구조체 배열 스택과 동적 할당 배열 스택의 차이점

5645350 성호준

```
void push(StackType* sp, element item) //데이터 삽입
{
        if (is_full(sp))
        {
                fprintf(stderr, "스택 포화 에러\n");
                return;
        }
        else
        {
                sp->data[++(sp->top)] = item;
        }
}
```

//구조체 스택의 push함수 부분

구조체 스택에서는 스택이 가득 찼을 때, 더 이상 스택에 값을 넣지 못하고 스택 포화 에러가 뜬다.

(▲ 구조체 스택을 사용한 결과값 일부)

```
30  void push(StackType* sp, element item)
31  {
32          if (is_full(sp))
33          {
34                  sp->capacity=sp->capacity*2;
35                  sp->data = (element*)realloc(sp->data, sp->capacity * sizeof(element));
36          }
37          sp->data[++(sp->top)] = item;
38  }
```

//동적 배열 스택의 push함수 부분

구조체 스택과는 다르게, 동적 배열 스택에서는 스택이 가득차더라도
공간의 크기를 늘려주어 스택 포화 에러가 없다.
realloc 함수는 이미 동적할당되어 있는 공간의 크기를 늘려주는 역할을 한다. 그래서
realloc 함수를 사용하여 원래 스택에 있는 값은 그대로 유지되고, 크기만 늘어나게 된다.
위 코드에서는 만약 스택이 가득 찬다면 스택의 크기를 두 배 늘려 준다.

(▲ 동적 배열 스택을 사용한 결과값 일부)