Tree--->path

Writer : platypus

概要

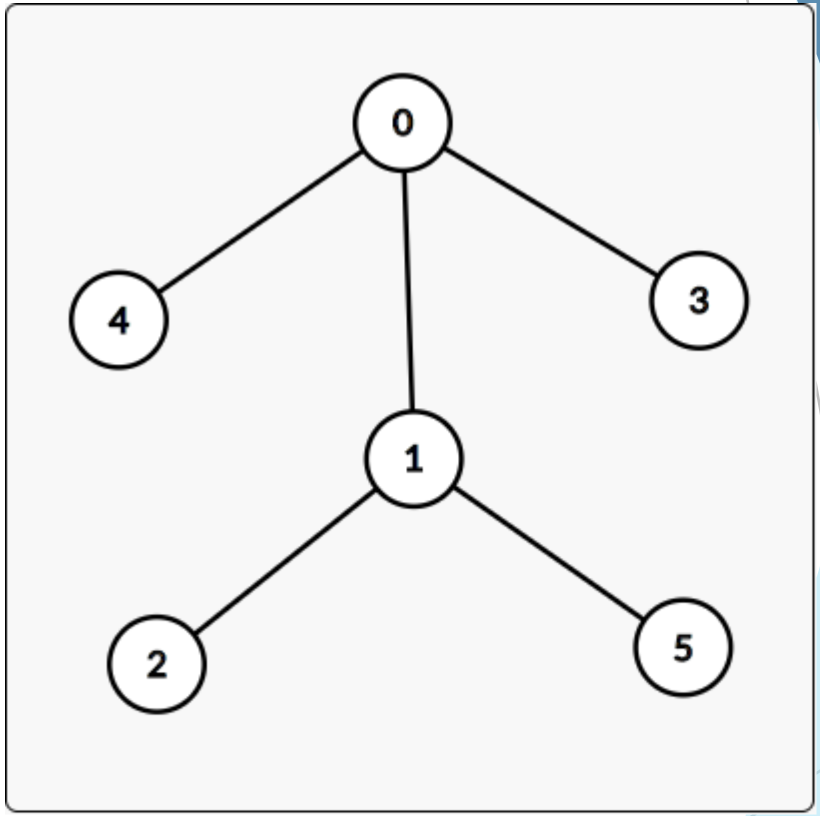
- 連結な無向木が与えられる。
- 以下の手順でこの木を一つの連結なパスにせよ

1. 好きなだけ頂点を追加する
2. ステップ1.で追加した頂点と、与えられた木の頂点の間に好きなだけ辺を結ぶ
3. 辺を好きなだけ削除する

- ステップ1.で追加する頂点の個数を最小化したい。
  目的を達成するために必要な最小の頂点数を求めよ。

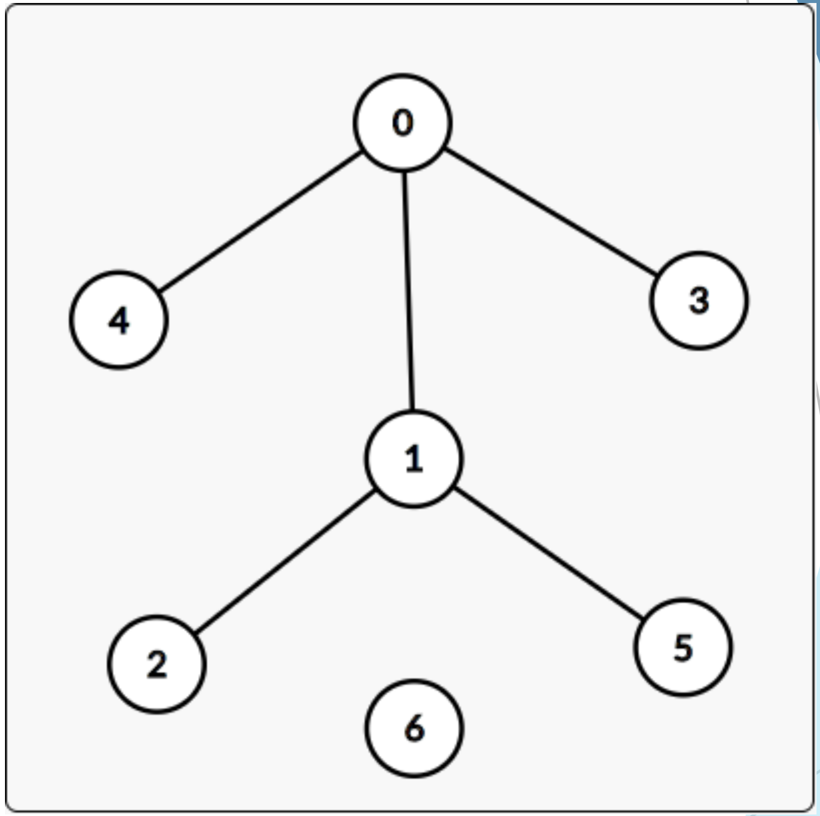
具体例

- 木があります



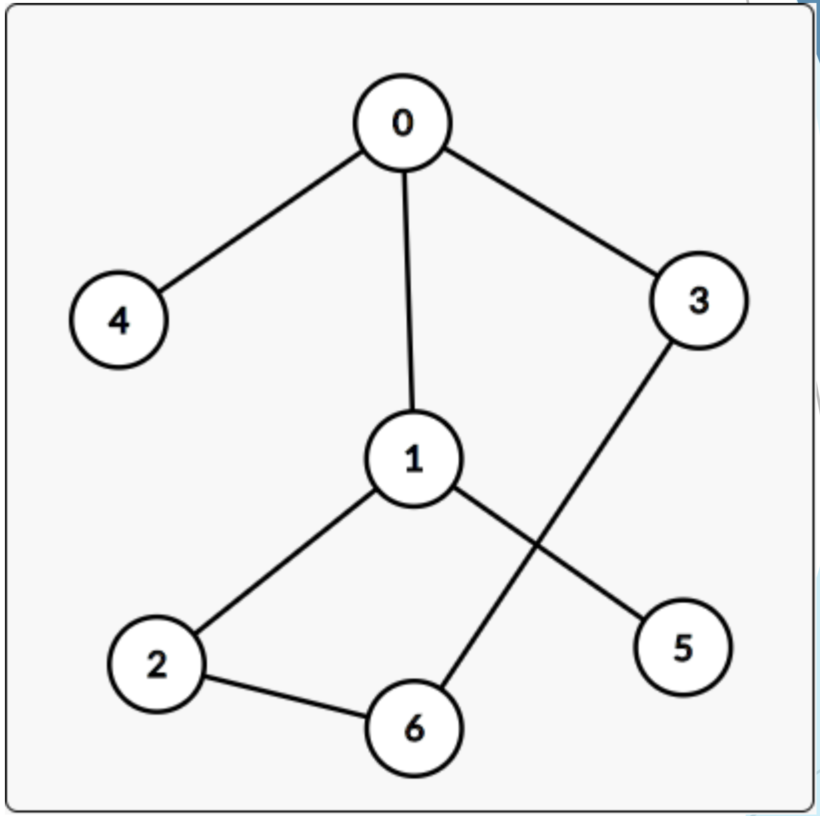
具体例

- 木があります
- 頂点を追加します



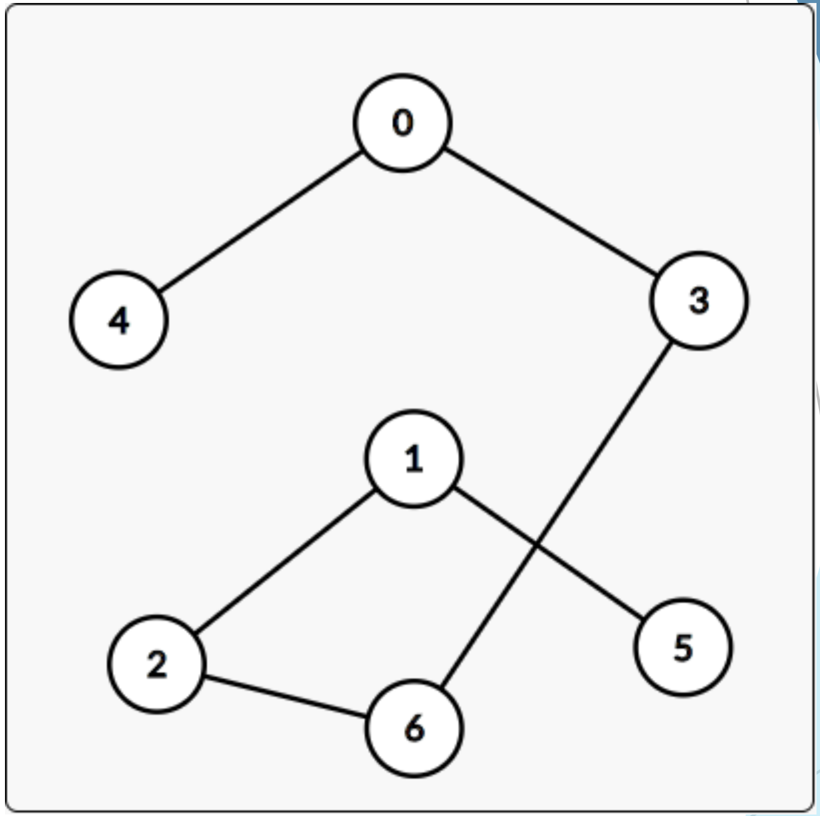
具体例

- 木があります
- 頂点を追加します
- 追加した頂点と木の頂点の間に辺を追加します



具体例

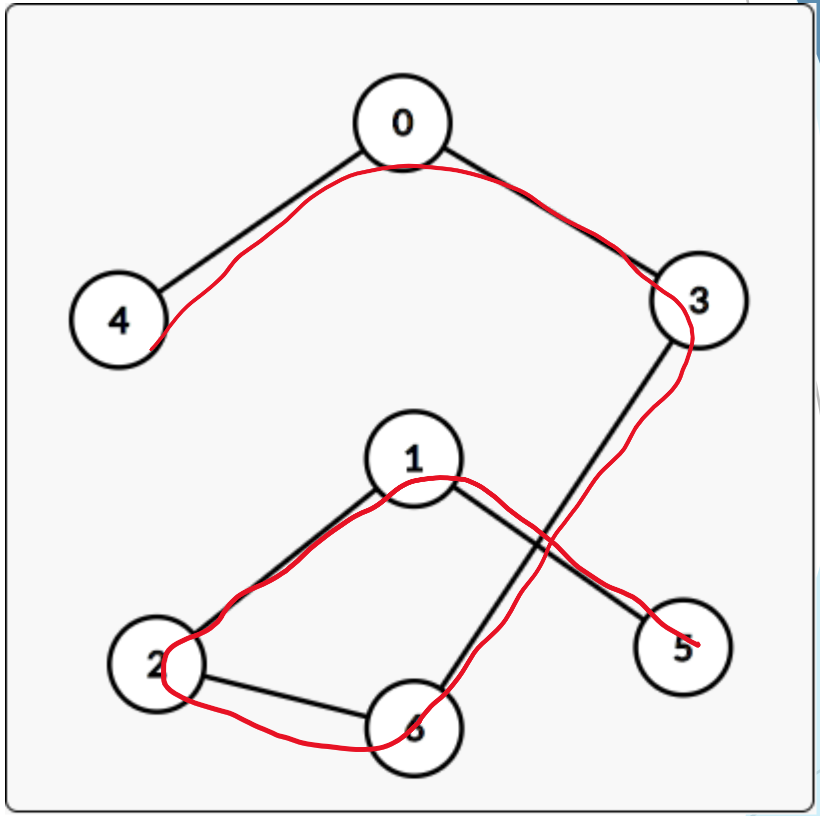
- 木があります
- 頂点を追加します
- 追加した頂点と木の頂点の間に辺を追加します
- 最後に、辺を削除しパスを作ります



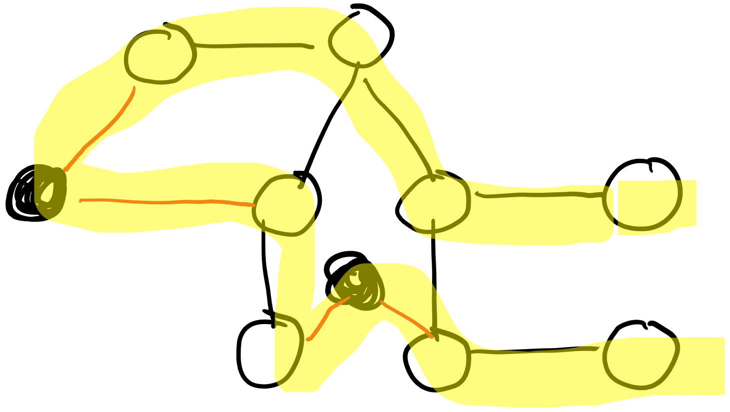
具体例

- 木があります
- 頂点を追加します
- 追加した頂点と木の頂点の間に辺を追加します
- 最後に、辺を削除しパスを作ります

- 今回は一つの頂点の追加でパスを作れたため、答えは1



考察



- 既存の頂点を◯、新しく追加された頂点を●で表現すると…
- 最終的にできるパスは◯ ◯… ◯ ◯ ● ◯ ◯… ◯ ◯ ● ◯ ◯… ◯ ◯ のように、M+1個の◯の塊の間にM個の●が挟まっている状態になる
(●が最終的なパスの端に来ても意味がないため)
- M+1個の◯の塊は元の木上ではパスである
- よって、
「木をM+1個のパスに分解できる⇔新しく加えるべき頂点数はM以下である」
- 木をパスに分解する問題に帰着できた！

木DP

- 木を最小で何個のパスに分解できるかを計算しよう
- 木DPを使う
- 木を頂点1を親とした根付き木に変換する
- dp(x,0):=頂点xを根とした部分木は最小で何個のパスに分解できるか？ただし、頂点xは分解したパスの端点
- dp(x,1):=頂点xを根とした部分木は最小で何個のパスに分解できるか？ただし、頂点xは分解したパスの端点**ではない**
- dp(x):=min{dp(x,0),dp(x,1)}
- 以上のようにdp式を定義する

漸化式(dp(x,0))

- dp(x,0)について
- 頂点xは分解したパスの端点でなければならない
  1. 頂点xのみを一つのパスとして見た場合、答えは1+Σdp(ch)
  2. 頂点xからある子ch'の部分木に向けてパスを伸ばす場合、答えは(Σdp(ch))-dp(ch')+dp(ch',0)
  - 1.はO(xの子の数)で計算できる。2.はdp(ch')-dp(ch',0)が最大のch'を採用すれば式全体が最小になるため、O(xの子の数)で計算できる。
- 以上より、漸化式は
- dp(x,0)=MIN{1+Σdp(ch), (Σdp(ch))-dp(ch')+dp(ch',0)}
(ch'はxの子の中でdp(ch')-dp(ch',0)が最大のもの)

漸化式(dp(x,1))

- dp(x,1)について、
- まず大前提として、xが子を1個以下しか持たない場合、どのように分解してもパスの端点になりえない
  - この場合は適当に例外値を返そう
- ある二つの子ch1,ch2について、ch1からxを経由してch2へ伸びていくパスを考えた場合、答えは(Σdp(ch))-dp(ch1)+dp(ch1,0)-dp(ch2)+dp(ch2,0)
- これも、dp(ch')-dp(ch',0)が最大の二つの頂点をch1,ch2に採用すれば式全体が最小になり、計算量はO(xの子の数)
- 以上より、漸化式は
- dp(x,1)=(Σdp(ch))-dp(ch1)+dp(ch1,0)-dp(ch2)+dp(ch2,0)
  (ch1,ch2はxの子の中でdp(ch')-dp(ch',0)が最大の二つ)
  (子が1個以下しかない場合は∞)

まとめ

- 以上の式を用いて、dp(1)を計算し、**それを-1したもの**が答えである。
- 各部分木について子供の数しか計算量がかからないため、
全体の計算量はO(N)である。

- 辺をパスに分解する問題とは対照的で、頂点をパスに分解する問題には
このような木DPが有効である。