Assembly Language – Procedures (1)

5-1 External Library, Stack Operation

Objective: Understanding the operation and implementation of Stack

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ; Print String in reverse order  .data  myString BYTE "kcatS"  .code1  main PROC  mov ecx, LENGTHOF myString  mov esi, 0  L1:  movzx eax, myString[esi]  push eax  inc esi  loop L1  L2:  pop eax  ; Pop the string from the stack  loop L2;WriteChar: to display char in *AL*  mov ecx, stringSize  ##L3:  pop eax  call WriteChar  loop L3  exit  main ENDP  END main  HELLOWORLD | 1. Based on the codes in the left, after the program stops at L1 the first time, the values in the registers are:  |  |  |  |  | | --- | --- | --- | --- | | EAX | 75e13c33h | EBP | 0012ff94h | | EBX | 7ffdf000h | ESP | 0012ff8ch | | ECX | 00000005h | ESI | 00000000h | | EDX | 00401000h | EDI | 00000000h |     b. For question a, try to fill in the data status in Stack when the program is executed to L2.   |  |  |  |  | | --- | --- | --- | --- | | EAX | 00000053h | EBP | 0012ff94h | | EBX | 7ffdf000h | ESP |  | | ECX |  | ESI |  | | EDX | 00401000h | EDI | 00000000h |   C. Following Problem **b**, what are the data.currently stored in the stack?   |  |  | | --- | --- | | 0012ff78h |  | | 0012ff7ch |  | | 0012ff80h |  | | 0012ff84h |  | | 0012ff88h |  | | 0012ff8ch | 75e13c45h |   The table below gives ASCII Codes in hexadecimal  "kcatS"   |  |  | | --- | --- | | Character | ASCII Code | | 'S' | 53h | | 'a' | 61h | | 'c' | 63h | | 'k' | 6bh | | 't' | 74h | |