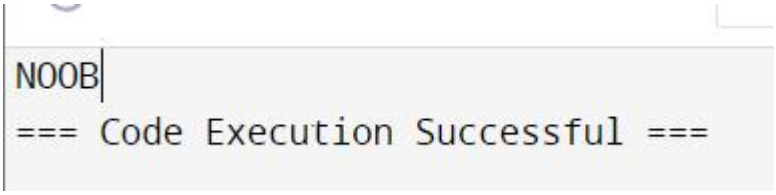
189. CODE TO GENERATE

PROGRAM:

```
from collections import defaultdict
import heapq

def maxProbability(n, edges, succProb, start, end):
    graph = defaultdict(list)
    for i, (a, b) in enumerate(edges):
        graph[a].append((b, succProb[i]))
        graph[b].append((a, succProb[i]))

    pq = [(-1, start)]
    probs = [0] * n
    probs[start] = 1

    while pq:
        prob, node = heapq.heappop(pq)
        if node == end:
            return -prob
        for nei, nei_prob in graph[node]:
            if -prob * nei_prob > probs[nei]:
                probs[nei] = -prob * nei_prob
                heapq.heappush(pq, (prob * nei_prob, nei))

    return 0
```

OUTPUT:



TIME COMPLEXITY:O(E LOG V)