**프로그램 보고서 #3**

1612169 배현진

(1) 사용한 각 함수에 대한 설명

1. int binsearch(int list[], int searchnum, int left, int right) {

int middle;

while (left <= right) {

middle = (right + left) / 2;

printf("\ncurrent middle: %d", middle);

if (list[middle] > searchnum)

right = middle - 1;

else if (list[middle] < searchnum)

left = middle + 1;

else

return middle;

}

return -1;

}

- Input: 정수형 배열인 list[], 배열에서 찾을 숫자인 searchnum, 배열의 가장 첫번째 인덱스인 left, 마지막 인덱스인 right

- Output: int

- 알고리즘 설명:

(1) 왼쪽 index가 오른쪽 index보다 작거나 같은 동안 while문을 수행한다.

1. 배열의 중간 index인 middle을 구한다. 그런 다음 현재 middle를 출력한다.

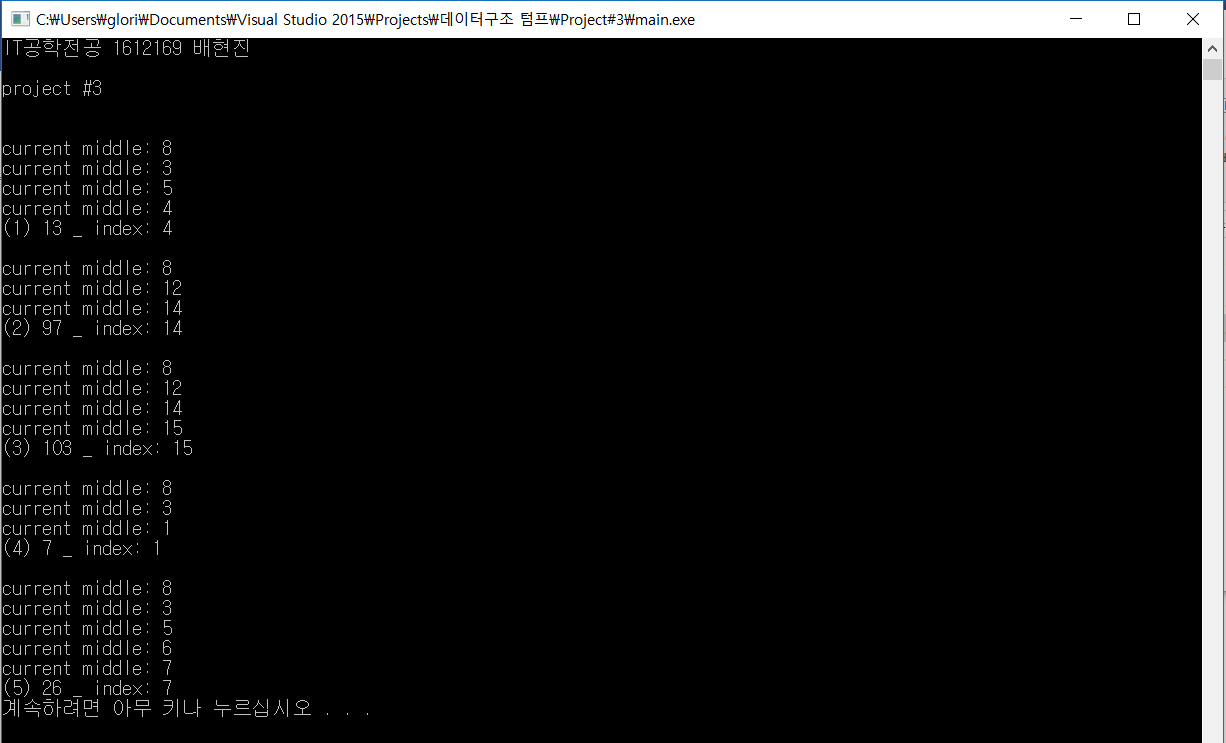
2. 만약 배열의 중간이 searchnum보다 큰 경우, 왼쪽 절반으로 탐색 범위를 줄인다.

3. 만약 배열의 중간이 searchnum보다 작은 경우, 오른쪽 절반으로 탐색 범위를 줄인다.

4. 배열의 중간과 searchnum이 같은 경우, 그 위치를 리턴한다.

(2) 어느 경우에도 속하지 않는 경우에는 -1을 출력한다.

(2) 실행 결과 화면 단계별 Capture



<전체화면>

(3) 주의사항

- Binary Search 연산을 수행할 때 배열을 사용해서 연산을 하기 때문에 index를 사용했다. 따라서 위치가 0부터 시작한다.