

Practica 1 - Subrutinas

28 de agosto de 2018

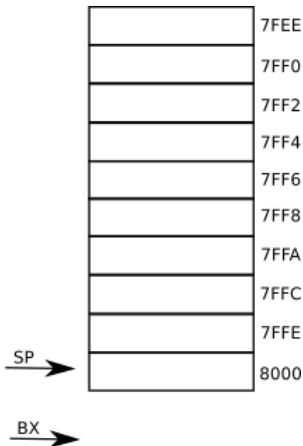
Subrutinas

```

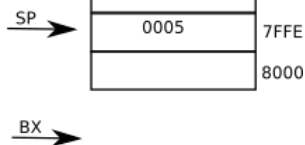
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
      MOV BX, SP
      PUSH CX
      PUSH AX
      PUSH DX
      ADD BX, 6
      MOV CX, [BX]
      MOV BX, [BX]
      ADD BX, 2
      MOV AX, [BX]
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      SUB BX, 4
      MOV AX, [BX]
      MOV BX, AX
      MOV [BX], DX
      POP DX
      POP AX
      POP CX
      POP BX
      RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END

```



```
ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END
```



The diagram illustrates a memory stack. It consists of a vertical column of 10 rectangular boxes representing memory cells. To the right of each box is its corresponding hexadecimal address. From top to bottom, the addresses are: 7FEE, 7FF0, 7FF2, 7FF4, 7FF6, 7FF8, 7FFA, 7FFC, 7FFE, and 8000. An arrow labeled 'SP' (Stack Pointer) points to the box at address 7FFC, which contains the value '0003'. The box at address 7FFE contains the value '0005'. The bottom-most box at address 8000 is empty. An arrow labeled 'BX' (Base Register) points to the bottom of the stack, below the 8000 address.

	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
SP →	7FFC 0003
	7FFE 0005
	8000

BX →

Diagram illustrating the memory stack layout. The stack grows downwards (increasing address). The memory addresses and their contents are:

Address	Content
7FEE	
7FF0	
7FF2	
7FF4	
7FF6	
7FF8	
7FFA	OFFSET RES
7FFC	0003
7FFE	0005
8000	

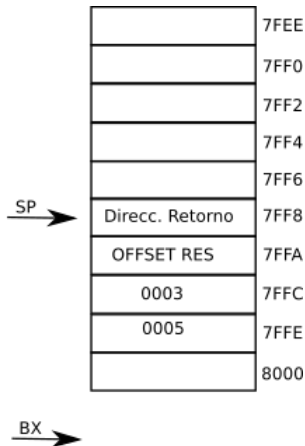
The SP register points to address 7FFA, and the BX register points to address 8000.

Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END
  
```



	7FEE
	7FF0
	7FF2
	7FF4
SP →	Valor de BX 7FF6
	Direcc. Retorno 7FF8
	OFFSET RES 7FFA
	0003 7FFC
	0005 7FFE
	8000

BX →

Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END

```



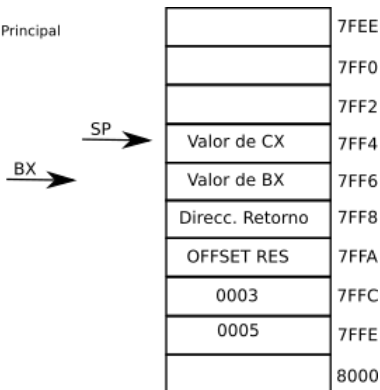
	7FEE
	7FF0
	7FF2
	7FF4
Valor de BX	7FF6
Direcc. Retorno	7FF8
OFFSET RES	7FFA
0003	7FFC
0005	7FFE
	8000

Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END
  
```



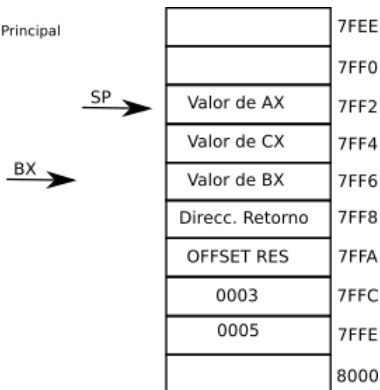
Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
      MOV BX, SP
      PUSH CX
      PUSH AX
      PUSH DX
      ADD BX, 6
      MOV CX, [BX]
      MOV BX, [BX]
      ADD BX, 2
      MOV AX, [BX]
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      SUB BX, 4
      MOV AX, [BX]
      MOV BX, AX
      MOV [BX], DX
      POP DX
      POP AX
      POP CX
      POP BX
      RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END

```

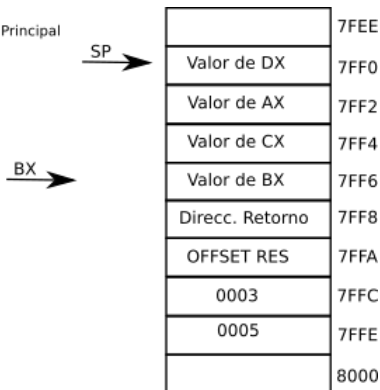


Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END
  
```

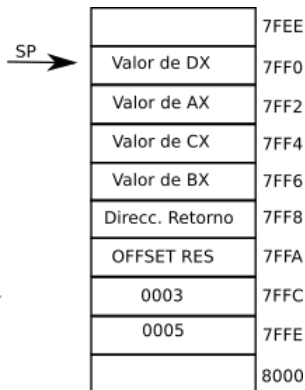


Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END
  
```

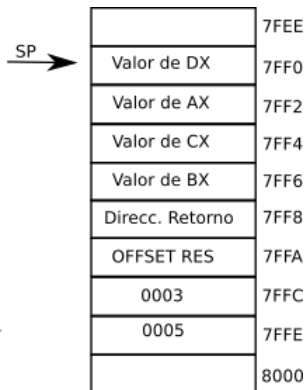


Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END
  
```



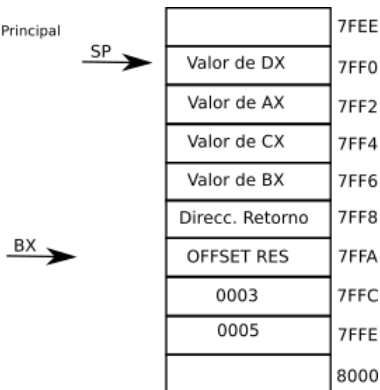
Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END

```



SP →

$$\xrightarrow{\text{BX}}$$

SP →

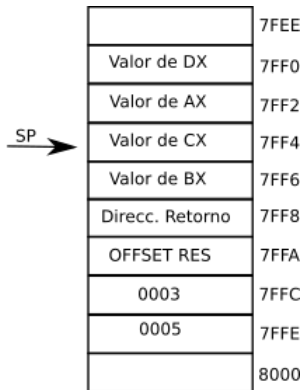
$$\xrightarrow{BX}$$

Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END
  
```



SP →

$$\xrightarrow{\text{BX}}$$

	7FEE
Valor de DX	7FF0
Valor de AX	7FF2
Valor de CX	7FF4
Valor de BX	7FF6
Direcc. Retorno	7FF8
OFFSET RES	7FFA
0003	7FFC
0005	7FFE
	8000

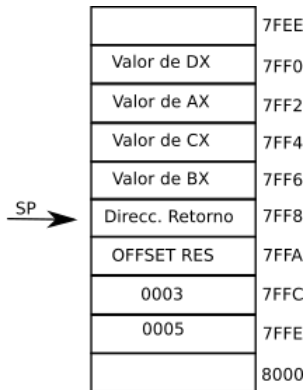
Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END

```



Subrutinas

```

ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    MOV BX, SP
    PUSH CX
    PUSH AX
    PUSH DX
    ADD BX, 6
    MOV CX, [BX]
    MOV BX, [BX]
    ADD BX, 2
    MOV AX, [BX]
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    SUB BX, 4
    MOV AX, [BX]
    MOV BX, AX
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP BX
    RET
  
```

ORG 2000H ; Programa Principal
 MOV AX, NUM1
 PUSH AX
 MOV AX, NUM2
 PUSH AX
 MOV AX, OFFSET RES
 PUSH AX
 MOV DX, 0
 CALL MUL
 POP AX
 POP AX
 POP AX
 HLT
 END

SP →

BX →

	7FEE
Valor de DX	7FF0
Valor de AX	7FF2
Valor de CX	7FF4
Valor de BX	7FF6
Direcc. Retorno	7FF8
OFFSET RES	7FFA
0003	7FFC
0005	7FFE
	8000

SP →

$$\xrightarrow{BX}$$

	7FEE
Valor de DX	7FF0
Valor de AX	7FF2
Valor de CX	7FF4
Valor de BX	7FF6
Direcc. Retorno	7FF8
OFFSET RES	7FFA
0003	7FFC
0005	7FFE
	8000

```

ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
HLT
END

```



```
ORG 2000H ; Programa Principal
MOV AX, NUM1
PUSH AX
MOV AX, NUM2
PUSH AX
MOV AX, OFFSET RES
PUSH AX
MOV DX, 0
CALL MUL
POP AX
POP AX
POP AX
HLT
END
```

SP →

$$\xrightarrow{\text{BX}}$$