

HOMEWORK #1

제출마감 : 2022. 9. 12. 09:00(AM)

제출장소 : 자연대연구실험동 연구실(313동 313호, VBLab) 앞에 있는 **리포트함** 혹은 **PLATO**

과제폴더

제출방법 : 프로그램 코드를 요구하는 프로그래밍을 직접해야 하는 경우를 제외하고 모든 숙제(과제)는 **손으로 답안을 작성(워드프로세서, 편집기 사용하지 않음)**해야 한다. 제출방법은 제출장소에 마감시간 이전에 직접 제출하거나, 온라인에 제출해야 하는 경우는 손으로 작성한 리포트를 스캔한 파일을 지정한 폴더에 제출한다.

1. 자료구조란 무엇인가?
 2. 자료구조를 공부하는 이유는 무엇인가?
 3. 알고리즘과 프로그램을 비교 설명하라.
 4. 자료구조를 설계할 때의 과정을 설명하라
 5. 화제마을에 있는 정미소는 일주일에 한번씩 문을 열기 때문에 마을 사람들은 이때를 기다려서 벼를 찍어 쌀을 만든다. 어느날 그 앞을 지나가다가 쌀 열섬을 쌓아 둔 것을 보았다. 이 정미소 앞에 쌓여 있는 쌀의 갯수를 추정해 보아라. [주]실현가능하도록 적절한 가정을 사용해도 무방하다.
-
6. 알고리즘의 빠르기를 측정하기 위한 2가지 요소는 무엇인가?
 7. 알고리즘을 빠르기를 표기할 때 점근적 방법을 사용하는 이유는 무엇인가?
 8. 점근적 표기법의 3가지 방법에 대해 설명하라.

1. 자료구조란 무엇인가?

A. 자료구조는 자료들을 효율적으로 사용할 수 있도록, 컴퓨터 메모리 내에서 자료를 구성하는 방법이다. 그로 인해, 그에 대한 정보 검색도 가능할 뿐 아니라 자료를 생산적으로 관리하고자 함이다. 대표적으로 사용되는 자료구조에는 리스트(List), 배열(Array), 스택(Stack), 큐(Queue), 힙(Heap), 트리(Tree), 그래프(Graph)가 있다.

2. 자료구조를 공부하는 이유는 무엇인가?

A. 자료구조는 문제를 풀기 위해 컴퓨터를 효율적으로 사용하려면, 필요한 아이디어를 잘 배치하고 좋은 방법으로 처리해야 하기 때문에 꼭 필요하다. 이를 배움으로써, 각 자료구조가 갖는 비용(costs)과 이득(benefits)에 대한 개념을 강화하며, 일반적인 자료구조를 학습해 기본 자료구조의 툴킷(toolkit)을 형성할 수 있다. 또한, 새로운 자료구조나 아는 사람이 안는 자료구조의 이점을 판단(분석, 생산성 따지기)해 그의 비용을 어떻게 평가할 지 이해하기 위해서이다.

3. 알고리즘과 프로그램을 비교 설명하라.

A. 알고리즘은 '절차'라고 생각하고, 프로그램은 그로 구현한 것이라고 보아야 한다. 즉 알고리즘은 문제를 해결하는 방법이고, 정확하고 길이가 유한해야 하며 모든 입력값에 대해 반드시 종료되어야 한다. 그로 프로그래밍 언어로 실제 구현한 것이 프로그램인 것이다.

4. 자료구조를 설계할 때의 과정을 설명하라.

A. 자료구조를 설계할 때에는 우선, 주어진 문제 해결에 필요한 자원을 결정하기 위해서 문제를 철저히 분석해야 한다. 이후 기본 단위를 결정하고 그 연산에 대해 자원 한계를 설정한다. 이후 이러한 요구사항에 가장 적합한 자료구조를 선택하면 된다.

5. 화제 마을에 있는 정미도는 열수일에 한 번씩 문을 열어 마을에 이를 사람들은 이때를 기다려서 벼를 찧어 쌀을 만든다. 어느날 그 앞을 지나가다가 쌀 찧는 성을 쌓아 둔 것을 보았다. 이 정미도 앞에 쌓여 있는 쌀알의 개수를 추정해 보이라. [주] 쌀 찧는 속도: 100kg/100kg = 500 (cup)

A. 쌀 한 컵이 150g 정도이다. 여기 쌀알이 1,000개 있다고 가정해보자. 쌀 한 가마니에는 쌀이 약 500 컵 있다고 하면, 그것은 쌀이 500,000 톤 있는 것이다. (한 가마니 무게는 80kg, 20kg/100kg = 500 (cup)) 쌀 한 성에 쌀 2 가마니 정도이니, 쌀 한 성에 쌀알은 약 1,000,000 톤 있다. 정미도 앞에는 쌀이 총 몇 톤 있으므로, 구하고자 하는 쌀알의 개수는 약 10,000,000 톤이다. (천만)

6. 알고리즘의 버그를 측정하기 위한 2가지 요소는 무엇인가?

A. 인지 성장률 (Growth Rate)을 고려해야 한다. 이는 입력값의 크기에 따라 일의 양이 어떻게 변화하는지를 나타내는 것으로, 주로 입력 크기 n 에 따라 $T(n)$ 으로 표현한다. 각 대안마다 성능이 다르므로 공정하게 평가하기 위해 $T(n)$ 을 제안한다.

또한 시간 복잡도를 분석하는 방법이 있는데, 이는 Best case, Worst case 및 Average case 4가지로 생각할 수 있다. Best case의 경우 첫 번째 원소에 찾고자 하는 것이 있으면, 알고리즘은 1번 수행되고, Worst case의 경우 마지막 원소가 찾는 원소이므로 n 번 수행된다. Average case는 배열 안에 찾고자 하는 원소가 있으므로, 평균적으로 알고리즘은 $n/2$ 번 수행된다. 여기서는 distribution (분포도)을 알아야 하고, real-time으로 작업할 때는 Worst case를 주로 고려하게 된다.

7. 알고리즘의 버그를 표기할 때 점근적 방법을 사용하는 이유는 무엇인가?

A. 상수 계수와 중요치 않은 항을 제거한 것을 점근적 표기법이라고 하는데, 이를 이용하면 이해하기 쉬운 부분을 없애고 알고리즘의 실행 시간 측정에서 성장률 (Growth Rate)이 집중될 수 있기 때문이다.

8. 점근적 표기법의 3가지 방법에 대해 설명하라.

A. 먼저, 「Big-O 표기법」은, 실행 시간이 있어서 상한선을 나타낸 것이다. 즉 '최대한'의 시간을 갖는 것이고, 「Big-Omega 표기법」은 반대로 실행 시간의 하한선을 나타낸 것이다. '최소한'의 시간이 걸린다고 표현할 수 있다. 마지막으로 「Big-Theta 표기법」은 실행 시간의 점진적 한계 범위 (근접한 한계값의 범위)를 나타내는 것이다.