プログラミングコンテストチャレンジブック（蟻本）

距離の求め方

それぞれのアルゴリズムの計算量は、

・Dijkstra 法：O(M log N) （実装によりますが）

・Bellman-Ford 法：O(NM)

・Warshall-Floyd 法：O(N^3)

小課題１では N ≦ 100, M ≦ 200 で、小課題２では N ≦ 100, M ≦ 4000 なので、どの方法を使っても間に合う。

小課題３では N ≦ 100000, M ≦ 200000 なので、Dijkstra 法を使わないと間に合わない。

小課題1（15点）

制約

・N ≦ 100

・M ≦ 200

・C ≦ 100

・Di ≦ 10

Di（道の長さ）がとても小さい。

小課題1（15点）

X の値として意味のあるものを考える。

・広場 1 から最も遠い広場までの距離より大きい X は試す意味がない。

→ それより大きくしても地下道は増えない。

$N \leqq 100$, $D_i \leqq 10$ なので、広場 1 から最も遠い広場までの距離は高々 1000 程度。

小課題1（15点）

X を 0 から 1000 まで試し、それぞれの場合にかかるコストのうちの最小値を求めればよい。

かかるコストは素直に、地下道でつながるような広場を調べ、補修しなければならない道を調べればよい。

計算量は、O(試すXの個数 * (N+M))（+最短路の計算量）

であり、間に合う。

ちなみに、計算量を表す時の“+”はmaxのような意味を表します。

例えば、O(N+M) = O(max(N,M))

小課題2(45点）

制約

・$N \leqq 100$

・$M \leqq 4000$

・$C \leqq 100000$

・$D_i \leqq 100000$

D_i（道の長さ）が大きい。

小課題2(45点）

X の値として意味のあるものをさらに考える。

異なるいくつかの値で地下道のつなぎ方が同じならば、その中で最も小さい値のみを試せばよい。

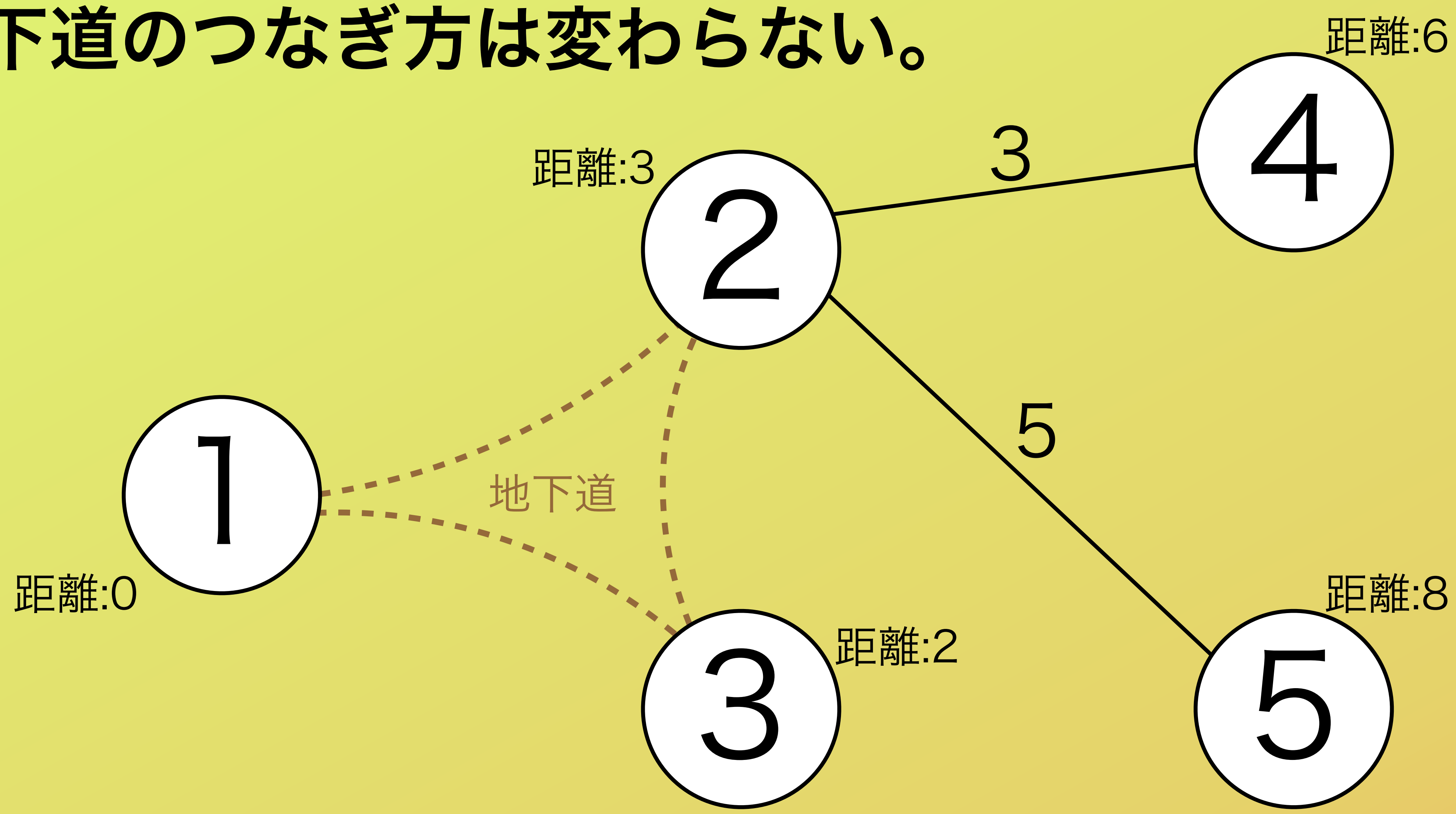
→ ちょうど地下道が増えるようなぎりぎりの値のみを試せばよい。

→ 広場 1 からある広場までの距離の値のみを X として試せばよい。

サンプル

X を 3 にした場合と 4 や 5 にした場合では、

地下道のつなぎ方は変わらない。



小課題2(45点)

広場 1 から各広場までの距離の値をそれぞれ X の値として試し、それぞれの場合にかかるコストの最小値を求めればよい。

かかるコストは小課題 1 と同様にして調べればよい。

X の候補は高々 N 通り。

計算量は、O(N * (N+M)) (+最短路の計算量)

であり、間に合う。

小課題3（40点）

制約

・$N \leqq 100000$

・$M \leqq 200000$

・$C \leqq 100000$

・$D_i \leqq 100000$

N,M が大きい。

小課題3(40点）

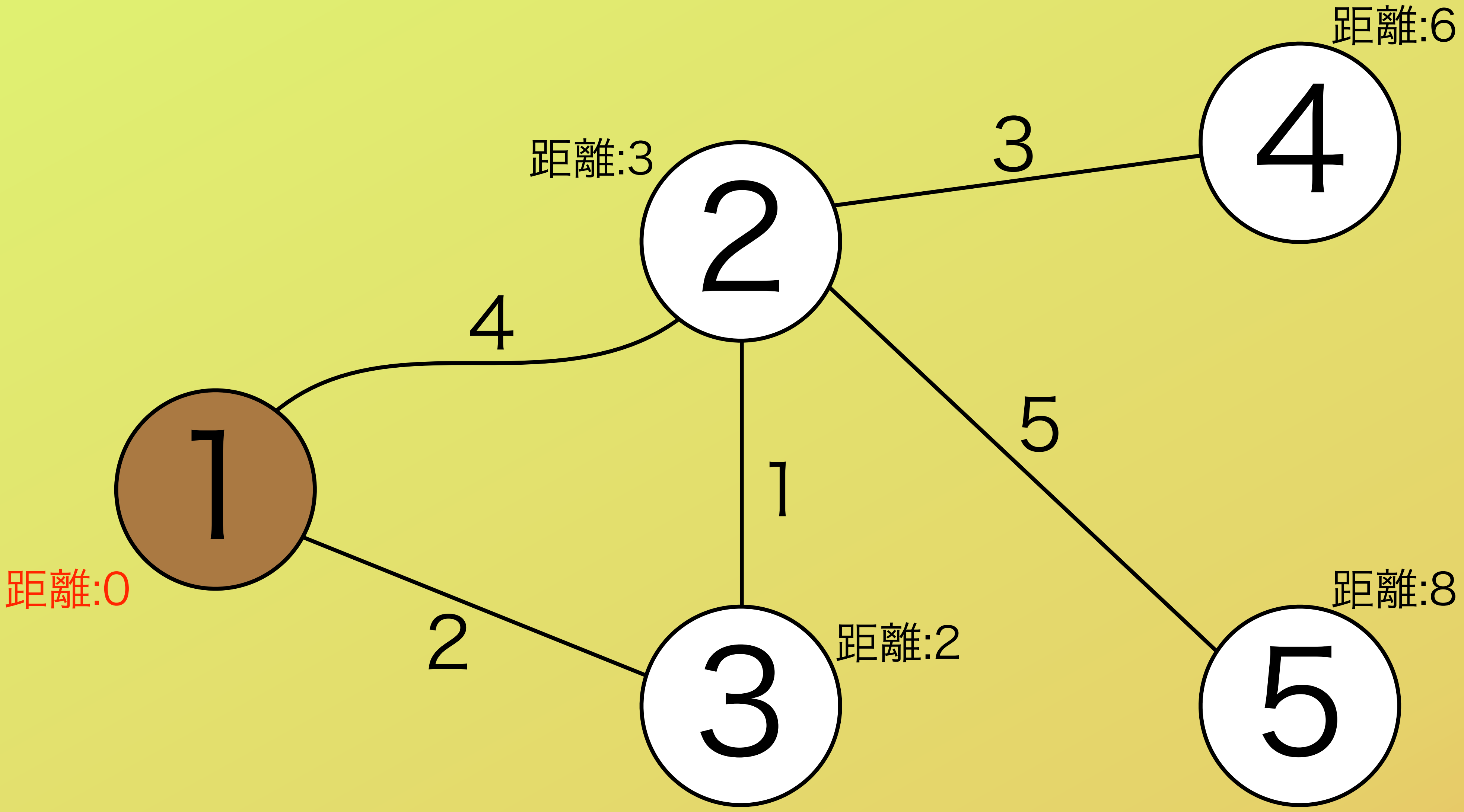
X の値が増えていくにつれ、地下道の本数は増えていき、補修する必要のある道の本数は減っていく。

言い直すと、X の値を小さい方から試していくとき、補修する必要のある道の本数は単調に減少する。

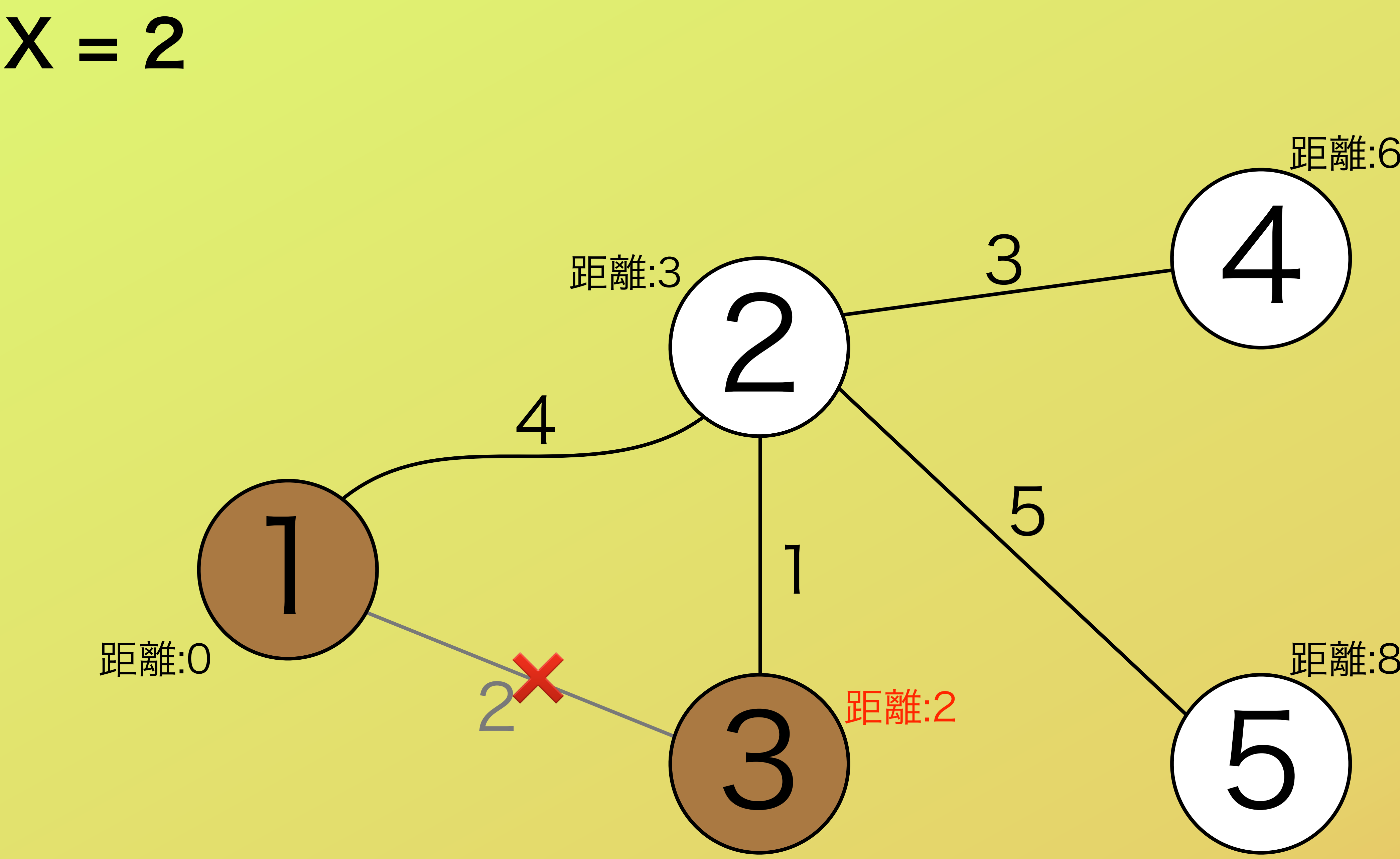
X の値を増やすたびに、それによって補修する必要の無くなった道を消していくことにする。

サンプル

X = 0

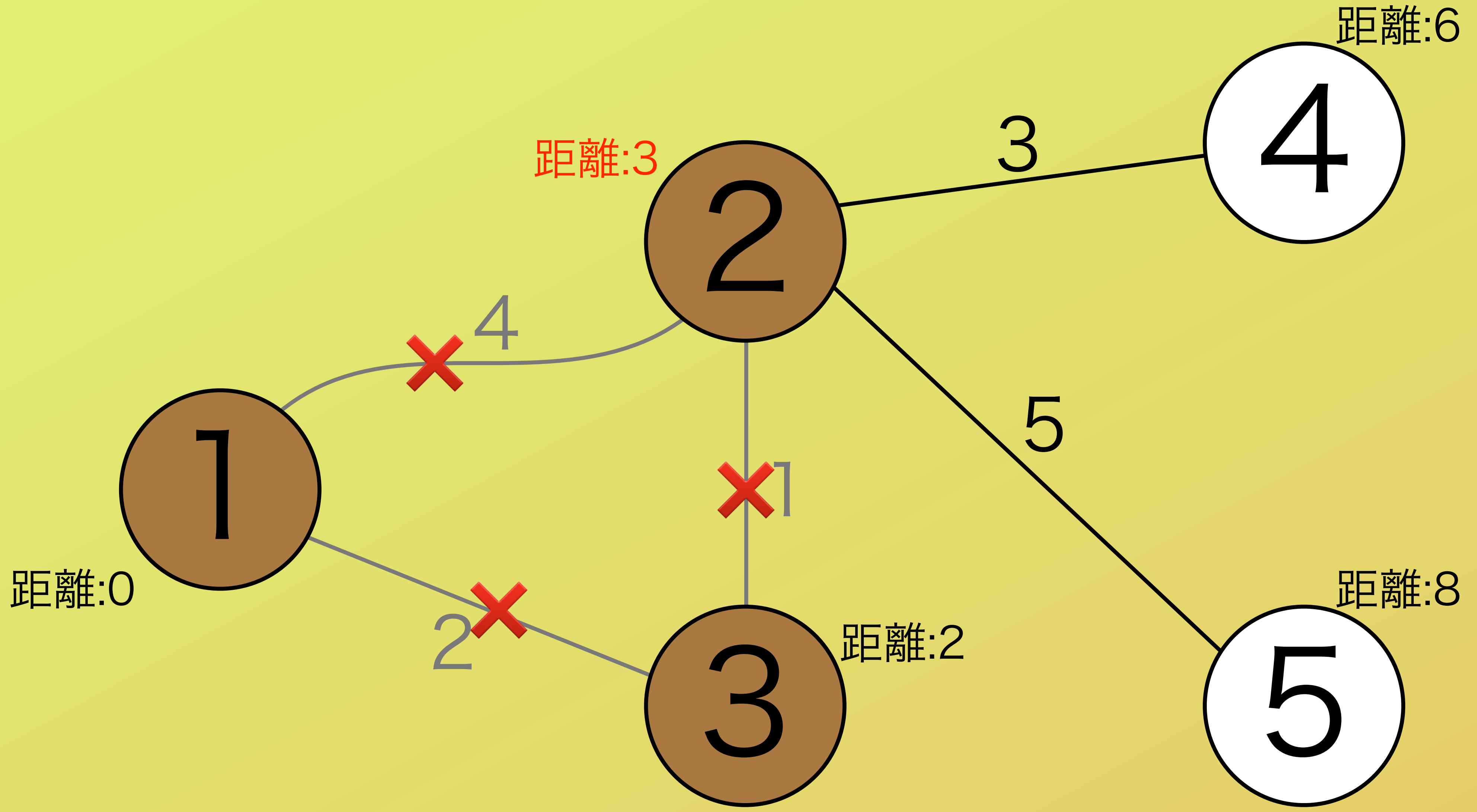


サンプル



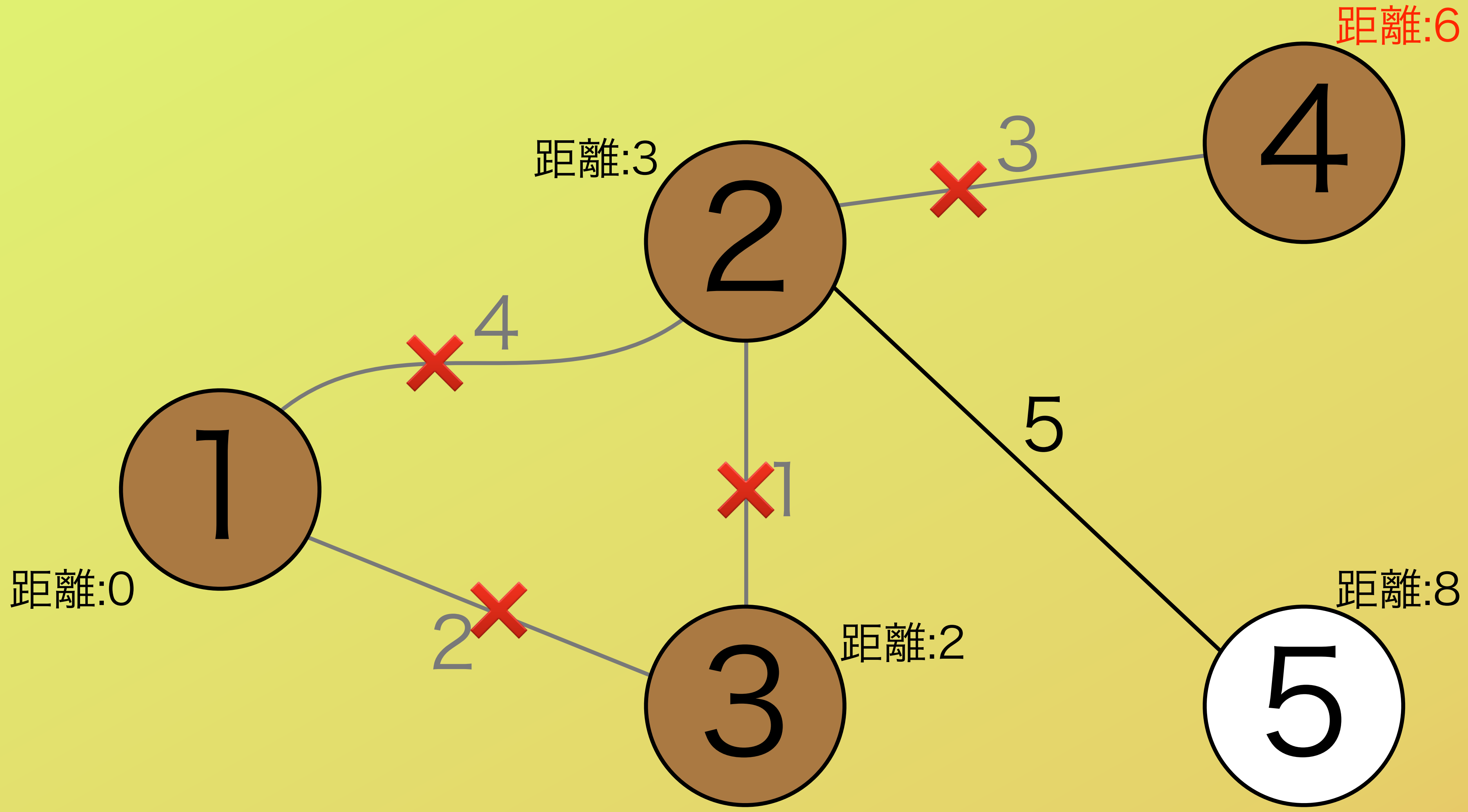
サンプル

X = 3



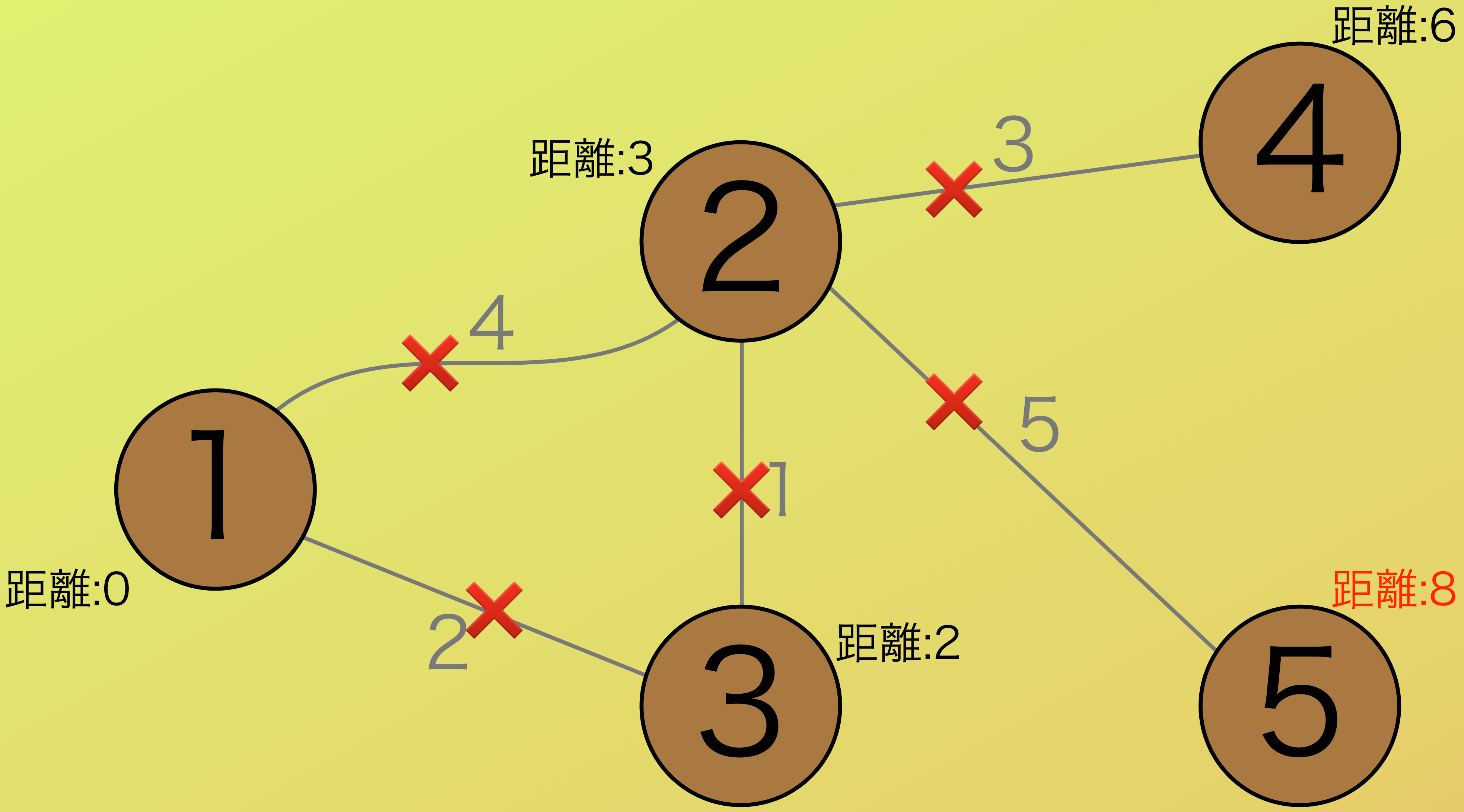
サンプル

X = 6



サンプル

X = 8



小課題3（40点）

X の値として広場１から広場ｉまでの距離の値を試すとき、新たに補修する必要がなくなる道は？

・広場ｉに直接つながっている道

・かつ、相手の広場が既に地下道でつながっている広場であるような道

ということは X の値を試すごとに、広場ｉに直接つながっている道それぞれについて、「相手の広場が既に地下道でつながっているかどうかを調べ、つながっているなら道をなくし、そうでないなら何もしない」という操作をすれば良い。

小課題3（40点）

計算量は、

O(N +「各広場につながっている道の本数の和」)

であるが、それぞれの道がつながっている広場が２個ずつであることを考えると「各広場につながっている道の本数の和」は「道の本数×２」に等しいため、

O(N + M)

となる。試す X の値をソートすることも考慮すると、

O((N log N) + N + M) (+最短路の計算量)

となり、最短路を求めるアルゴリズムにDijkstra法を使っていれば間に合う。

得点分布