

# Practica 1 - Subrutinas

1 de octubre de 2024

## Ejercicio 7 - Multiplicación de números sin signo

- El programa utiliza una subrutina para multiplicar dos números sin signo mayores que cero.
- Se pasan los números a multiplicar por valor.
- Se pasa por referencia la dirección donde se debe guardar el resultado.
- Todos los parámetros se pasan a través de la pila

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV  BX, SP
    ADD  BX, 12
    MOV  CX, [BX]
    ADD  BX, 2
    MOV  AX, [BX]
    SUB  BX, 4
    MOV  BX, [BX]
    MOV  DX, 0
```

```
SUMA: ADD  DX, AX
      DEC  CX
      JNZ  SUMA
      MOV  [BX], DX
      POP  DX
      POP  AX
      POP  CX
      POP  BX
      RET
```

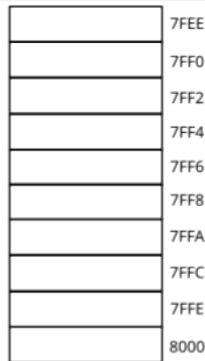
```
ORG 2000H ; Programa Principal
```

```
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
```

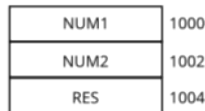
```
ORG 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES   DW ?
```

SP →



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV  BX, SP
    ADD  BX, 12
    MOV  CX, [BX]
    ADD  BX, 2
    MOV  AX, [BX]
    SUB  BX, 4
    MOV  BX, [BX]
    MOV  DX, 0
```

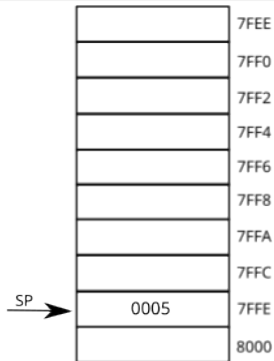
```
SUMA: ADD  DX, AX
      DEC  CX
      JNZ  SUMA
      MOV  [BX], DX
      POP  DX
      POP  AX
      POP  CX
      POP  BX
      RET
```

```
ORG 2000H ; Programa Principal
```

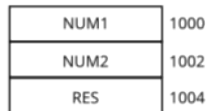
```
MOV  AX, NUM1
PUSH AX
MOV  AX, NUM2
PUSH AX
MOV  AX, OFFSET RES
PUSH AX
MOV  DX, 0
CALL MUL
POP  AX
POP  AX
POP  AX
HLT
END
```

```
ORG 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES   DW ?
```



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
```

```
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      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 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP →

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

PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

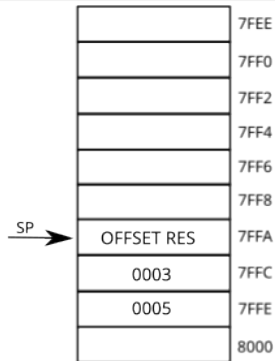
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

ORG 2000H ; Programa Principal

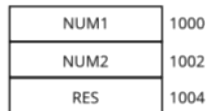
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END

ORG 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL

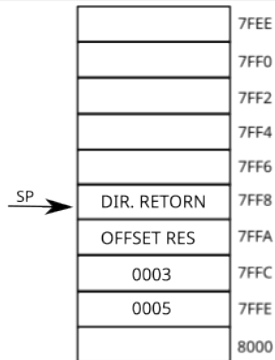
MUL: PUSH BX  
PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0  
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

ORG 2000H ; Programa Principal

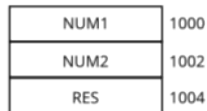
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END

ORG 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL

MUL: **PUSH BX**

PUSH CX

PUSH AX

PUSH DX

MOV BX, SP

ADD BX, 12

MOV CX, [BX]

ADD BX, 2

MOV AX, [BX]

SUB BX, 4

MOV BX, [BX]

MOV DX, 0

SUMA: ADD DX, AX

DEC CX

JNZ SUMA

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

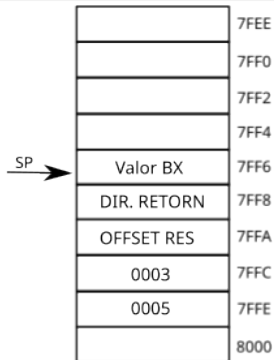
END

ORG 1000h ; Variables

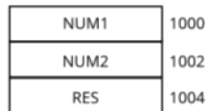
NUM1 DW 5H

NUM2 DW 3H

RES DW ?



PILA



VARIABLES



# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL

MUL: PUSH BX

PUSH CX

PUSH AX

PUSH DX

MOV BX, SP

ADD BX, 12

MOV CX, [BX]

ADD BX, 2

MOV AX, [BX]

SUB BX, 4

MOV BX, [BX]

MOV DX, 0

SUMA: ADD DX, AX

DEC CX

JNZ SUMA

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 1000h ; Variables

NUM1 DW 5H

NUM2 DW 3H

RES DW ?

SP →

	7FEE
	7FF0
	7FF2
Valor CX	7FF4
Valor BX	7FF6
DIR. RETURN	7FF8
OFFSET RES	7FFA
0003	7FFC
0005	7FFE
	8000

PILA

NUM1	1000
NUM2	1002
RES	1004

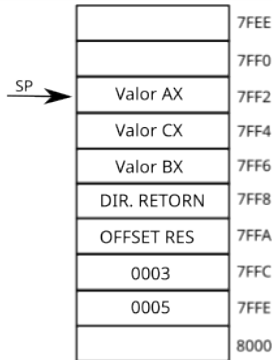
VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

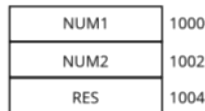
```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    MOV [BX], DX
    POP DX
    POP AX
    POP CX
    POP 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 1000h ; Variables
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```



## PILA



## VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    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 1000h ; Variables
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP →

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

**PILA**

NUM1	1000
NUM2	1002
RES	1004

**VARIABLES**

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
```

```
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
SUMA: ADD DX, AX
    DEC CX
    JNZ SUMA
    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 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP  
BX →

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

PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

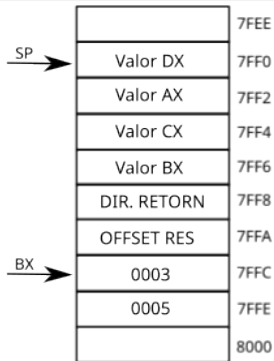
# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
**ADD BX, 12**  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0  
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
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 1000h ; Variables  
NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



## PILA

NUM1	1000
NUM2	1002
RES	1004

## VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
**ADD BX, 2**  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

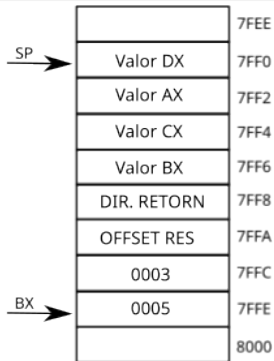
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

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 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL  
MUL: PUSH BX
```

```
    PUSH CX  
    PUSH AX  
    PUSH DX  
    MOV BX, SP  
    ADD BX, 12  
    MOV CX, [BX]  
    ADD BX, 2  
    MOV AX, [BX]
```

```
    SUB BX, 4
```

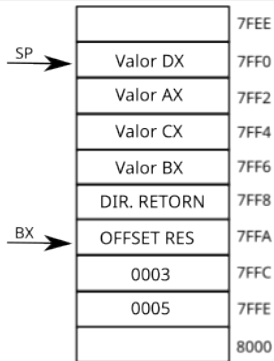
```
    MOV BX, [BX]  
    MOV DX, 0  
SUMA: ADD DX, AX  
    DEC CX  
    JNZ SUMA  
    MOV [BX], DX  
    POP DX  
    POP AX  
    POP CX  
    POP BX  
    RET
```

```
ORG 2000H ; Programa Principal
```

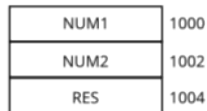
```
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END
```

```
ORG 1000h ; Variables
```

```
NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?
```



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL  
MUL: PUSH BX
```

```
    PUSH CX  
    PUSH AX  
    PUSH DX  
    MOV BX, SP  
    ADD BX, 12  
    MOV CX, [BX]  
    ADD BX, 2  
    MOV AX, [BX]  
    SUB BX, 4  
    MOV BX, [BX]  
    MOV DX, 0
```

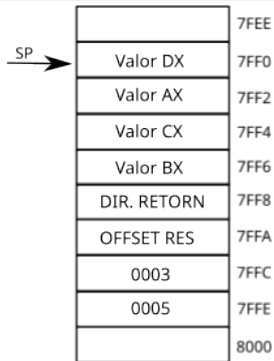
```
SUMA: ADD DX, AX  
      DEC CX  
      JNZ SUMA  
      MOV [BX], DX  
      POP DX  
      POP AX  
      POP CX  
      POP BX  
      RET
```

```
ORG 2000H ; Programa Principal
```

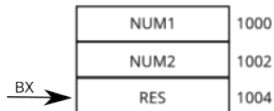
```
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END
```

```
ORG 1000h ; Variables
```

```
NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?
```



PILA



VARIABLES



# Ejercicio 7 - Pasaje de parámetros por pila

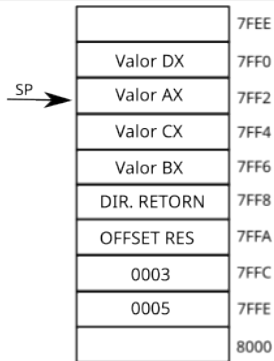
ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

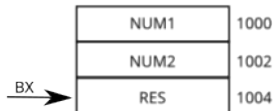
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
**POP DX**  
POP AX  
POP CX  
POP BX  
RET

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 1000h ; Variables  
NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

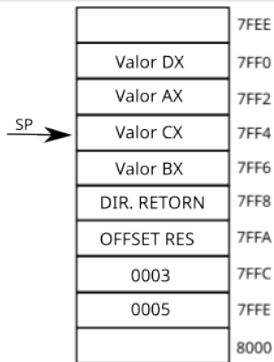
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

ORG 2000H ; Programa Principal

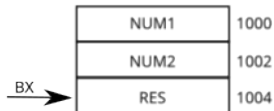
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END

ORG 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

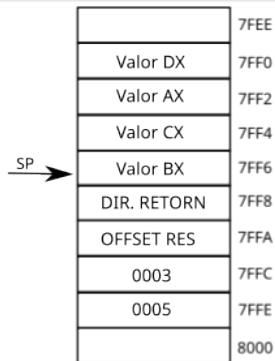
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

ORG 2000H ; Programa Principal

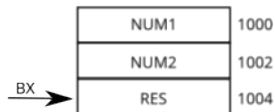
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
POP AX  
HLT  
END

ORG 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
```

```
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      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 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP →

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

PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL

MUL: PUSH BX

PUSH CX

PUSH AX

PUSH DX

MOV BX, SP

ADD BX, 12

MOV CX, [BX]

ADD BX, 2

MOV AX, [BX]

SUB BX, 4

MOV BX, [BX]

MOV DX, 0

SUMA: ADD DX, AX

DEC CX

JNZ SUMA

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

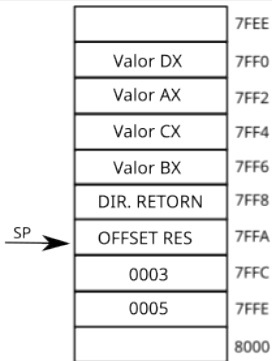
END

ORG 1000h ; Variables

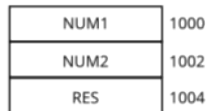
NUM1 DW 5H

NUM2 DW 3H

RES DW ?



PILA



VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
```

```
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      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
```

```
ORG 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP →

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

PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

```
ORG 3000H ; Subrutina MUL
MUL: PUSH BX
```

```
    PUSH CX
    PUSH AX
    PUSH DX
    MOV BX, SP
    ADD BX, 12
    MOV CX, [BX]
    ADD BX, 2
    MOV AX, [BX]
    SUB BX, 4
    MOV BX, [BX]
    MOV DX, 0
```

```
SUMA: ADD DX, AX
      DEC CX
      JNZ SUMA
      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
```

```
ORG 1000h ; Variables
```

```
NUM1 DW 5H
NUM2 DW 3H
RES DW ?
```

SP →

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

PILA

NUM1	1000
NUM2	1002
RES	1004

VARIABLES

# Ejercicio 7 - Pasaje de parámetros por pila

ORG 3000H ; Subrutina MUL  
MUL: PUSH BX

PUSH CX  
PUSH AX  
PUSH DX  
MOV BX, SP  
ADD BX, 12  
MOV CX, [BX]  
ADD BX, 2  
MOV AX, [BX]  
SUB BX, 4  
MOV BX, [BX]  
MOV DX, 0

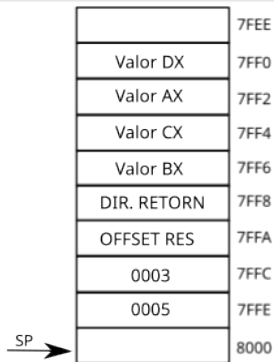
SUMA: ADD DX, AX  
DEC CX  
JNZ SUMA  
MOV [BX], DX  
POP DX  
POP AX  
POP CX  
POP BX  
RET

ORG 2000H ; Programa Principal

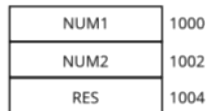
MOV AX, NUM1  
PUSH AX  
MOV AX, NUM2  
PUSH AX  
MOV AX, OFFSET RES  
PUSH AX  
MOV DX, 0  
CALL MUL  
POP AX  
POP AX  
HLT  
END

ORG 1000h ; Variables

NUM1 DW 5H  
NUM2 DW 3H  
RES DW ?



PILA



VARIABLES



## Ejercicio 10 - SWAP (intercambio)

- Escribir una subrutina SWAP que intercambie dos datos de 16 bits almacenados en memoria.
- Los parámetros deben ser pasados por referencia desde el programa principal a través de la pila.

# Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	0000
BX	0000
CX	0000
DX	0000

REGISTROS



	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
	7FFC
	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1000
BX	0000
CX	0000
DX	0000

REGISTROS



	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
	7FFC
	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

# Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1000
BX	0000
CX	0000
DX	0000

## REGISTROS



	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
	7FFC
	7FFE
	8000

## PILA

1234	val1 1000
5678	val2 1002

## VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

ORG 3000H  
SWAP: PUSH BX  
PUSH AX  
PUSH DX  
  
MOV BX, SP  
ADD BX, 8  
MOV BX, [BX]  
MOV CX, [BX]  
  
MOV BX, SP  
ADD BX, 10  
MOV BX, [BX]  
MOV DX, [BX]  
  
MOV BX, SP  
ADD BX, 8  
MOV BX, [BX]  
MOV [BX], DX  
  
MOV BX, SP  
ADD BX, 10  
MOV BX, [BX]  
MOV [BX], CX  
  
POP DX  
POP AX  
POP BX  
RET

ORG 1000H  
val1 DW 1234H  
val2 DW 5678H  
  
ORG 2000H  
MOV AX, offset val1  
PUSH AX  
MOV AX, offset val2  
PUSH AX  
CALL SWAP  
HLT

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
DIR. RETORNO	7FFC
1002	7FFE
1000	
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

ORG 3000H

SWAP: PUSH BX

PUSH AX

PUSH DX

MOV BX, SP

ADD BX, 8

MOV BX, [BX]

MOV CX, [BX]

MOV BX, SP

ADD BX, 10

MOV BX, [BX]

MOV DX, [BX]

MOV BX, SP

ADD BX, 8

MOV BX, [BX]

MOV [BX], DX

MOV BX, SP

ADD BX, 10

MOV BX, [BX]

MOV [BX], CX

POP DX

POP AX

POP BX

RET

ORG 1000H

val1 DW 1234H

val2 DW 5678H

ORG 2000H

MOV AX, offset val1

PUSH AX

MOV AX, offset val2

PUSH AX

CALL SWAP

HLT

AX

1002

BX

0000

CX

0000

DX

0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES



## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS



	7FEE
	7FF0
	7FF2
	7FF4
	7FF6
	7FF8
	7FFA
	7FFC
	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS



	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX
      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FF4
CX	0000
DX	0000

REGISTROS

SP  
BX →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	0000
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
BX →	1002 7FFC
	1000 7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1002
CX	0000
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
	1002 7FFC
	1000 7FFE
	8000

PILA

	1234	val1 1000
BX →	5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1002
CX	5678
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

BX →

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FF4
CX	5678
DX	0000

REGISTROS

SP  
BX →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FFE
CX	5678
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
	1002 7FFC
BX →	1000 7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES



## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
	1002 7FFC
	1000 7FFE
	8000

PILA

BX →	1234 val1 1000
	5678 val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000
	1002
	7FF6
	0000
	7FF8
	DIR. RETORNO
	1002
	7FFC
	1000
	7FFE
	8000

PILA

BX →	1234	val1 1000
	5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FF4
CX	5678
DX	1234

REGISTROS

SP  
BX →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

# Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FFC
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
BX →	1002 7FFC
	1000 7FFE
	8000

PILA

1234	val1 1000
5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1002
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000
	7FF4
	1002
	7FF6
	0000
	7FF8
	DIR. RETORNO
	7FFA
	1002
	7FFC
	1000
	7FFE
	8000

PILA

	1234	val1 1000
BX →	5678	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1002
CX	5678
DX	1234

REGISTROS



	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA



1234	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FF4
CX	5678
DX	1234

REGISTROS

SP  
BX →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

1234	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	7FFE
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
	1002 7FFC
BX →	1000 7FFE
	8000

PILA

1234	val1 1000
1234	val2 1002

VARIABLES



## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000
	7FF4
	1002
	7FF6
	0000
	7FF8
	DIR. RETORNO
	7FFA
	1002
	7FFC
	1000
	7FFE
	8000

PILA

BX →	1234	val1
	1234	1000
		val2
		1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	1234

REGISTROS

	7FEE
	7FF0
	7FF2
SP →	0000 7FF4
	1002 7FF6
	0000 7FF8
	DIR. RETORNO 7FFA
	1002 7FFC
	1000 7FFE
	8000

PILA

BX →	5678	val1 1000
	1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

5678	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	1000
CX	5678
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

BX →

5678	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	5678
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

5678	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H

ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	5678
DX	0000

REGISTROS

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

5678	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP

```
ORG 3000H
SWAP: PUSH BX
      PUSH AX
      PUSH DX

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV CX, [BX]

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV DX, [BX]

      MOV BX, SP
      ADD BX, 8
      MOV BX, [BX]
      MOV [BX], DX

      MOV BX, SP
      ADD BX, 10
      MOV BX, [BX]
      MOV [BX], CX

      POP DX
      POP AX
      POP BX
      RET
```

```
ORG 1000H
val1 DW 1234H
val2 DW 5678H
```

```
ORG 2000H
MOV AX, offset val1
PUSH AX
MOV AX, offset val2
PUSH AX
CALL SWAP
HLT
```

AX	1002
BX	0000
CX	5678
DX	0000

REGISTROS

SP →

	7FEE
	7FF0
	7FF2
0000	7FF4
1002	7FF6
0000	7FF8
DIR. RETORNO	7FFA
1002	7FFC
1000	7FFE
	8000

PILA

5678	val1 1000
1234	val2 1002

VARIABLES

## Ejercicio 10 - SWAP (intercambio)

El programa cumple con el enunciado, pero...

- No desapila los dos parámetros de la pila
- La subrutina preserva los registros AX, BX y DX. Pero modifica los registro BX, CX y DX. No es necesario preservar AX, CX si.