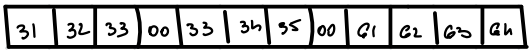
String functions - Memory layout & using them in data transfer instr.

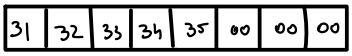
* String type constants does only the reservation of the required space, the filling order of that memory area are in the order that the char. appear

Ex:

aC dd '123' '345' 'abcd'

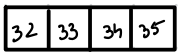


aG dd '12345'



* has to be filled up to the next dword

a7 dw '23', '45'



In NASM '...' = "..." but in C '...' ≠ "..." !

the following def. provide the same mem. conf

```
dd 'nine chars'
dd 'nine','char','s'
db 'nine chars', 0, 0, 0
```

mov eax, 'abcd' ; EAX = 0x 64 63 62 61

Constant	Constant table
'abcd'	'dcba'
'2345'	'5432'

only if it is in the constant table (in memory it would be stored in reverse)

mov dword [a], '2345' will be DS:[401 0000], 35 34 33 32

this is the actual 'value' of the constant

the hexadecimal values associated to that memory

Short JMP and Long JMP

JMP - unconditional jmp

in TASM on 16 bits you had 2 restrictions:

- conditional jumps
- Loop label

short jumps

the distance must be at most 127 bytes

in NASM the condition was eliminated ⇒ No restriction for conditional jumps

BUT for loops:

```
Again
    mov eax, 89
    --------
    jmp further
        resd 1000h
further:
    mov ebx, 17
    ------
```

the distance is > 127 bytes ⇒ it is NOT short jumps

```
loop Again ; syntax error ; short jump is out of range + byte data exceeds bounds
```

replacing loop Again with { dec ecx / jnz Again } will NOT be an error

Continuare vineri

JMP instruction analysis NEAR and FAR jump

↳ jump to a 4 byte label ⇒ NEAR jump

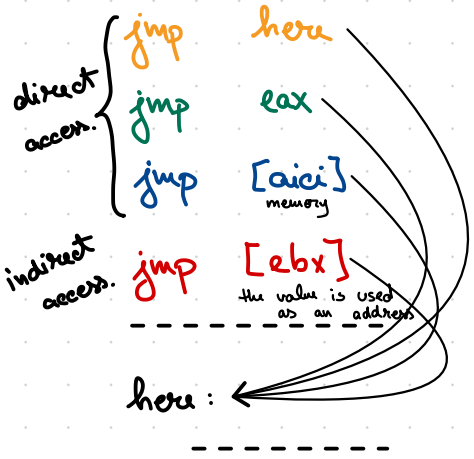
How do you do a FAR jump ? With a FAR address : segment + offset ⇒ 6 byte address

↳ you can perform a FAR jump only by using a 6 byte variable pointer

↳ CS : EIP ⟶ currently executed instruction

* a FAR jump is a legal way in which you can change the CS : EIP value
* with a jump you change EIP

```
segment data
    aici dd here ← initilized with the offset of the label "here"
segment code
    mov eax, [aici]
    mov ebx, aici
    ------------
```



jmp [ss: ESP + 12] is still a near jump

! you must use FAR

jmp far [ss: ebx + 12] is an actual far jump

9b 7a 52 C1 c2 C5 ⇒ EIP: C1 52 7a 9b CS: C5 c2

* Rezumat 15 dec ! all important information

* vezi în Olly : push [var] dacă are DS: sau SS: