**데이터구조 및 프로그래밍실습 3분반**

**설계 프로젝트**

학번 : 202117423

이름 : 문세현

본인의 Github 주소 : https://github.com/sehyunMoon123/dada-structure-lab-2024

(본 과제의 코드를 본인의 Github에 업로드 하세요)

**문제 1.**

1. 본 문제를 풀기위해 사용한 데이터 구조를 제시하고 간단히 설명하세요.

입력받은 숫자들을 리스트 형태로 저장하고 리스트의 숫자들을 모두 더한 후 리스트의 길이, 즉 숫자 개수로 나누어 평균을 구한다. 입력값은 모두 5의 배수이므로 평균이 자연 수가 나오지 않는 경우는 고려하지 않아도 된다. 중앙값은 리스트를 오름차순으로 정렬 했을 때 중앙(3번째)에 오는 값이다. 입력값이 5개이므로 숫자의 개수가 짝수개인 경우 는 고려하지 않아도 된다. 조건에 맞지 않는 입력이 들어오거나 숫자가 아닌 입력이 들어온 경우 조건에 맞는 숫자 또는 유효한 숫자를 입력하라는 문구를 출력한다.

1. 본인이 작성한 파이썬 Code 캡처 이미지를 첨부하고 Algorithm Analysis를 수행하세요.

sum(input)은 리스트의 요소들을 더하는 것이므로 시간복잡도는 O(n), len(input)은 리스 트의 길이를 구하는 것이므로 시간복잡도는 O(1), sorted(input)는 리스트를 정렬하는 것 이므로 시간복잡도는 O(nlogn), sorted\_input[2]는 리스트의 요소를 고르는 것이므로 시간 복잡도는 O(1), 입력을 받는 부분의 시간복잡도는 O(1)이므로 코드의 시간복잡도는 O(nlogn)이다.



**문제 2.**

1. 본 문제를 풀기위해 사용한 데이터 구조를 제시하고 간단히 설명하세요.

제일 처음 열린 괄호는 닫힐 때까지 기다려야 하고 닫는 괄호는 여는 괄호의 역순으로 나타나므로 스택을 사용한다. 입력의 괄호열 중 '('인 경우에는 '('를 스택에 추가하고 ')'인 경우에는 스택의 마지막 요소가 '('라면 서로 짝을 이루므로 '('를 스택에서 제거하고 스택에 요소가 없다면 ')'를 스택에 추가한다. 스택에 남은 괄호의 개수가 입력된 괄호열 을 올바른 괄호열로 만들기 위해 필요한 괄호의 최소 개수이다. 조건에 맞지 않는 입력 이 들어온다면 조건에 맞는 길이의 괄호열을 입력하라는 문구를 출력한다.

1. 본인이 작성한 파이썬 Code 캡처 이미지를 첨부하고 Algorithm Analysis를 수행하세요.

스택 연산의 시간복잡도는 O(1)이므로 for문의 시간복잡도는 O(n), 입력을 받는 부분의 시간복잡도는 O(1)이므로 코드의 시간복잡도는 O(n)이다.



**문제 3.**

1. 본 문제를 풀기위해 사용한 데이터 구조를 제시하고 간단히 설명하세요.

곰의 위치로부터 먹을 수 있는 벌집까지의 최단 경로를 구해야 하므로 너비 우선 탐색 (bfs)을 사용한다. problem3 함수는 곰의 초기 위치를 찾고 초기 위치에서부터 곰이 이동 할 수 있는 위치를 탐색하기 위해 bfs 함수를 호출한다. bfs 함수를 이용해 벌집을 먹고 곰의 크기와 같은 수의 벌집을 먹을 때마다 곰의 크기를 1씩 증가시킨다. 먹을 수 있는 벌집을 전부 먹은 경우 곰이 이동하며 소요된 시간을 반환한다. bfs함수는 너비 우선 탐 색(덱을 사용해 구현)으로 곰의 초기 위치에서부터 주위를 탐색한다. 상하좌우 순으로 탐 색하도록 설정하여 먹을 수 있는 벌집 중에서 상단 왼쪽 벌집부터 먹도록 한다. 이동 가 능한 위치를 큐에 추가하고 먹을 수 있는 벌집을 리스트에 추가한다. 먹을 수 있는 벌집 의 위치와 벌집까지의 거리를 반환한다. 입력을 받을 때 조건에 맞지 않는 입력이 들어 온다면 조건에 맞는 숲의 크기를 입력하라는 문구를 출력한다.

1. 본인이 작성한 파이썬 Code 캡처 이미지를 첨부하고 Algorithm Analysis를 수행하세요.

곰의 초기 위치를 찾는 부분은 이중 for문이므로 시간복잡도는 O(n^2), 덱을 사용하여 큐 연산의 시간복잡도가 O(1)이 되므로 bfs 함수의 시간복잡도는 N x N 크기의 숲을 탐 색하는 시간복잡도, 즉 O(n^2)이다. O(n^2)의 bfs 함수가 최대 n^2번 반복되므로 코드의 시간복잡도는 O(n^4)이다.

