

게임 개발

<https://www.acmicpc.net/problem/1516>

- 작업 문제(<https://www.acmicpc.net/problem/2056>) 문제와 동일하게 풀면 된다.
- Queue를 이용한 위상정렬 문제에 해당
- 현재 건물을 짓기 전에 지어야 하는 건물들 중 짓는 데 가장 오래 걸리는 건물의 시간이 현재 건물을 지을 수 있는 가장 빠른 시간이 된다.
- C++ 코드

```
while (!q.empty()) {
    int u = q.front();
    q.pop();

    for (int v : AdjList[u]) {
        ind[v] -= 1;
        cost[v] = max(cost[v], cost[u] + work[v]);
        if (ind[v] == 0) {
            q.push(v);
        }
    }
}

queue<int> q;
for (int i = 1; i <= N; ++i) {
    if (ind[i] == 0) {
        q.push(i);
        cost[i] = work[i];
    }
}
```

```
while (!q.empty()) {
    int u = q.front();
    q.pop();

    for (int v : AdjList[u]) {
        ind[v] -= 1;
        cost[v] = max(cost[v], cost[u] + work[v]);
        if (ind[v] == 0) {
            q.push(v);
        }
    }
}
```

게임 개발

<https://www.acmicpc.net/problem/1516>

- Python 코드

```
q = queue.Queue()
for i in range(1, len(cost)):
    if ind[i] == 0:
        q.put(i); cost[i] = work[i]

while not q.empty():
    u = q.get()
    for v in AdjList[u]:
        ind[v] -= 1
        cost[v] = max(cost[v], cost[u] + work[v])
        q.put(v) if ind[v] == 0 else None

for i in range(1, len(cost)):
    print(cost[i])
```
