알고리즘_HW07_결과보고서

201802088_류지현

1. 구현 아이디어

   주어진 데이터 c=3, k=3일 때, 낼 수 있는 금액은 1, 3, 9, 27이다. 총금액 P = 80원 일 때, 가장 높은 금액 동전부터 낮은 금액 동전 순서로 교환한다.

2. Greedy Choice

   1에서 설명했듯이 가장 높은 단위의 동전부터 교환하는 것을 이 문제의 Greedy Choice로 한다.

3. Greedy Choice Property

   27단위 동전을 먼저 교환한다고 하자. 그러면 2개의 동전을 교환하게 되고, 남은 금액은 P = 26원이 된다. 이후 27단위 동전을 교환할 수 없으므로 더 이상 영향을 주지 않게 된다. 이는 Greedy Choice Property를 만족한다.

   "***가장 높은 금액의 동전 교환을 가장 먼저 포함하는 최적 해가 반드시 존재한다.***"

4. Optimal Substructure Property

   가장 높은 단위의 동전을 먼저 교환을 하게 된다면, 가장 낮은 단위 동전을 먼저 교환하는 것보다 더 적은 수의 동전으로 교환할 수 있다. 이는 가장 높은 단위의 동전을 먼저 교환하는 것이 전체 교환 방법의 최적해임을 알 수 있다.

   "***i번째 높은 금액의 동전 교환 방법은 남은 교환 방법 중에서 가장 높은 금액의 동전 교환 방법이다.***"

5. 구현

Algorithm 1: exchageCoin

Data: $coin[], p$
Result: $result$

```
1 result = 0
2 n = coin.length − 1
3 for i = n down to 0 do
4 |   result = result + p/coin[i];
5 |   p = p%coin[i]
6 end
7 return result
```

6. 결과