Q1. How many *comparisons* will *Binary Search* make on input A = [2,3,4,5,6,7,8] when searching for 6? Show some intermediate steps of the computation.

Q2.
```c
stack_t* s = stack_create();
push(s, 1);
push(s, 2);
push(s, 3);
printf("%d ", pop(s));
push(s, 5);
push(s, 6);
printf("%d ", pop(s));
printf("%d ", pop(s));
push(s, 7);
```

a) What will be the output of this code fragment?
b) What will be the state of the stack in the end?
Show the intermediate steps of the computation.
Q1:
The algorithm first compares to 5. We have 6>5 hence we search for 6 in [6,7,8].
Then it compares to 7. We have 6<7 hence we search for 6 in [6]
We find 6.
Total: 3 comparisons.

Q2:

```c
stack_t* s = stack_create(); // s = []
push(s, 1);       // s = [1]
push(s, 2);       // s = [1,2]
push(s, 3);       // s = [1,2,3]
printf("%d ", pop(s)); // s = [1,2]  output 3
push(s, 5);       // s = [1,2,5]
push(s, 6);       // s = [1,2,5,6]
printf("%d ", pop(s)); // s = [1,2,5]  output 6
printf("%d ", pop(s)); // s = [1,2]  output 5
printf("%d ", pop(s)); // s = [1]  output 2
push(s, 7);       // s = [1,7]
```
In the end s = [1,7]