算法 2 DFS算法

Input: 序列s_i, 成本$C_{lb}(s_i)$, 最优路线s_i^*, 最优成本$C_{lb}(s_i^*)$, 当前概率p, 节点集合$\bar{J}$

if s_i已达到最大深度 & $C_{lb}(s_i) < C_{lb}(s_i^*)$ **then**
  $s_i^*, C_{lb}(s_i^*) :=$ 更新$(s_i, C_{lb}(s_i), s_i^*, C_{lb}(s_i^*))$
  return $s_i^*, C_{lb}(s_i^*)$
else
  for $j \in \bar{J}$ **do**
    stack := 记录$(s_i, C_{lb}(s_i), p)$
    if $C_{lb}(s_i + \{j\}) < C_{lb}(s_i^*)$ **then** % 剪枝
      $s_i := s_i + \{j\}$
      $\bar{J} := \bar{J} \backslash \{j\}$
      $p := p * q_j$
      $s_i^*, C_{lb}(s_i^*) :=$ DFS$(s_i, C_{lb}(s_i), s^*, C_{lb}(s^*), p, \bar{J})$ % 递归s_i,
      $C_{lb}(s_i), p :=$ 恢复(stack)
    end if
  end for
end if
return $s_i^*, C_{lb}(s_i^*)$