# Practice05-1: 문자열 뒤집기 20212426 정채호준 P05

## L1 루프: 스택에 문자열 푸시

이름: “jeongchaehojun” --size: 14

ESI는 문자열 인덱스를 따라 0부터 증가했고, EAX에는 해당 문자가 저장되어 스택에 push되었습니다. ECX는 14에서 시작해 반복마다 1씩 감소했고, ESP는 push될 때마다 4씩 감소하였습니다.

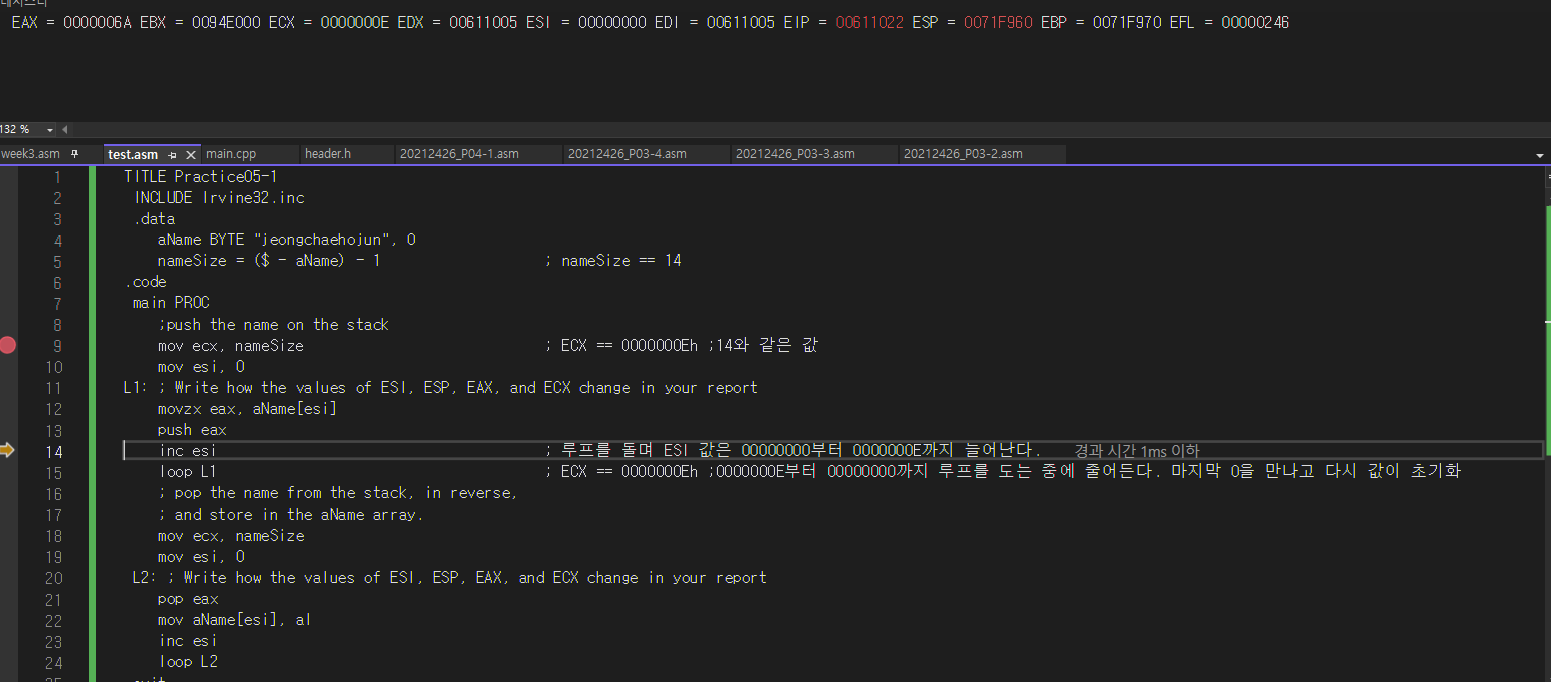
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 루프 | ESI | aName[ESI] | EAX (push 값) | ESP 변화 | ECX |
| 1 | 0 | 'j' | 0000006A | ESP - 4 | 13 |
| 2 | 1 | 'e' | 00000065 | ESP - 4 | 12 |
| ... | ... | ... | ... | ... | ... |
| 14 | 13 | 'n' | 0000006E | ESP - 4 | 0 |

## L2 루프: 문자열에 역순 저장

Jeongchaehojun -> nujoheahcgnoej

EAX에는 pop된 값이 들어오며 AL을 aName[ESI]에 저장하였습니다. ESI는 0부터 증가했고, ECX는 14에서 0까지 감소하였습니다. ESP는 pop할 때마다 4씩 증가했습니다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 루프 | EAX (pop 값) | AL | aName[ESI] 저장값 | ESI | ECX |
| 1 | 0000006E | 'n' | aName[0] = 'n' | 1 | 13 |
| 2 | 00000075 | 'u' | aName[1] = 'u' | 2 | 12 |
| ... | ... | ... | ... | ... | ... |
| 14 | 0000006A | 'j' | aName[13] = 'j' | 14 | 0 |



# Practice05-2: 배열의 합 구하기

루프는 배열 요소 5개를 순차적으로 더하여 EAX에 누적합니다. ESI는 4씩 증가하며 다음 배열 요소를 가리키고, ECX는 반복마다 감소합니다. ESP는 루프 중 변화가 없고, 함수 진입/종료 시 push/pop으로 변화합니다.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 루프 | EAX 누적값 | [ESI] 값 | ESI 변화 | ECX | ESP |
| 1 | 00010000h | 10000h | ESI + 4 → 004 | 4 | 유지 |
| 2 | 00030000h | 20000h | ESI + 4 → 008 | 3 | 유지 |
| 3 | 00060000h | 30000h | ESI + 4 → 00C | 2 | 유지 |
| 4 | 000A0000h | 40000h | ESI + 4 → 010 | 1 | 유지 |
| 5 | 000F0000h | 50000h | ESI + 4 → 014 | 0 | 유지 |

함수 진입 시 ESP는 003EFF6Ch에서 시작하여 push 두 번 후 003EFF68h이 되었고, 함수 종료 시 pop으로 원래대로 복구되었습니다.

