

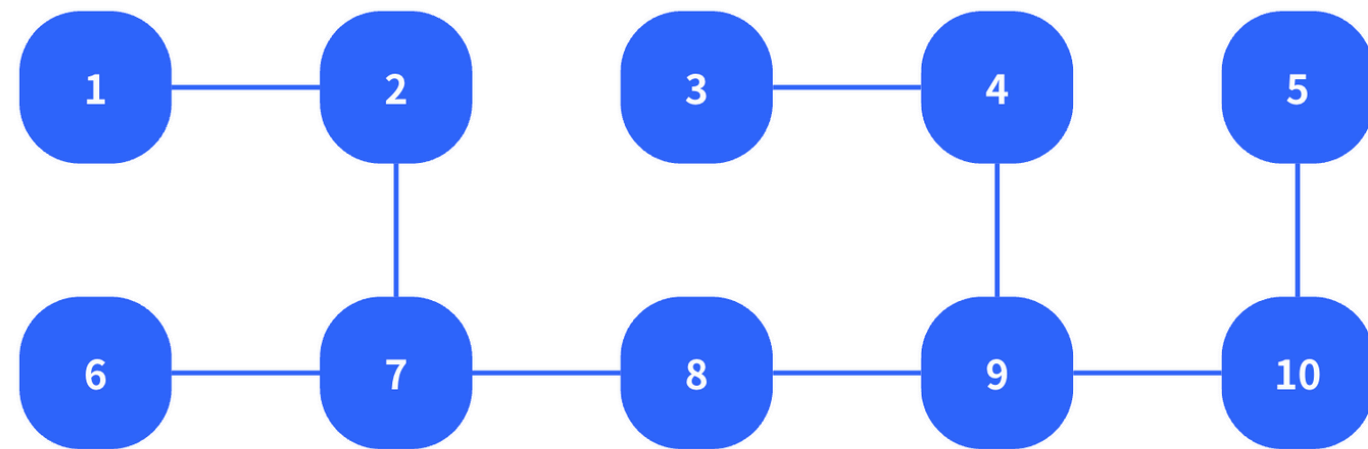
알고리즘 스터디 11회

최소 스패닝 트리를 만들어보자

CONTENTS

01	—————	최소 스패닝 트리란
02	—————	크루스칼 알고리즘 로직
03	—————	크루스칼 알고리즘 시뮬
04	—————	프림 알고리즘 로직
05	—————	프림 알고리즘 시뮬
06	—————	프림 vs 크루스칼
07	—————	간단한 예제

스패닝 트리란



스패닝 트리란

사이클이 존재하지 않고 모든 노드가 연결되어 있는 트리를 스패닝 트리라고 합니다.

- ✓ 간선의 개수 = 노드의 개수 - 1
- ✓ 하나의 집합으로 구성
- ✓ 최소 스패닝 트리 : 간선의 가중치 합이 최소

02

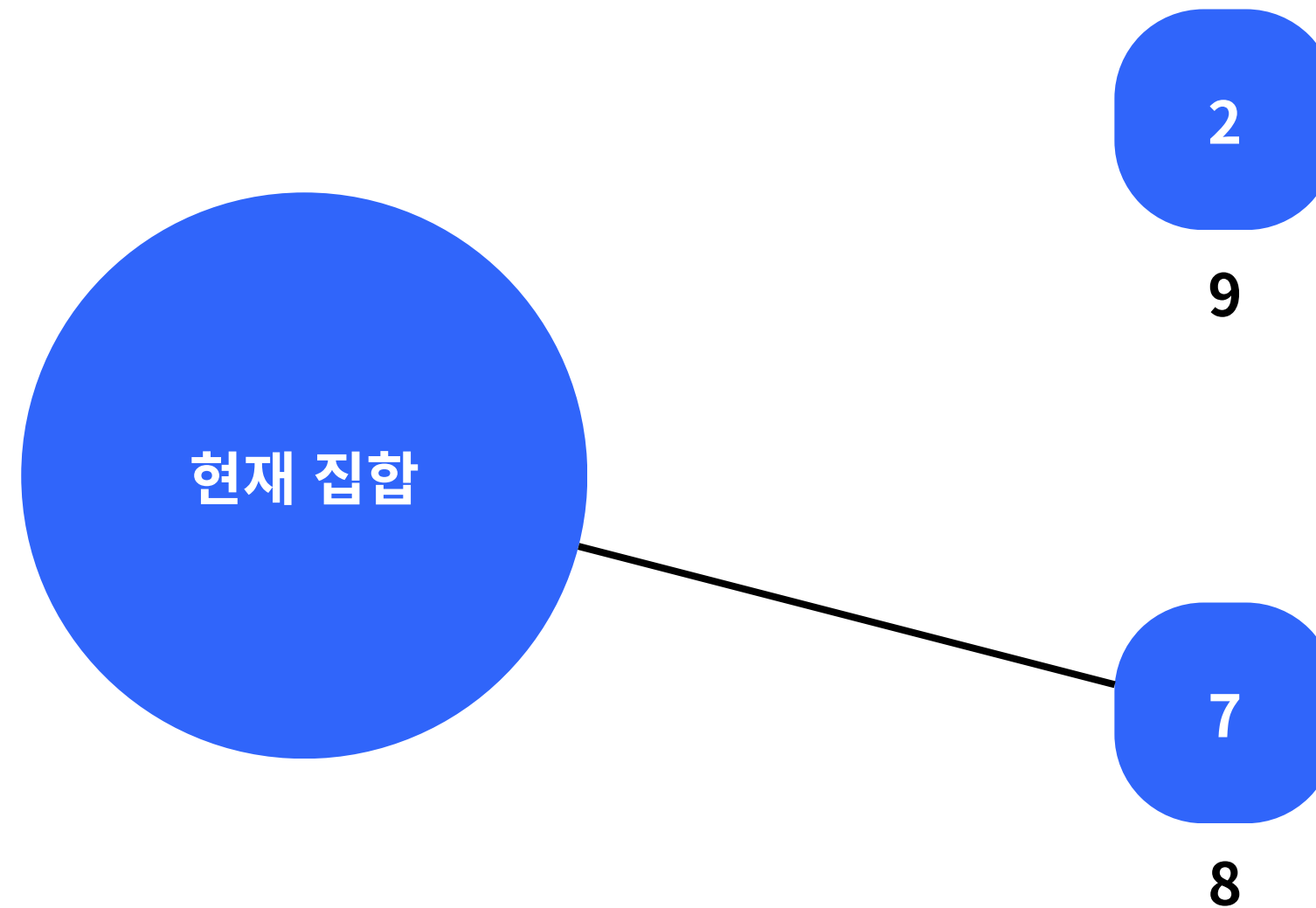
프림 알고리즘 로직



현재 집합에서 가장 가까운 노드부터 연결

02

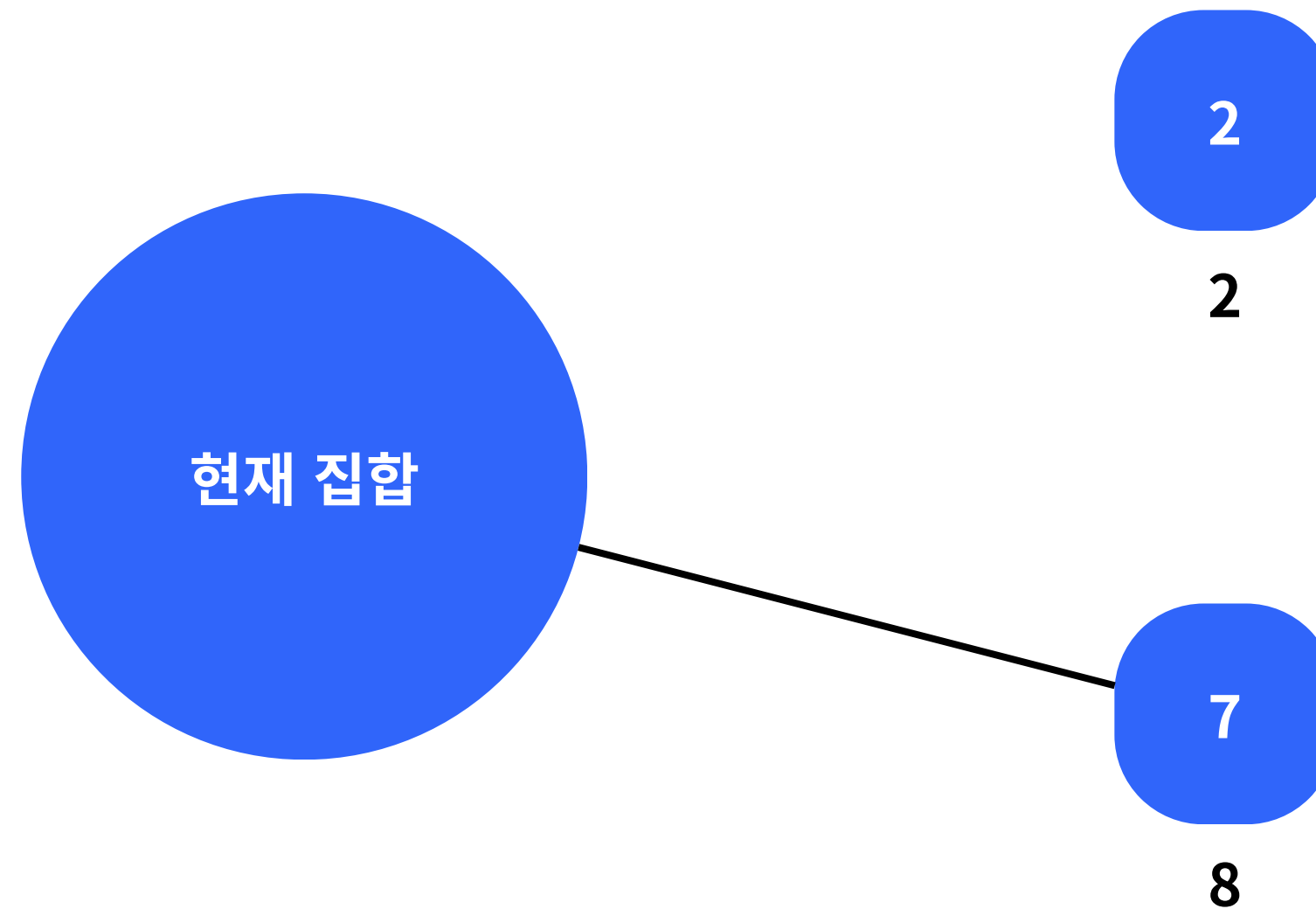
프림 알고리즘 로직



현재 집합에서 가장 가까운 노드부터 연결

02

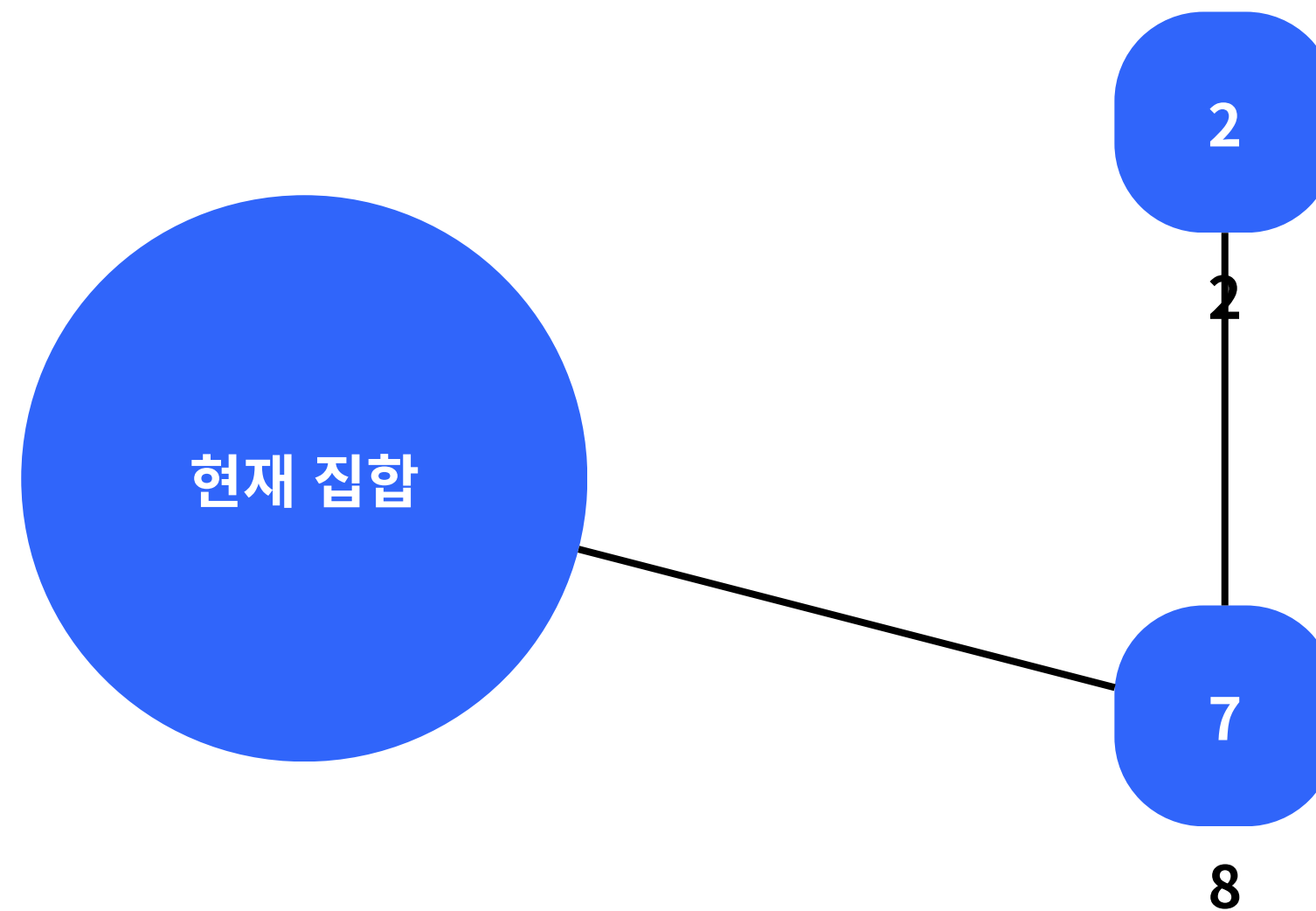
프림 알고리즘 로직



새로운 노드가 추가되면 정보 업데이트

02

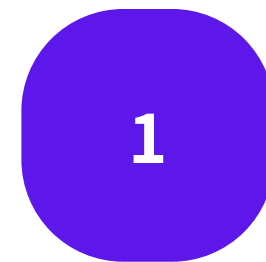
프림 알고리즘 로직



최소 거리 노드 이어주기

03

프림 알고리즘 시뮬



9

8



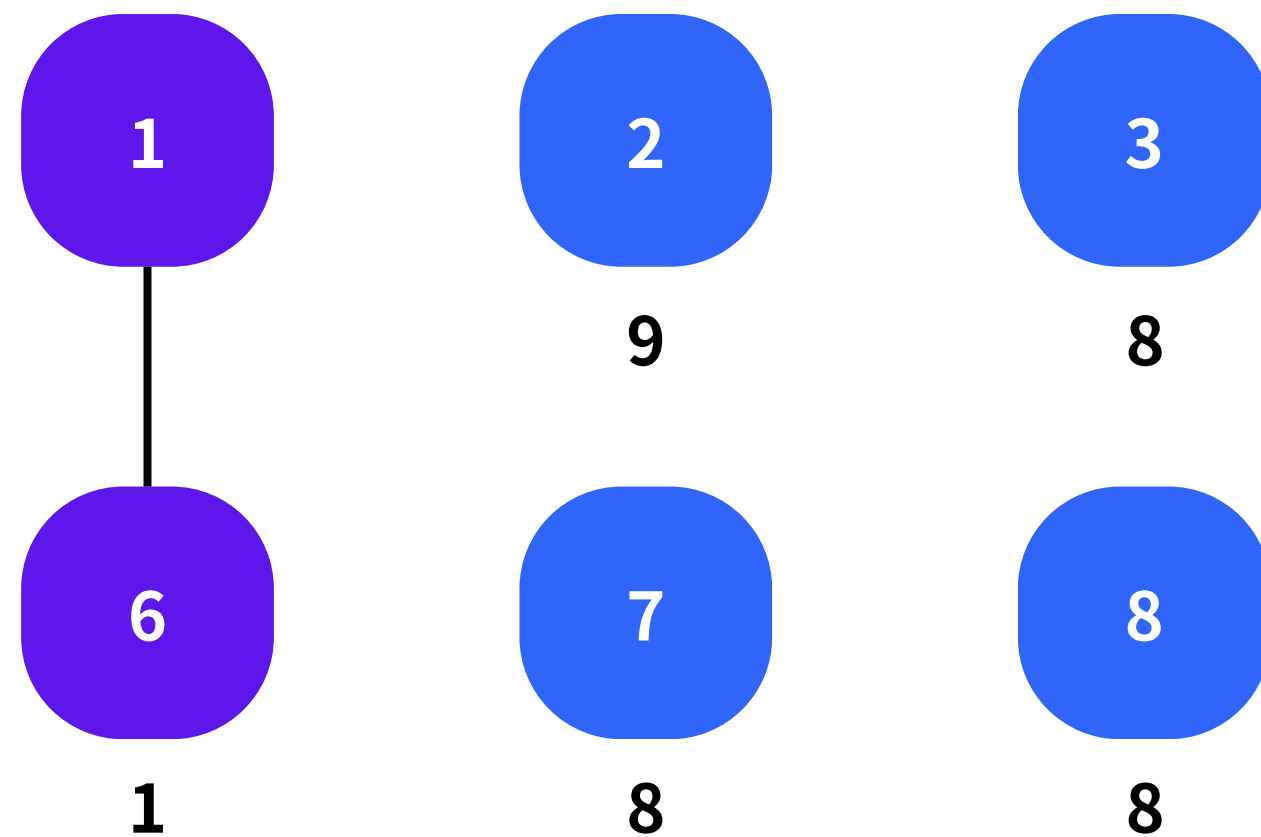
1

8

8

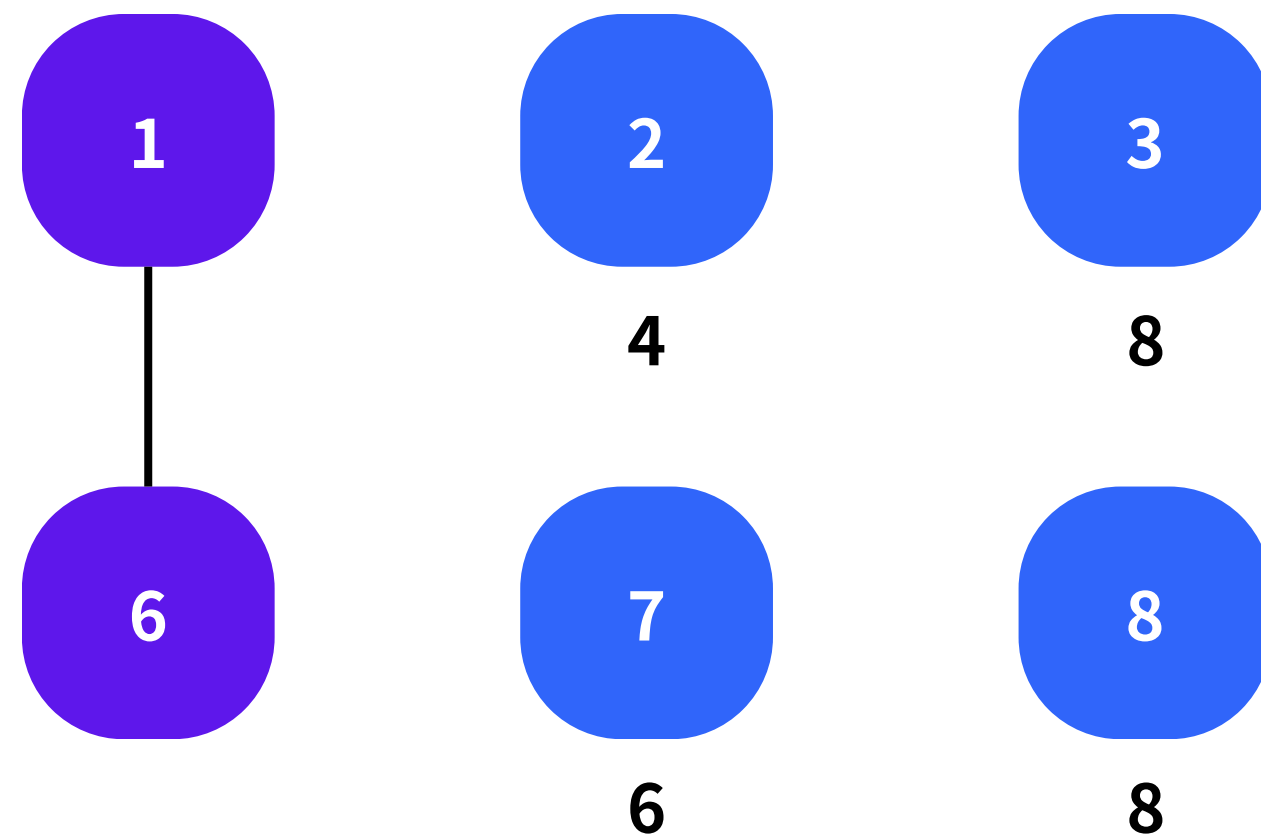
03

프림 알고리즘 시뮬



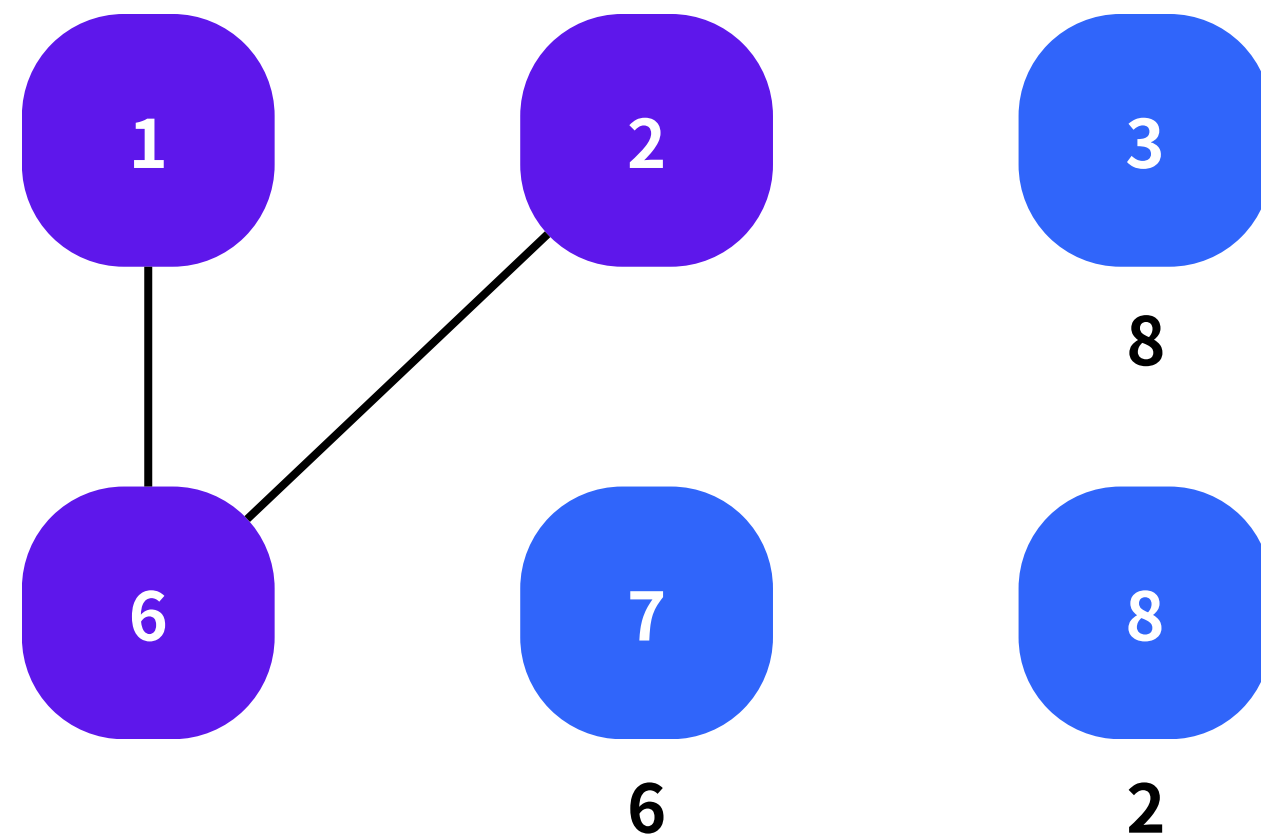
03

프림 알고리즘 시뮬



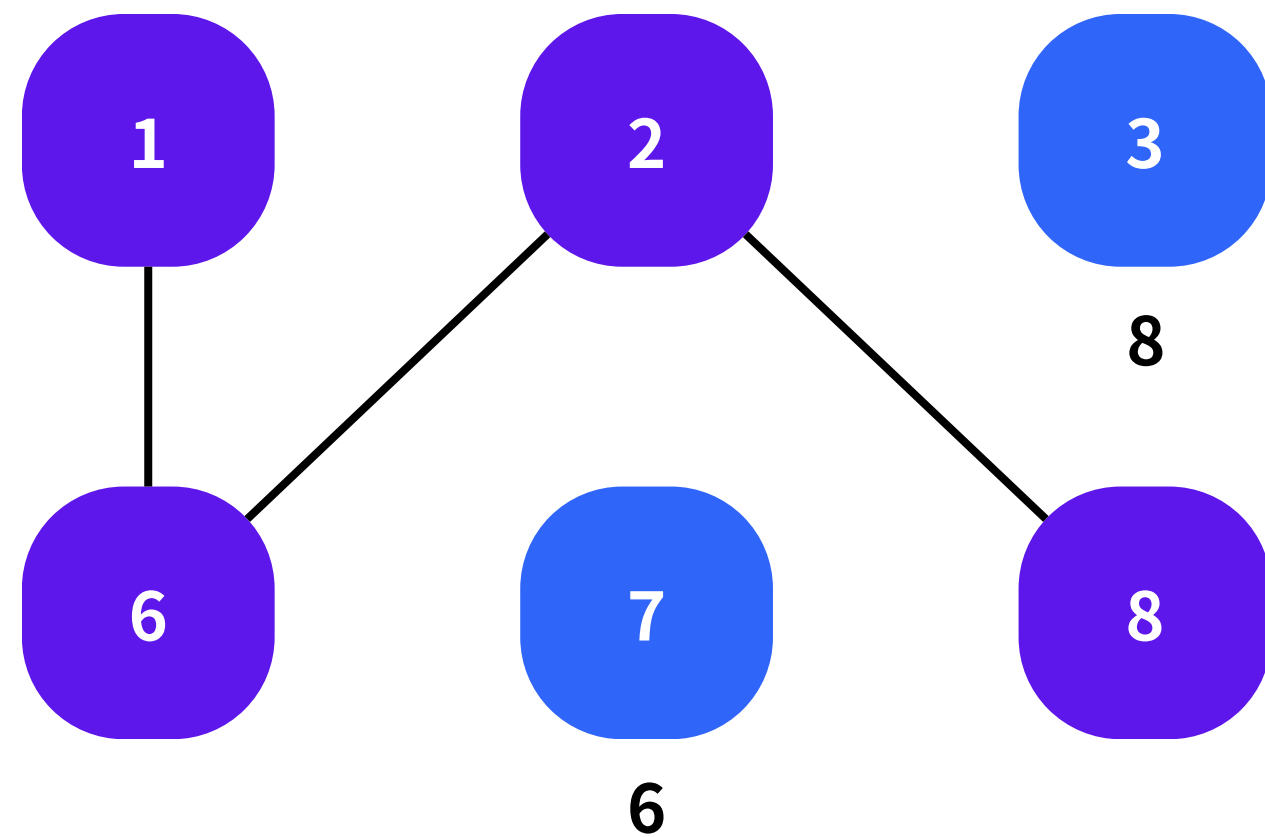
03

프림 알고리즘 시뮬



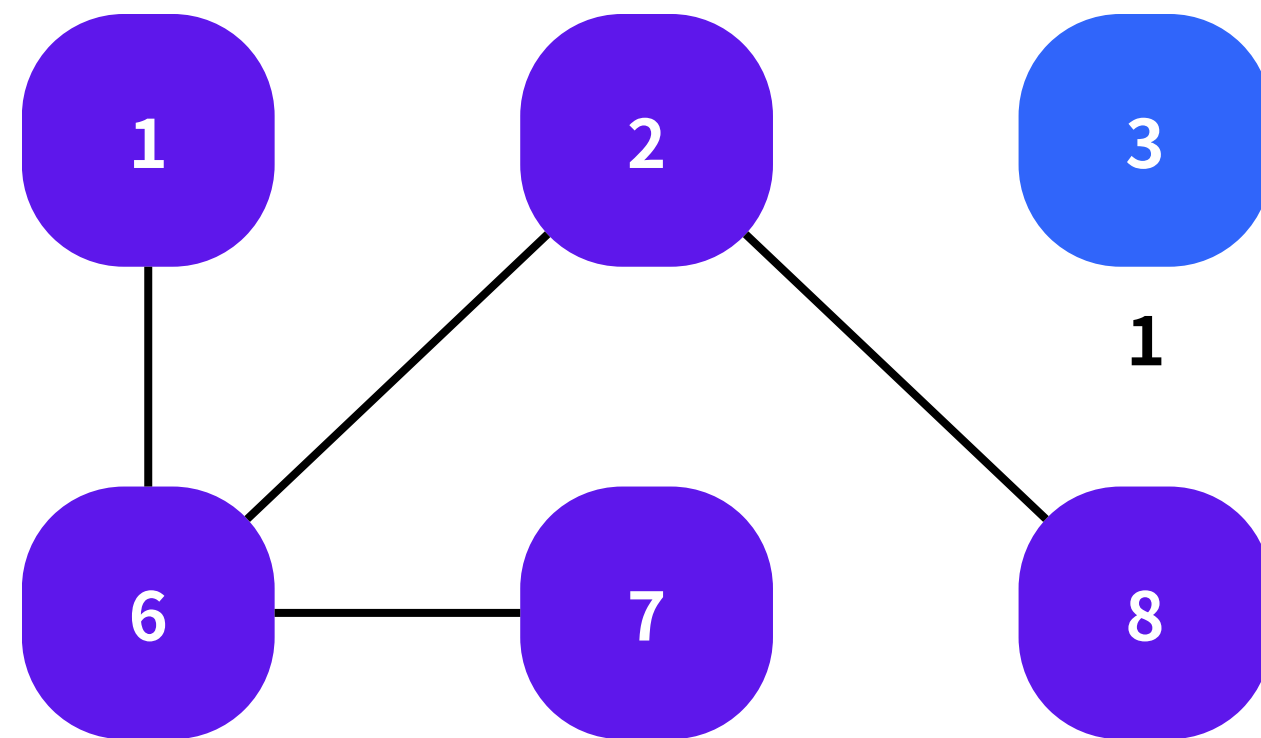
03

프림 알고리즘 시뮬



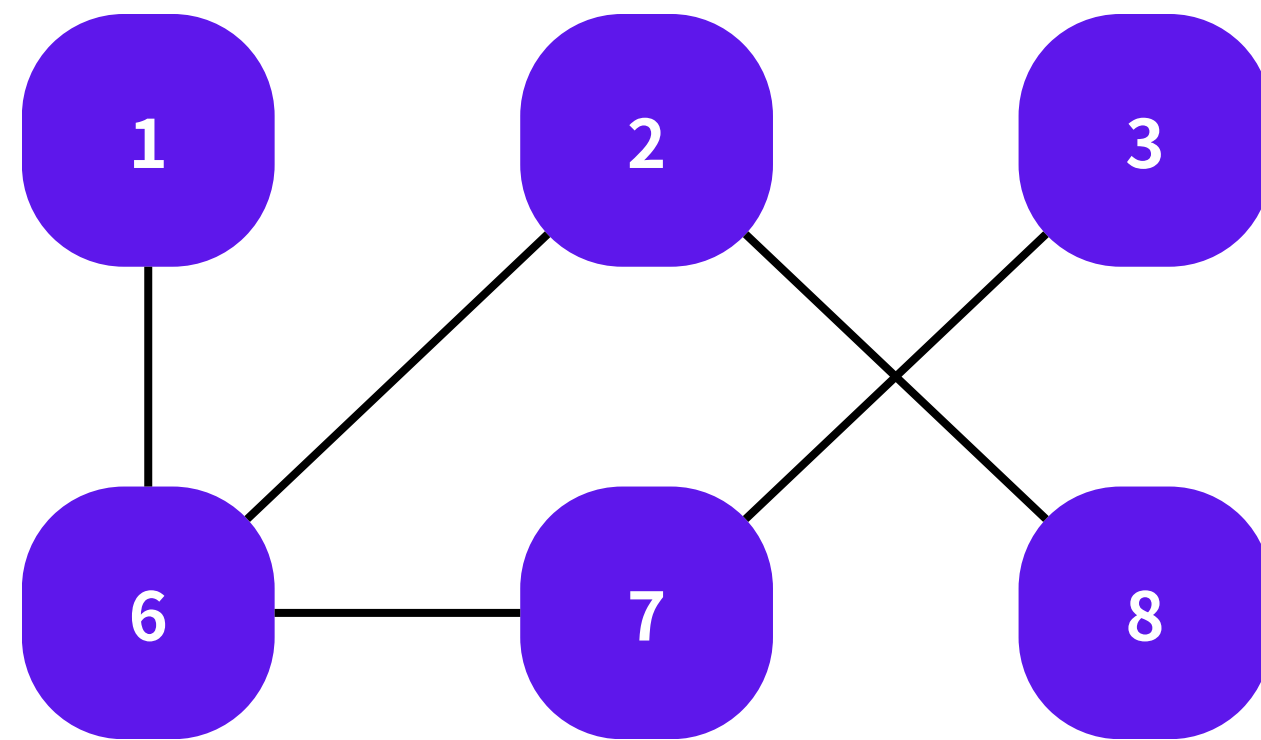
03

프림 알고리즘 시뮬



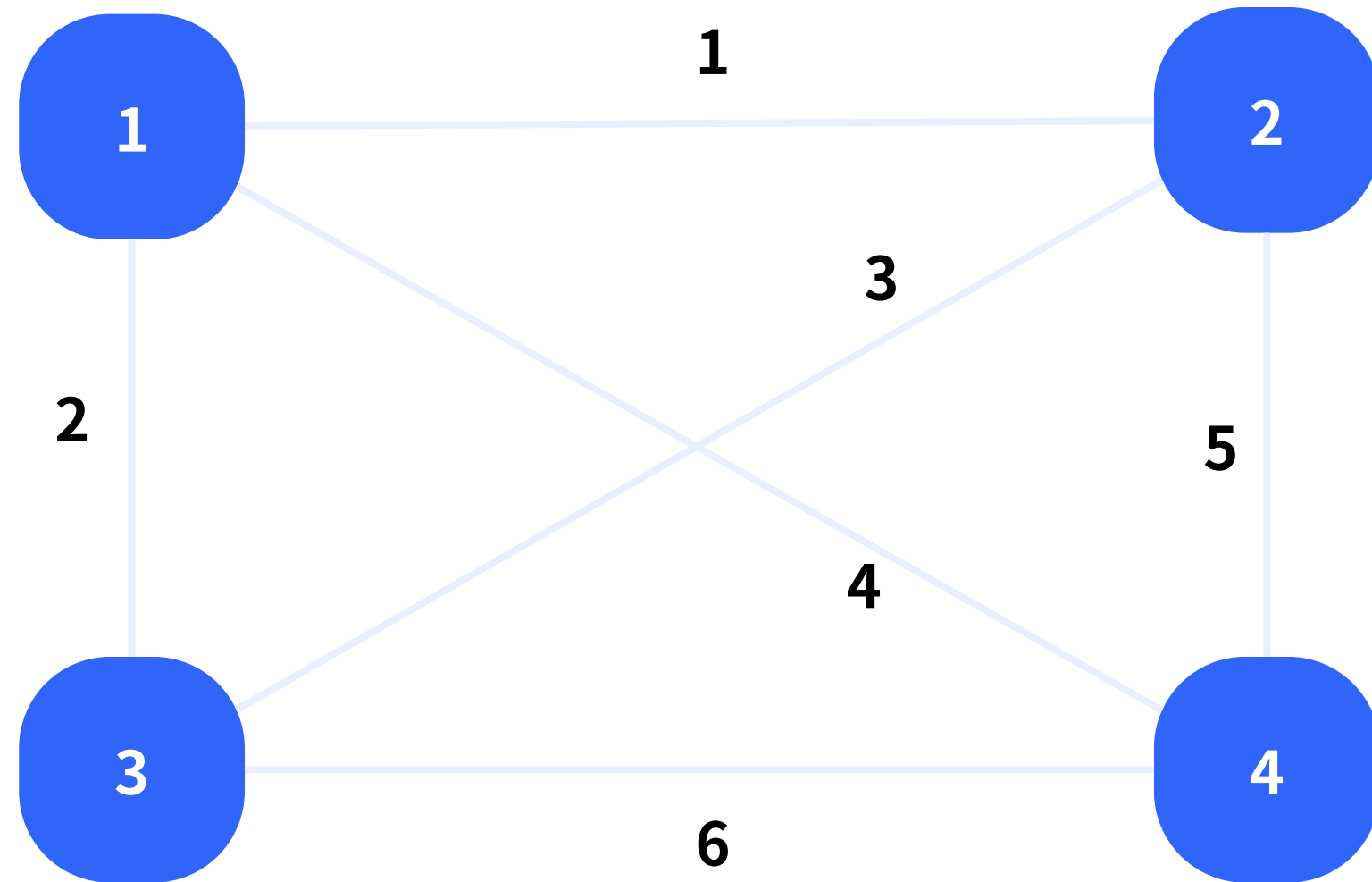
03

프림 알고리즘 시뮬



04

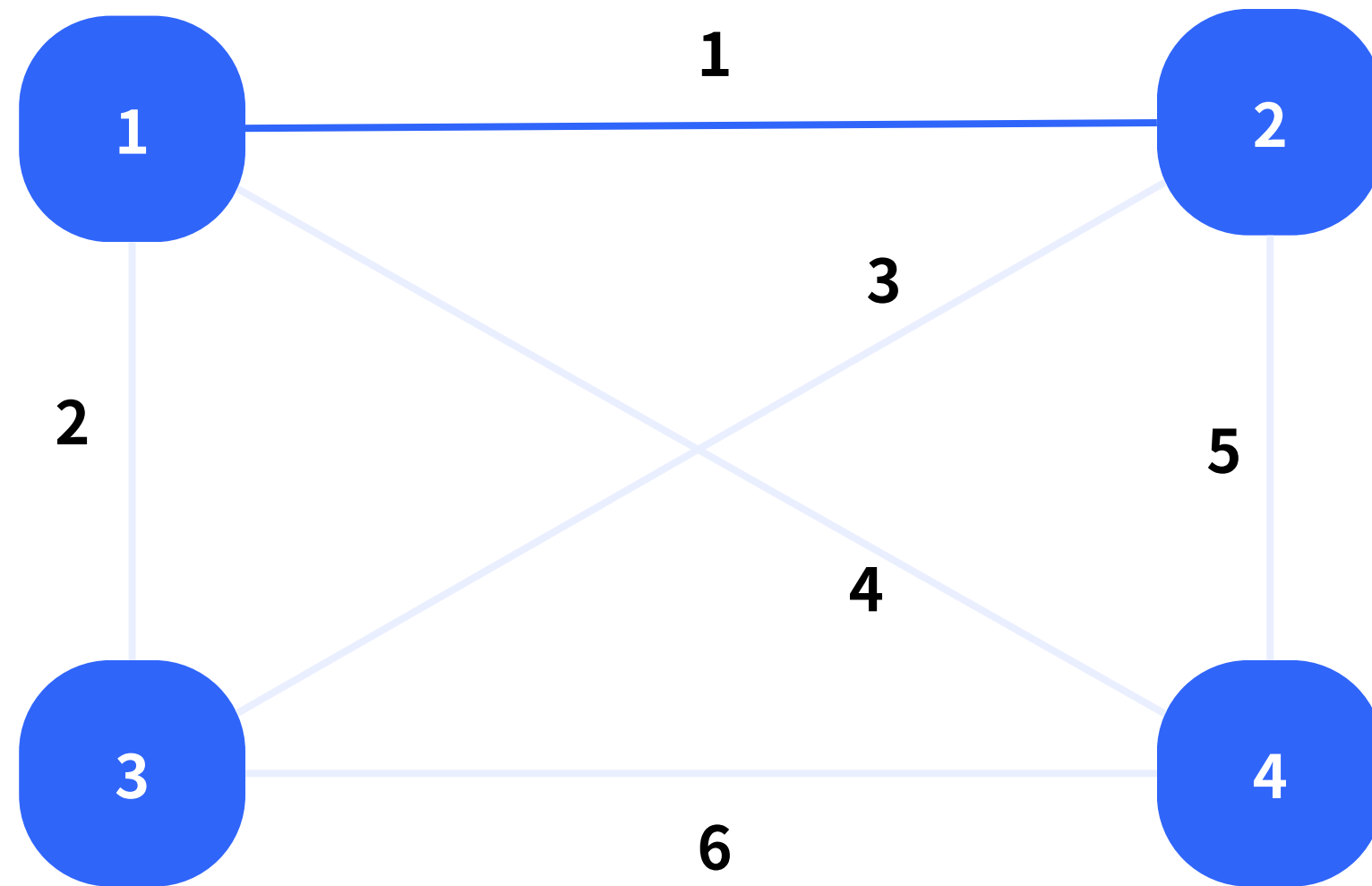
크루스칼 알고리즘 로직



가장 가중치가 낮은 간선부터 추가

05

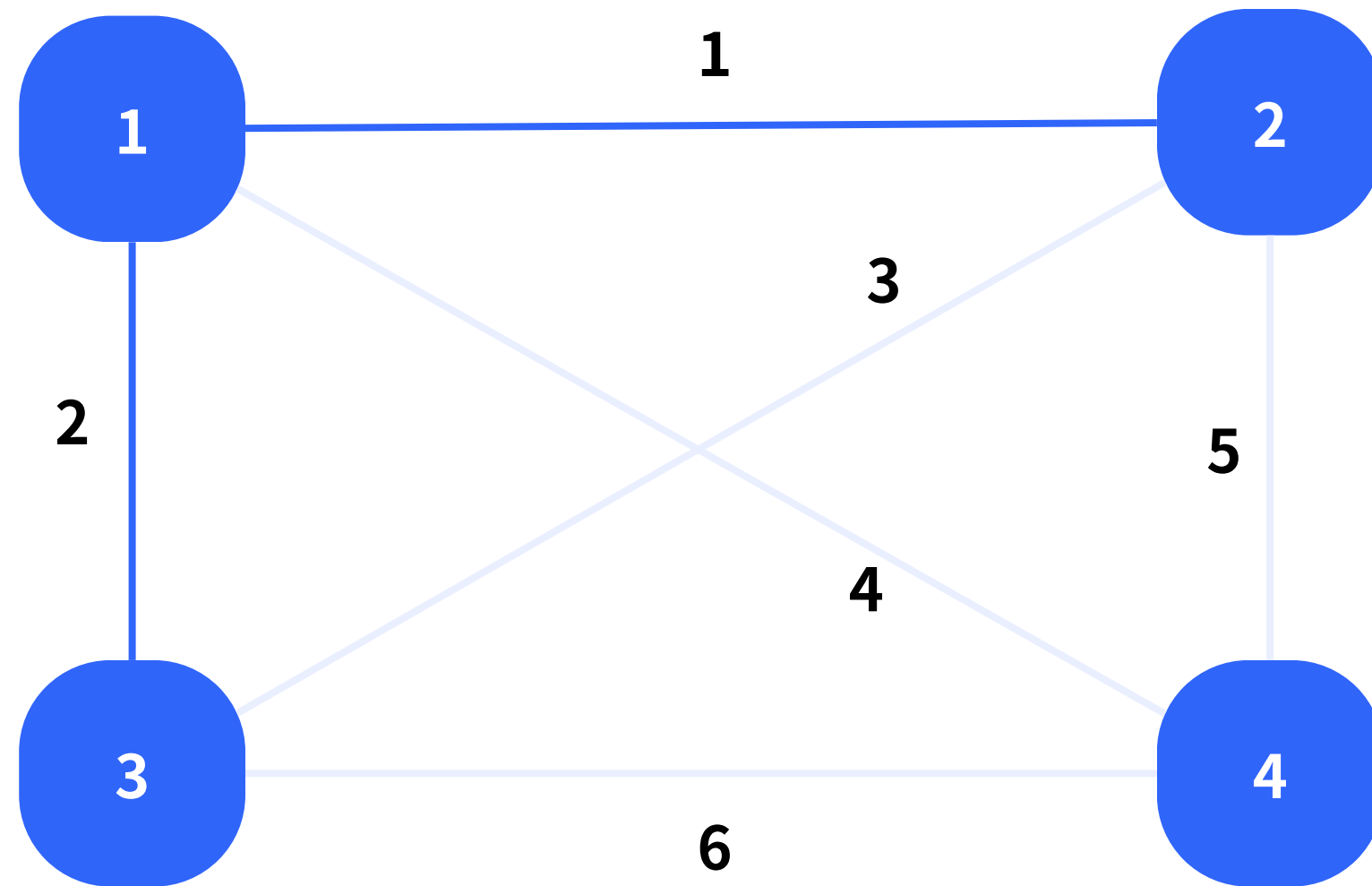
크루스칼 알고리즘 시뮬



가장 작은 1 가중치를 가진 1 추가

05

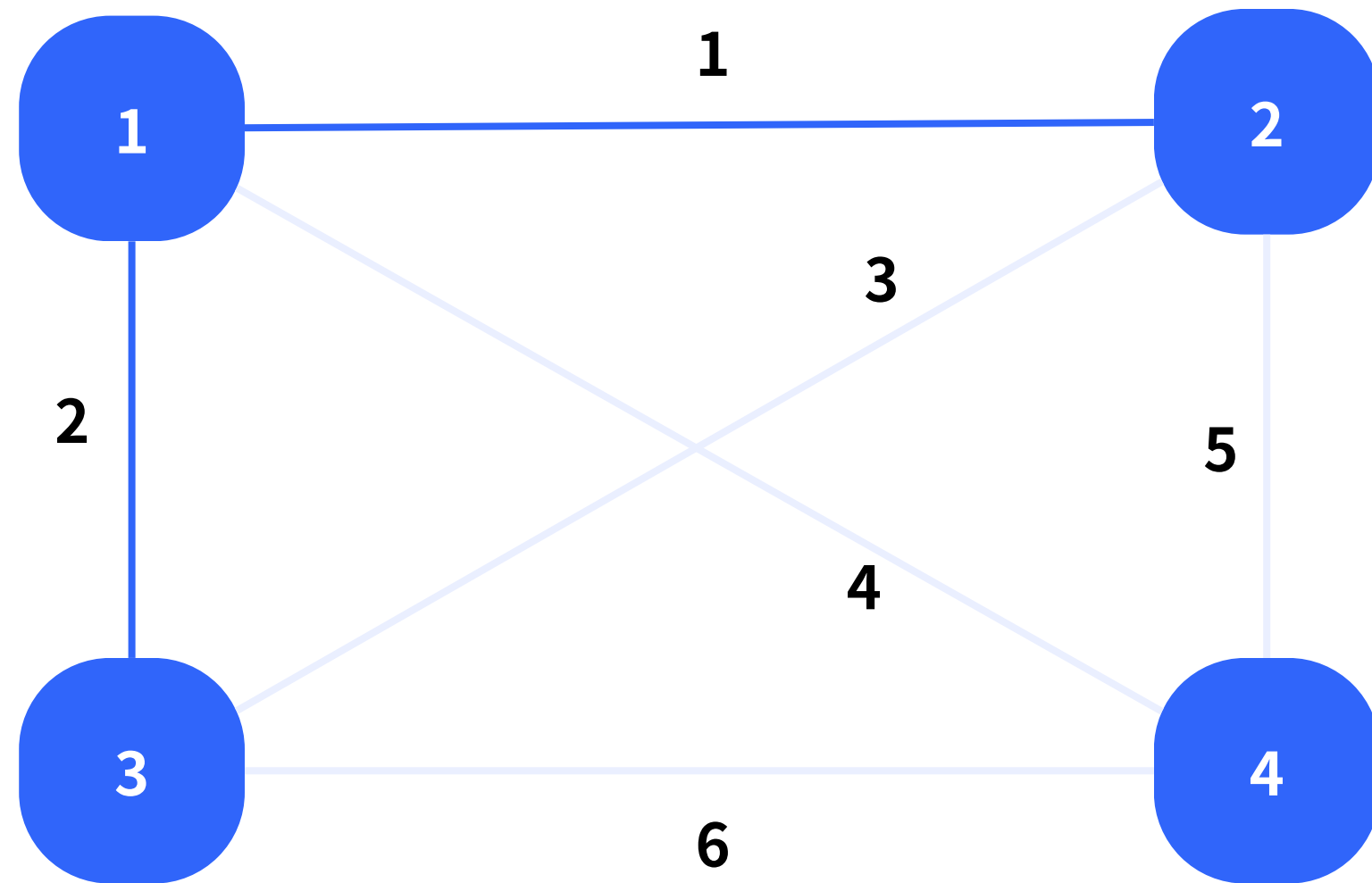
크루스칼 알고리즘 시뮬



그 다음 작은 2 가중치를 가진 2 추가

05

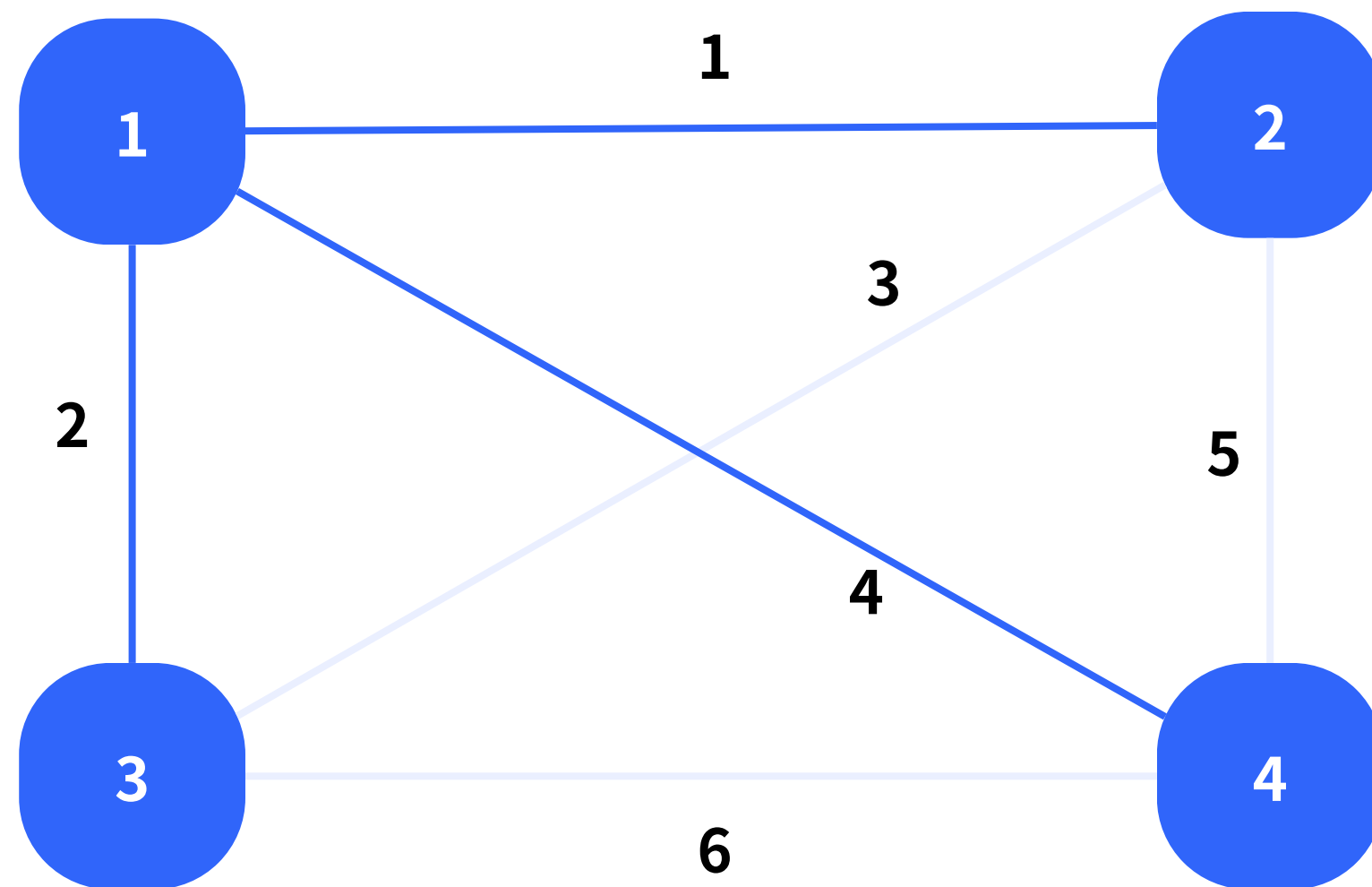
크루스칼 알고리즘 시뮬



3의 경우 이미 같은 그룹이라 넣지 않음

05

크루스칼 알고리즘 시뮬



4 추가

06

프림 vs 크루스칼

	프림	크루스칼
유니온 파인드	안써도됨	써야함
시간 복잡도	$O(E \log V + V \log V)$	$O(E \log E)$
유리한 그래프	Dense Graph (간선의 개수가 많은 그래프)	Sparse Graph (간선의 개수가 적은 그래프)

07

간단한 예제

boj.ma/1045

도로 성공



1 골드 I

시간 제한	메모리 제한	제출	정답	맞힌 사람	정답 비율
2 초	128 MB	2139	665	517	31.258%

문제

0부터 $N-1$ 까지의 번호가 매겨져 있는 N 개의 도시와 두 도시를 연결하는 도로가 있다. 도로에는 우선순위가 있는데, A 와 B 가 ($A < B$) 도로 x 로 연결되어 있고, C 와 D 가 ($C < D$) 도로 y 로 연결되어 있을 때, 튜플 $(A, B) < (C, D)$ 이면 $x > y$, 즉 x 의 우선순위가 더 높다. 여기서 $a_i \neq b_i$ 인 가장 작은 양의 정수 i 에 대해 $a_i < b_i$ 이면 $(a_1, \dots, a_k) < (b_1, \dots, b_k)$ 로 정의한다.

도로의 집합은 하나 이상의 도로가 우선순위에 대한 내림차순으로 정렬되어 있는 것이다. 집합 사이에도 우선순위가 있는데, 두 집합을 튜플로 나타냈을 때의 우선순위를 따른다. 한 집합에 있는 도로만으로 임의의 도시에서 임의의 도시로 이동할 수 있을 때, 그 집합은 연결되어 있다고 한다.

김지민이 할 일은 M 개의 도로를 가진 도로의 집합 중 연결되어 있으면서 우선 순위가 가장 높은 것을 찾는 것이다.

입력

첫째 줄에 도시의 개수 N 과 M 이 주어진다. N 은 50보다 작거나 같은 자연수이고, M 은 $N-1$ 보다 크거나 같고, 1,000보다 작거나 같은 정수이다. 둘째 줄부터 N 개의 줄에는 인접행렬이 주어진다. 즉 i 번째 행의 j 번째 열이 Y 이면 도시 i 와 j 를 연결하는 도로가 존재하고, N 이면 존재하지 않는다. i 와 j 가 연결되어 있으면 j 와 i 도 연결되어 있음이 보장되고, i 와 i 는 연결되어 있지 않다.

출력

만약 정답이 없을 때는 -1을 출력한다. 정답이 존재하면, 그 집합에 속하는 도로 중 0을 끝점으로 갖는 도로의 개수, 1을 끝점으로 갖는 도로의 개수, ..., $N-1$ 을 끝점으로 갖는 도로의 개수를 차례로 출력한다.