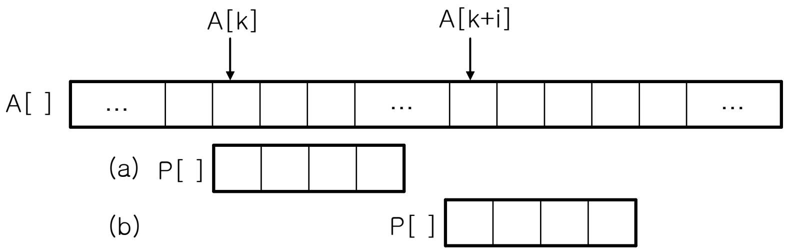
기말고사, Computer Algorithms

2007. 12. 3

Open Book, 65분

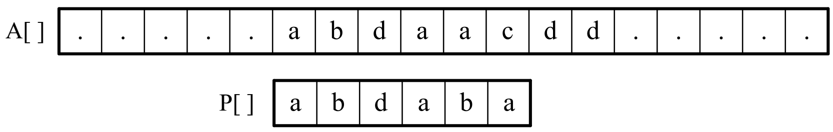
총 6 문제, 3 pages

1. (15점, 답만 쓰면 됨) 현재 텍스트와 패턴의 비교상황이 (a) 다음에 바로 (b)의 상태로 변했으면 i칸 점프했다고 말한다.



아래 그림은 패턴 “abdaba"가 텍스트 A[ ]에 있는지를 찾는 과정의 중간 상황이다. 방금 그림의 자리에서 패턴과 관련된 일을 ‘마치고’ 다음의 상태로 넘어가려고 하는 상황이다.

1.1 만일 수업시간에 배운 KMP algorithm이라면 몇칸을 점프하겠는가?
1.2 만일 수업시간에 배운 Boyer-Moore-Horspool algorithm이라면 몇칸을 점프하겠는가?



2. (15점) 문교수는 알고리즘 책을 심혈을 기울어 집필한 다음 에러없는 책을 썼다고 기분이 좋았다. 그렇지만 얼마 후부터 에러가 여기저기 보이기 시작했다. 책에 아래와 같이 더블해싱 함수의 예를 제공했는데 잠재적인 문제가 발견되었다.

$$h(x) = x \bmod 13,\ f(x) = x \bmod 11,\ h_i(x) = (h(x) + i f(x)) \bmod 13$$

더블해싱을 이렇게 했을 때 어떤 심각한 문제가 발생할 수 있는가?

3. (15점) 아래는 수업시간에 배운 DAG의 최단경로 알고리즘을 조금만 바꾸어본 것이다. 이렇게 해도 최단경로가 제대로 구해지는가? 대답이 No이면 반례를 제시하고, Yes이면 최단경로를 제대로 구한다는 것을 귀납적으로 증명하라.

```
DAG-ShortestPath(G, r)
{
    for each u∈V
            d_u←∞;
    d_r←0;
    G의 정점들을 topological sorting한다
    for each u∈V (topological order로)
            for each v∈To(u) ▹ To(u): 정점 u로 들어오는 간선이 있는 vertex들의 집합
                    if (d_v + w_{v,u} < d_u) then d_u←d_v + w_{v,u};
}
```

4. (각 10점) 최단경로 알고리즘에 대한 다음 물음에 답하라.
   4.1 Dijstra algorithm으로는 풀 수 없지만 Bellman-Ford algorithm으로는 풀 수 있는 문제의 예를 하나 제시하라. 단, vertex는 4개만 가진 예를 만들도록 하라.
   4.2 아래 Bellman-Ford algorithm이 제대로 작동을 한다는 것을 수학적 귀납법으로 증명하라.수학적 귀납법을 구성하는 초기조건, 귀납적 가정과 claim, claim의 증명이 포함되어야 한다. (Hint: ①행과 관련된 상황을 중심으로 생각하라.)

```
BellmanFord(G, r)
{
        for each u∈V
                d_u← ∞;
        d_r ← 0;
        for i ← 1 to |V|-1    ------------------------------ ①
                for each (u, v) ∈E
                        if(d_u + w_{u,v} < d_v) then d_v ← d_u + w_{u,v} ;
}
```

5. (20점) HAM-CYCLE이 NP-Complete임을 증명하는 방법은 다양하다. 아래 문제는 NP-Complete이다. 이로부터 HAM-CYCLE이 NP-Complete임을 증명하라.

HAM-ONE-EDGE

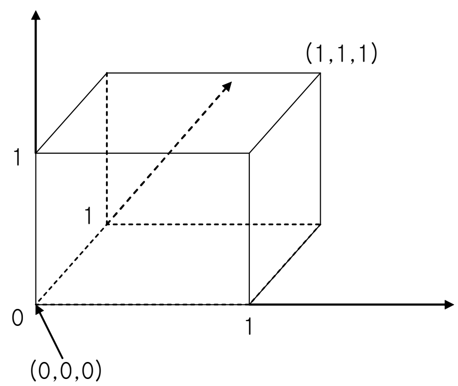
입력: $G=(V,E)$, $e \in E$

질문: e를 포함하는 Hamiltonian cycle이 존재하는가?

6. (15점) Gray coding은 수를 binary number로 표시하는 한 체계로서 이웃하는 수는 단 한 비트 만 차이가 나도록 설계된 코드를 말한다. 여기서 맨 처음수와 맨 끝수도 이웃한다고 정의한다. 아래는 0~15의 수를 Gray coding으로 표현한 예와 각각에 대응되는 일반 binary code를 나타낸 것이다. 여기서 소개한 Gray coding 이외에도 다른 Gray coding이 매우 많이 존재한다.

수	일반 binary code	Gray code
0	0000	0000
1	0001	0001
2	0010	0011
3	0011	0010
4	0100	0110
5	0101	0100
6	0110	0101
7	0111	0111
8	1000	1111
9	1001	1110
10	1010	1100
11	1011	1101
12	1100	1001
13	1101	1011
14	1110	1010
15	1111	1000

0~7을 3자리 Gray coding으로 나타내는 방법의 총수는 몇가지인가? 문제를 풀되 단순히 경우의 수를 따지는 것을 요구하지 않는다. 반드시 아래의 그림을 사용하고 Hamiltonian cycle과 연관을 시켜서 설명하라. 총 경우의 수가 제시되지 않거나 틀리더라도 Gray code와 이 그림, Hamiltonian cycle의 연관관계를 정확히 묘사하면 반 이상 점수를 줌.



(보너스 5점: 4자리 Gray coding의 총 수를 위의 방법을 확장하여 구하려면 어떻게 생각하면 되는지 기술하라. 그림을 정확히 그리고 방법을 정확히 기술하기만 하면 됨.)