

Chapter 8

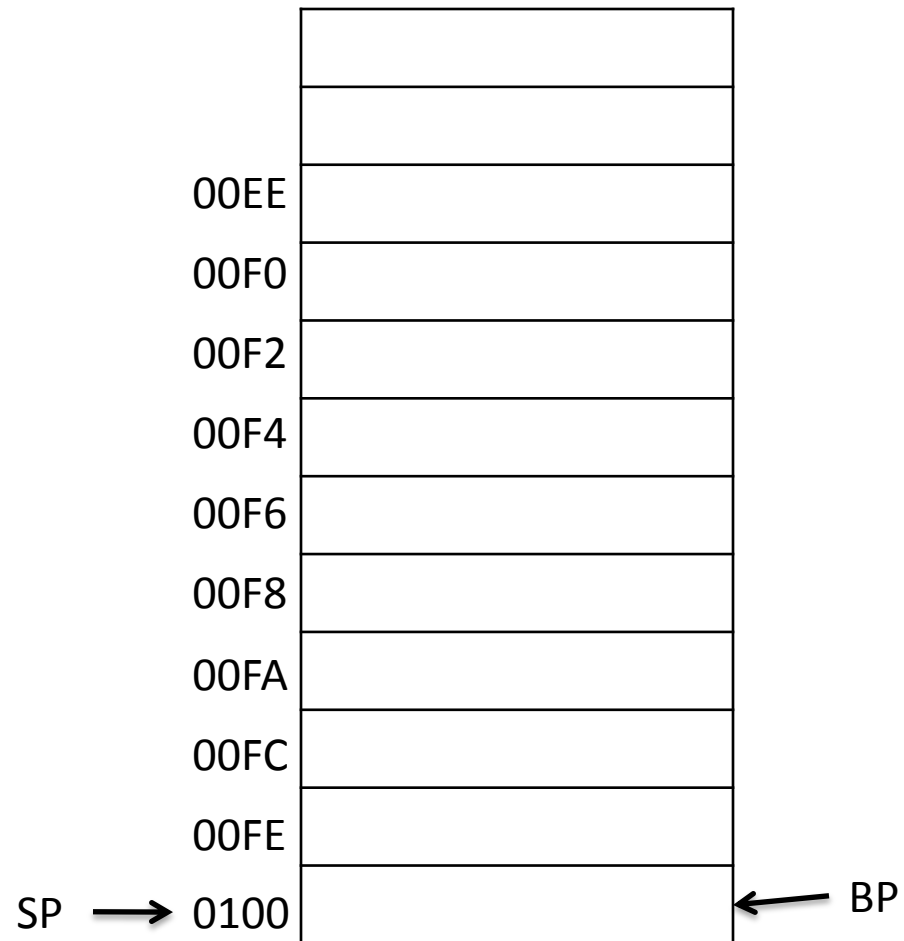
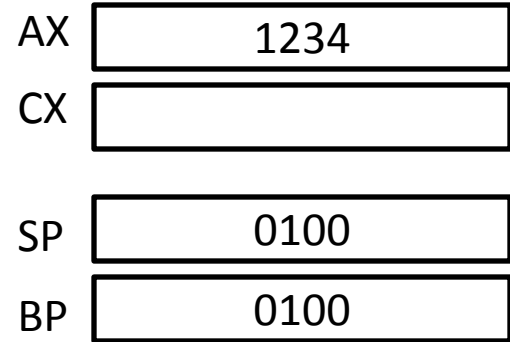
The Stack and Introduction to Procedures

PUSH

PUSH source

- source is a 16-bit register / memory word

PUSH AX

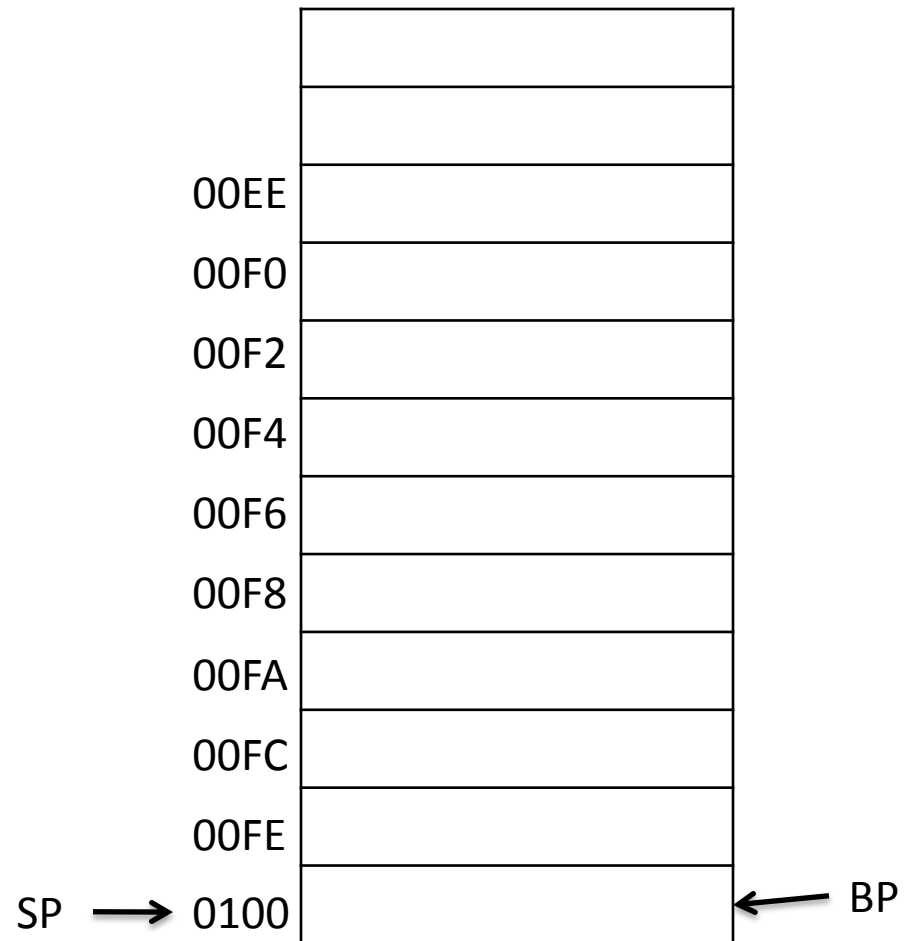
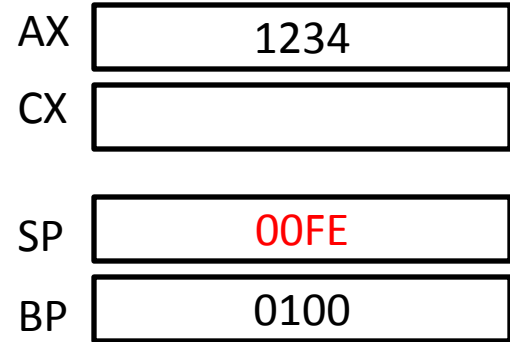


PUSH

PUSH source

•source is a 16-bit register / memory word

PUSH AX

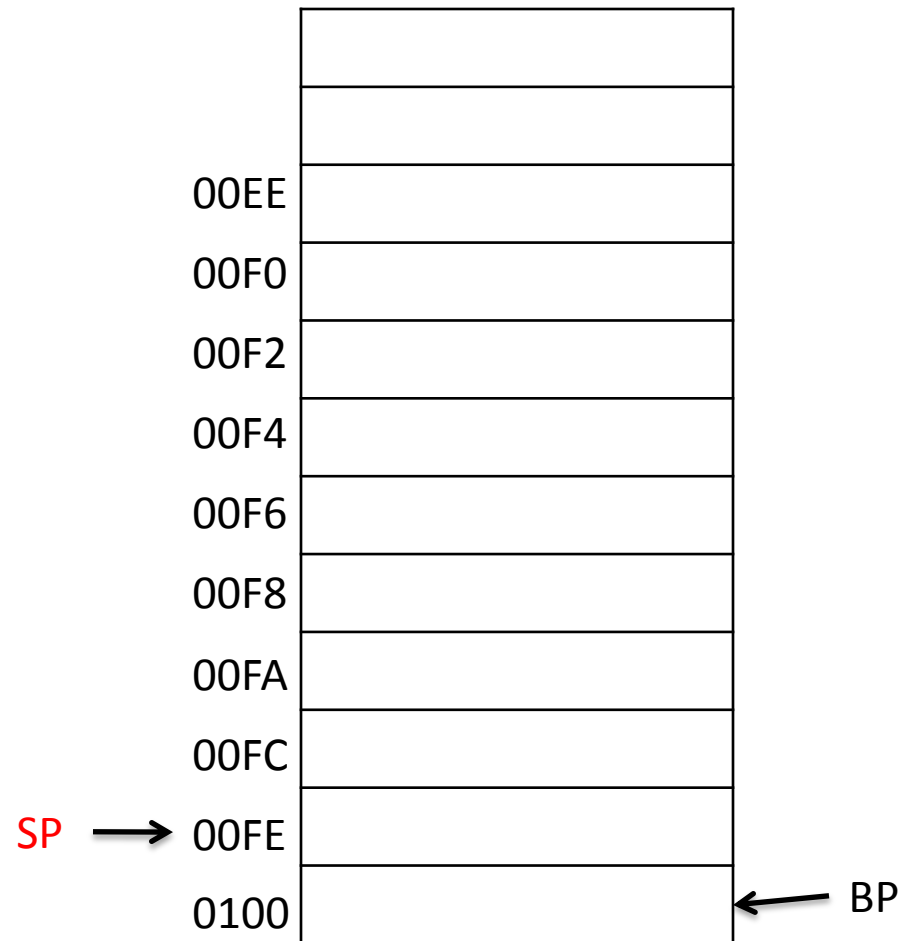
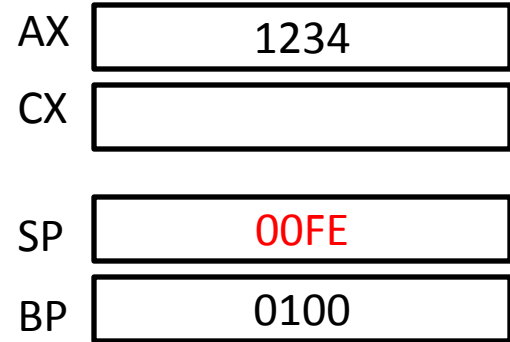


PUSH

PUSH source

- source is a 16-bit register / memory word

PUSH AX

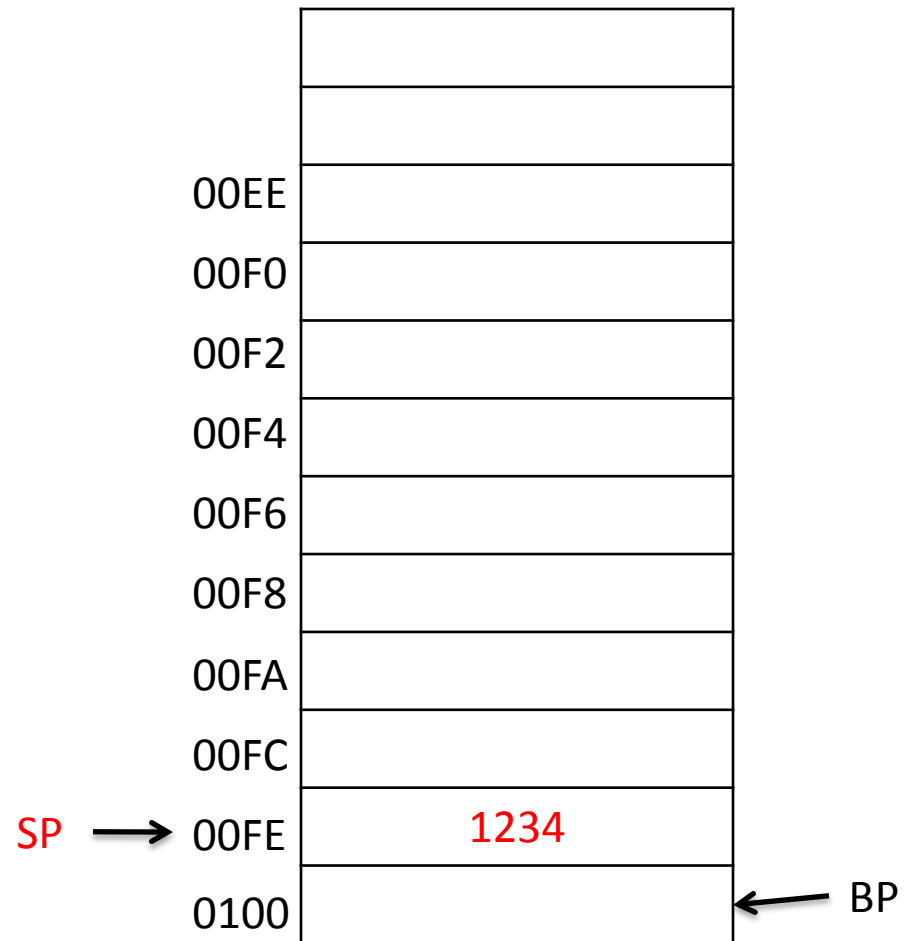
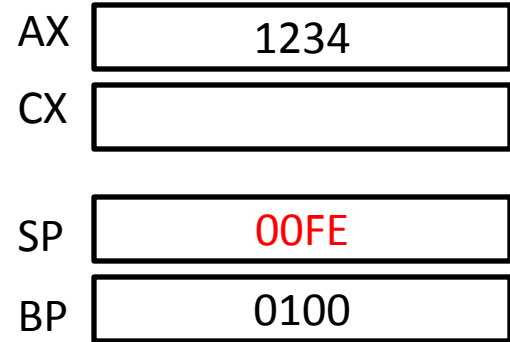


PUSH

PUSH source

- source is a 16-bit register / memory word

PUSH AX

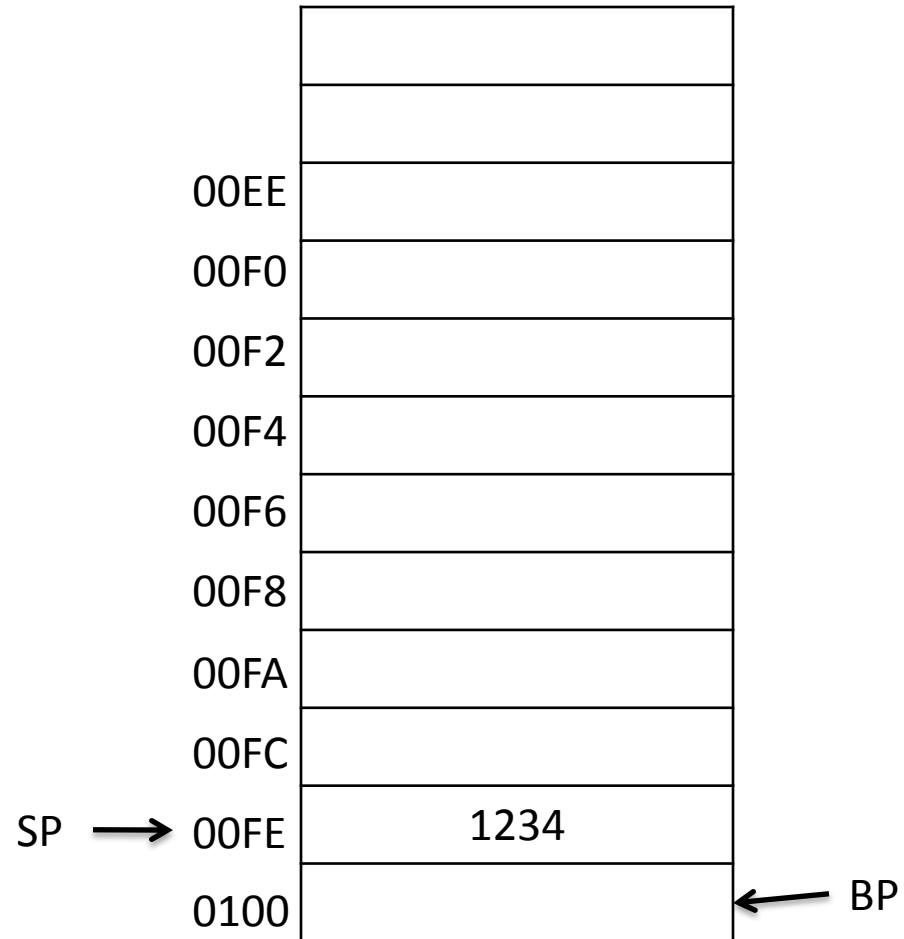
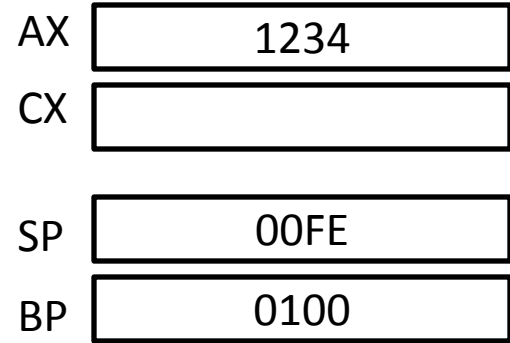


POP

POP destination

- destination is a 16-bit register(except IP register) / memory word

POP CX



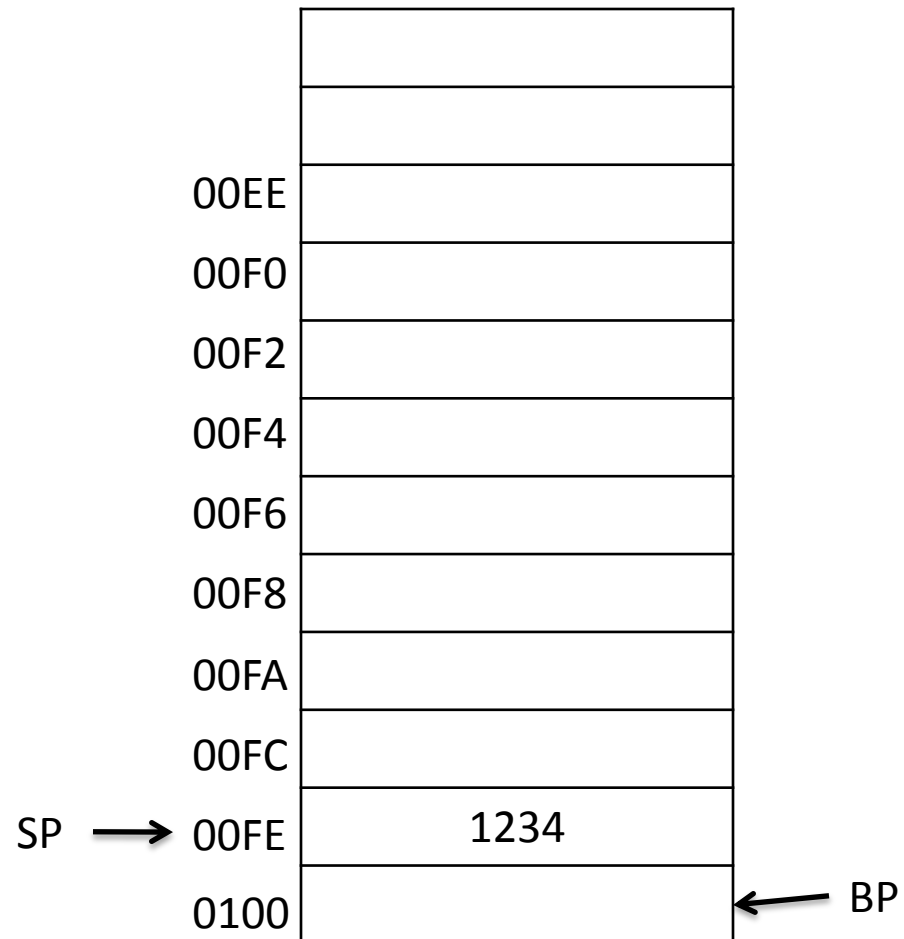
POP

POP destination

- destination is a 16-bit register(except IP register) / memory word

POP CX

AX	1234
CX	1234
SP	00FE
BP	0100

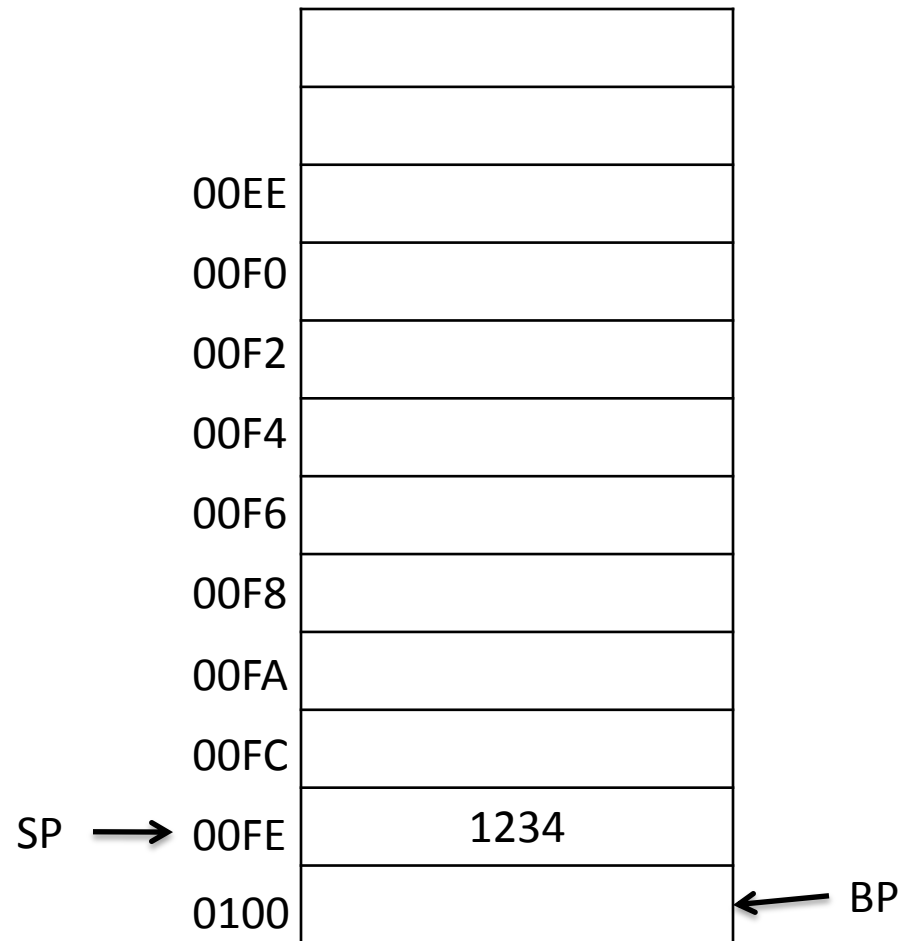
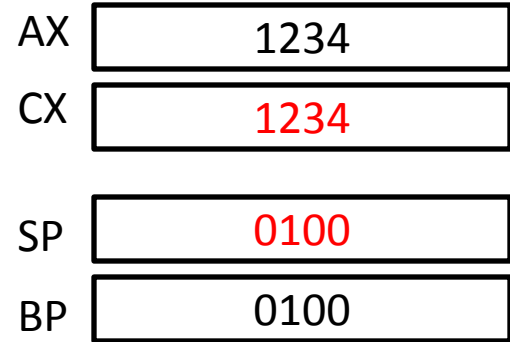


POP

POP destination

- destination is a 16-bit register(except IP register) / memory word

POP CX

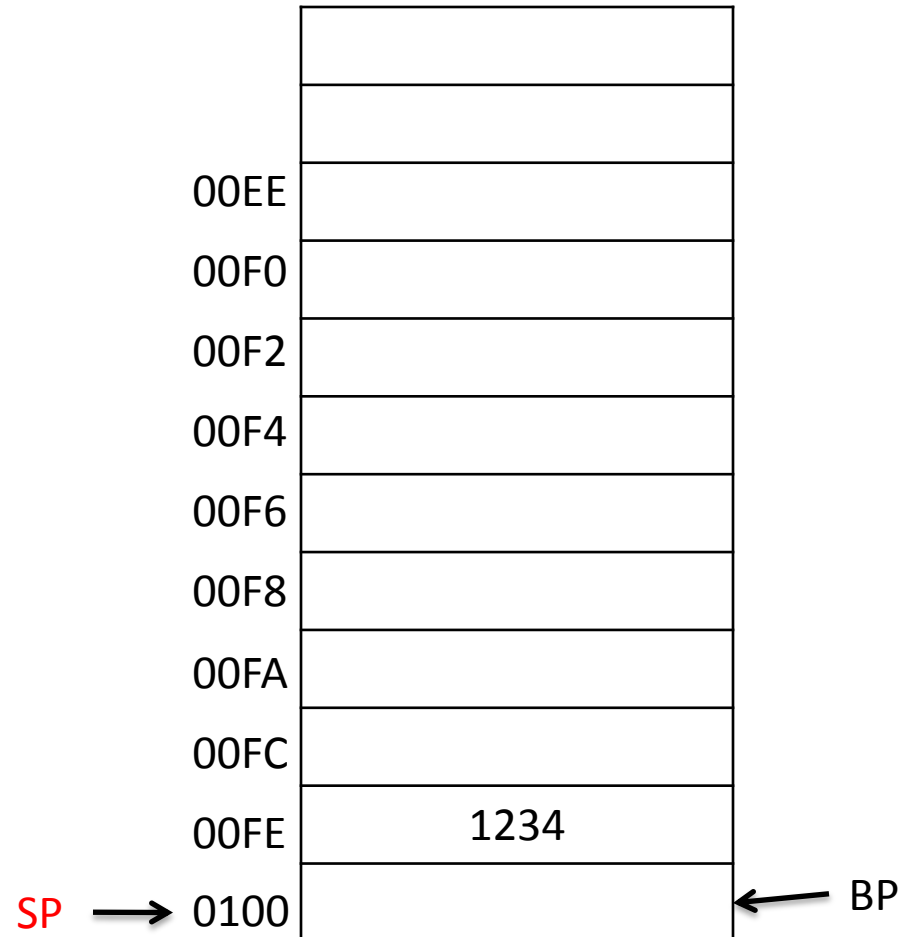
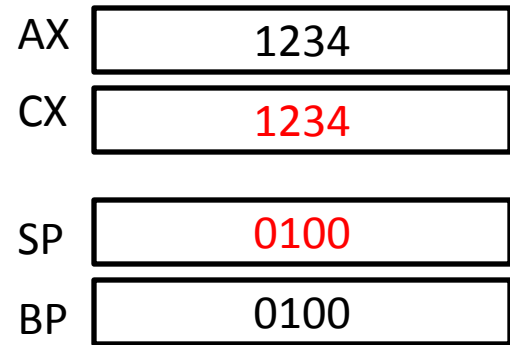


POP

POP destination

- destination is a 16-bit register(except IP register) / memory word

POP CX



PUSHF and POPF

PUSHF

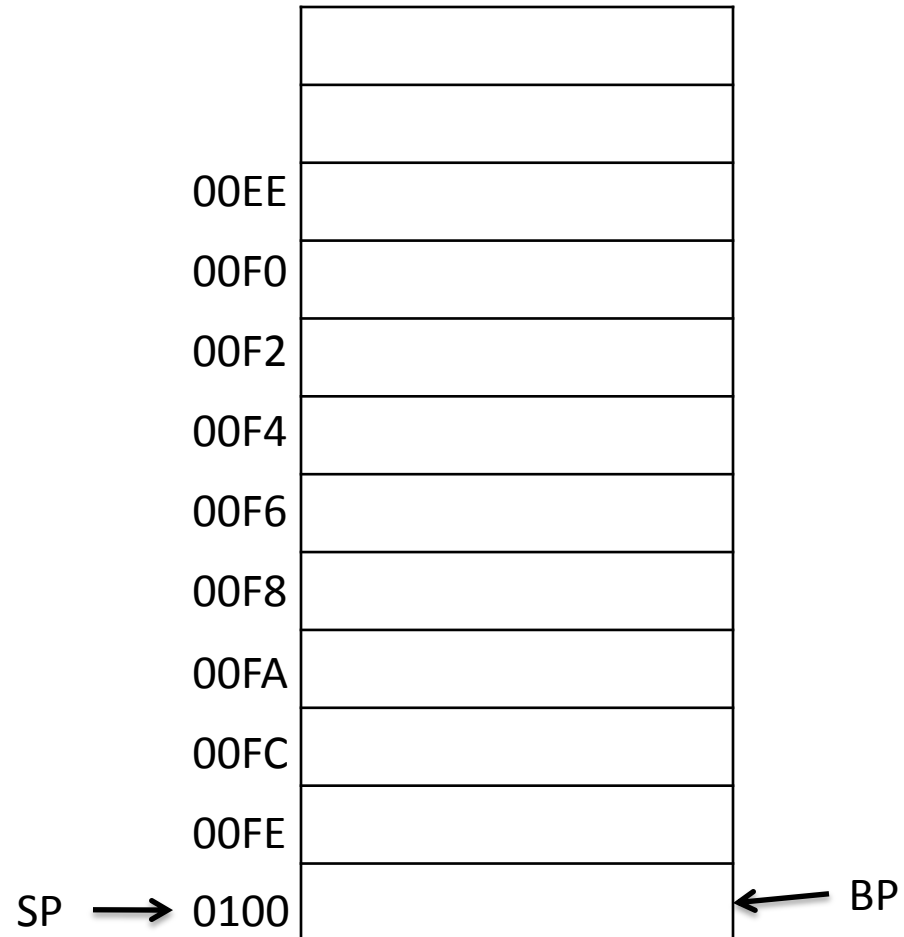
- No source operand is allowed here
- Pushes the contents of FLAGS register onto the stack

PUSHF

POPF

- Reverse of PUSHF
- Pops the top of the stack into FLAGS register

AX	1234
CX	
SP	0100
BP	0100



Order of PUSH and POP

PUSH A

PUSH B

PUSH C

...

...

POP C

POP B

POP A

Procedure Declaration

```
name    PROC    type  
; body of the procedure  
        RET  
name    ENDP
```

“name” is the user-defined name of the procedure

“type” is optional, can be “FAR”/”NEAR”

“RET” causes control back to the calling procedure

name PROC type
; body of the procedure
RET
name ENDP

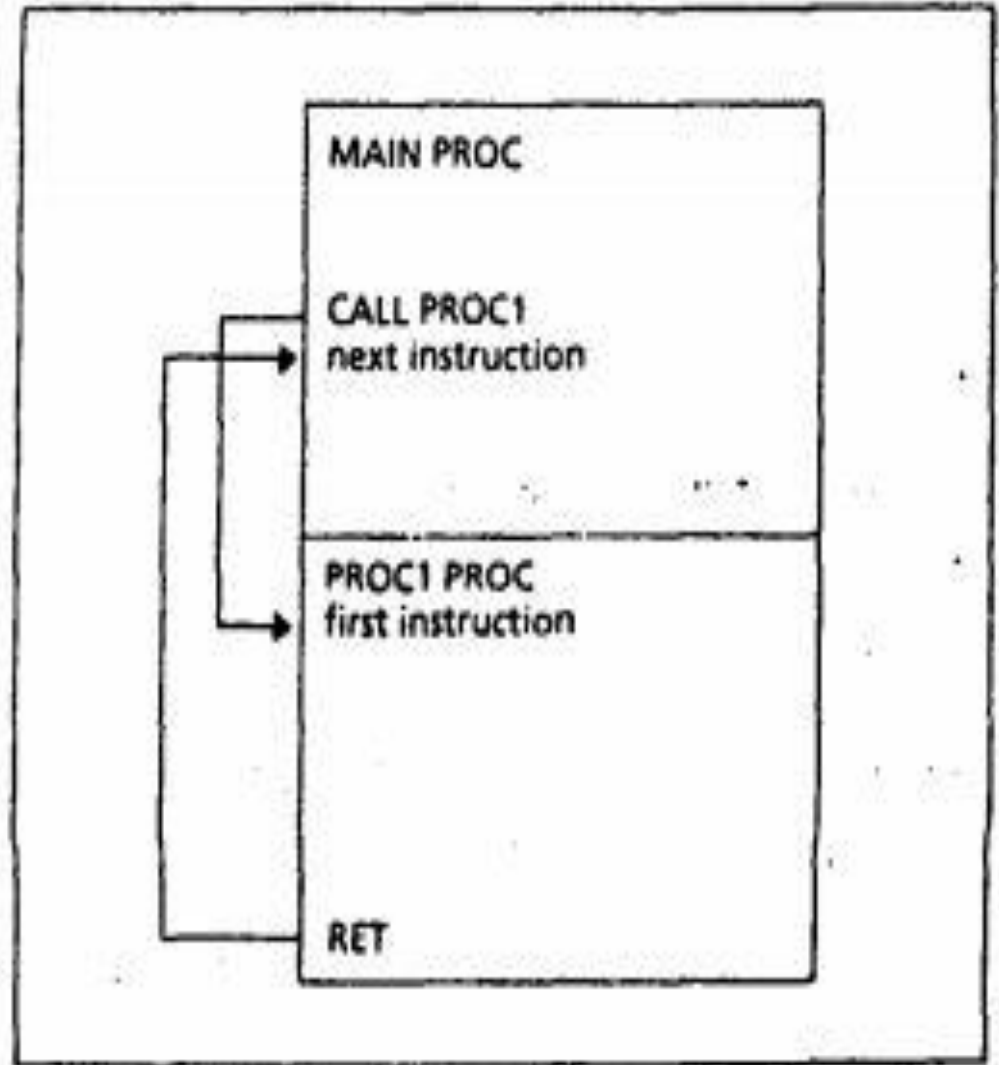


Fig: Procedure CALL and RET

CALL

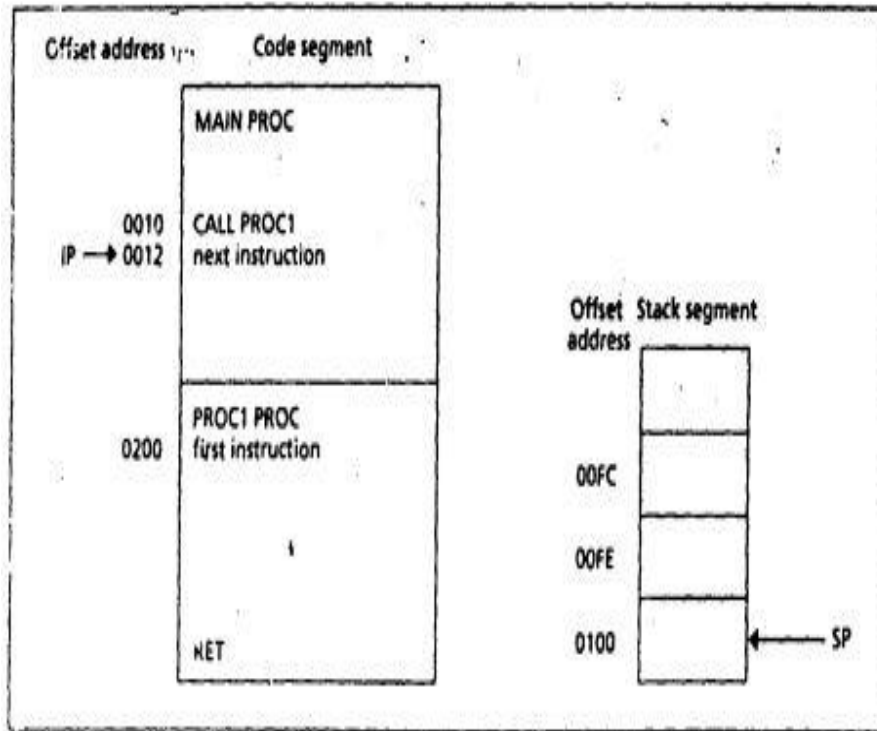


Figure 8.4A Before CALL

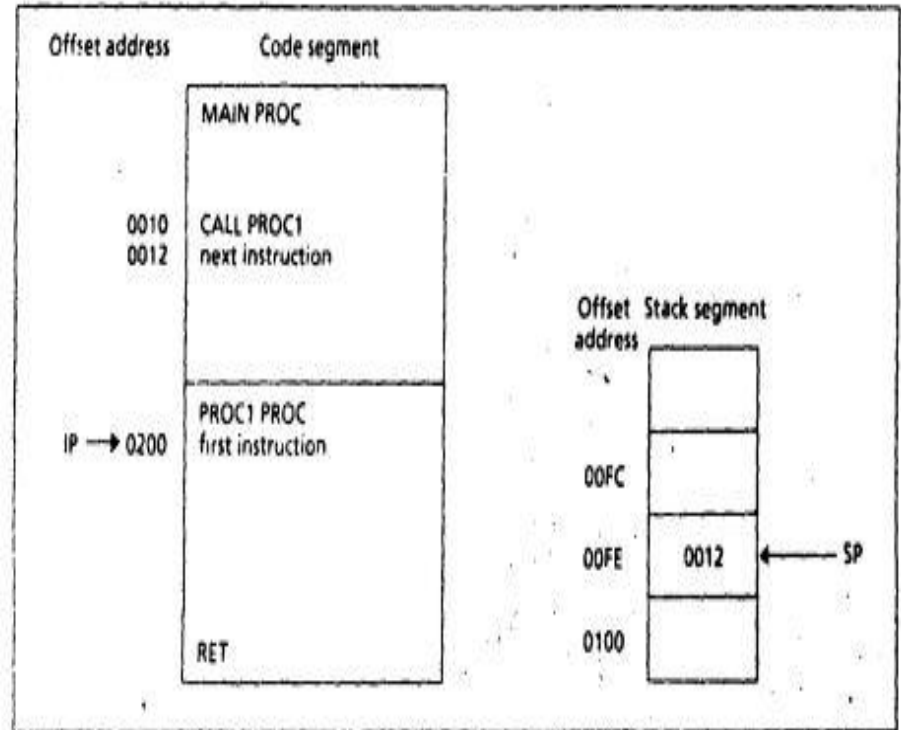


Figure 8.4B After CALL

RET

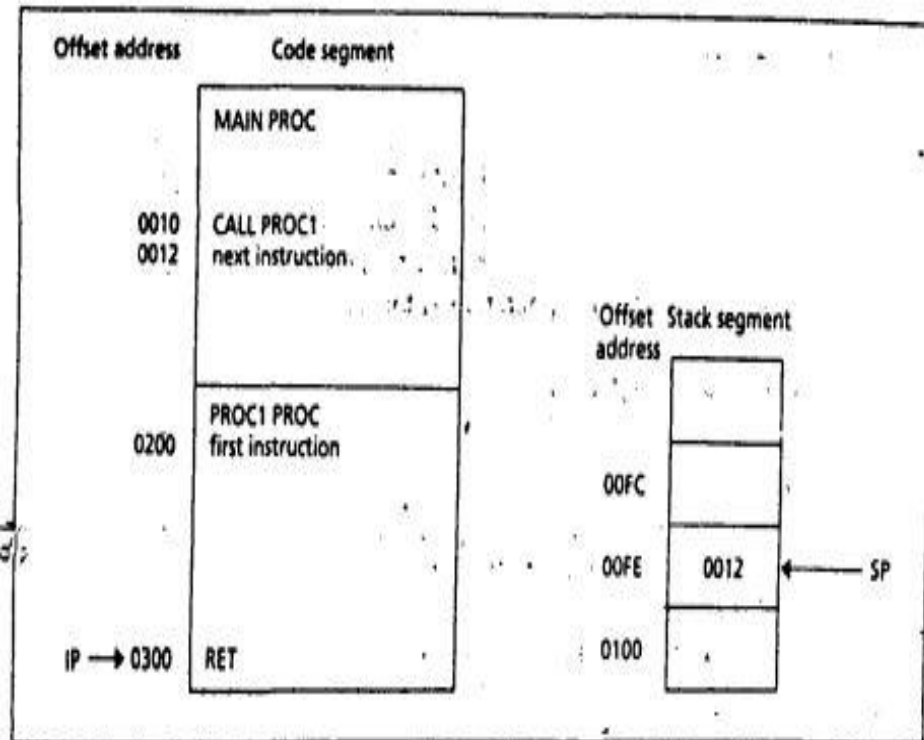


Figure 8.5A Before RET

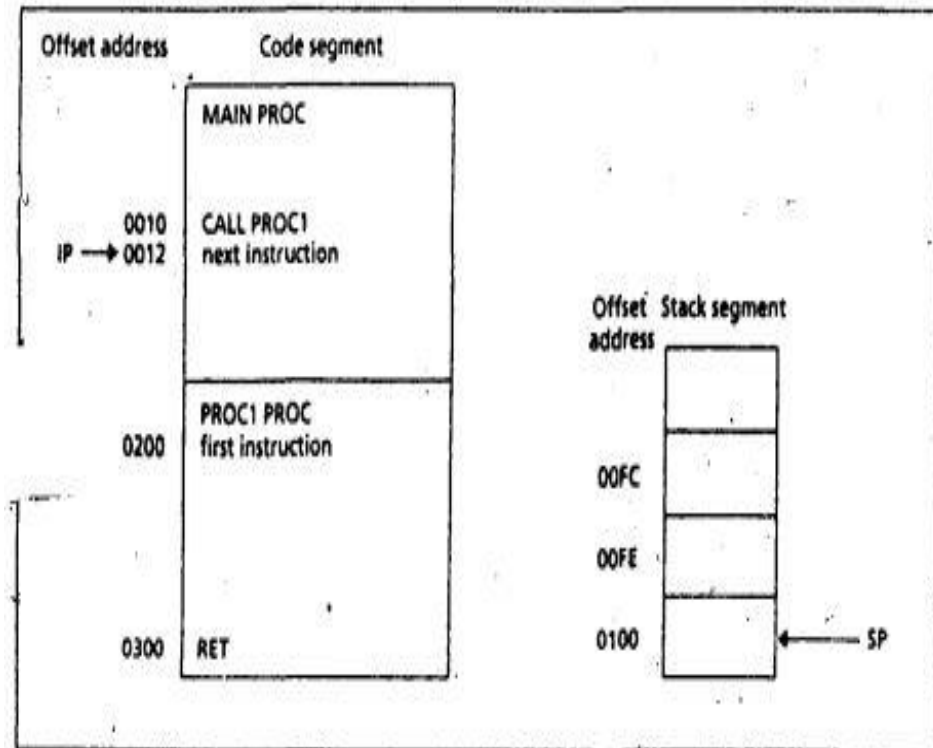


Figure 8.5B After RET

Chapter 17

Recursion

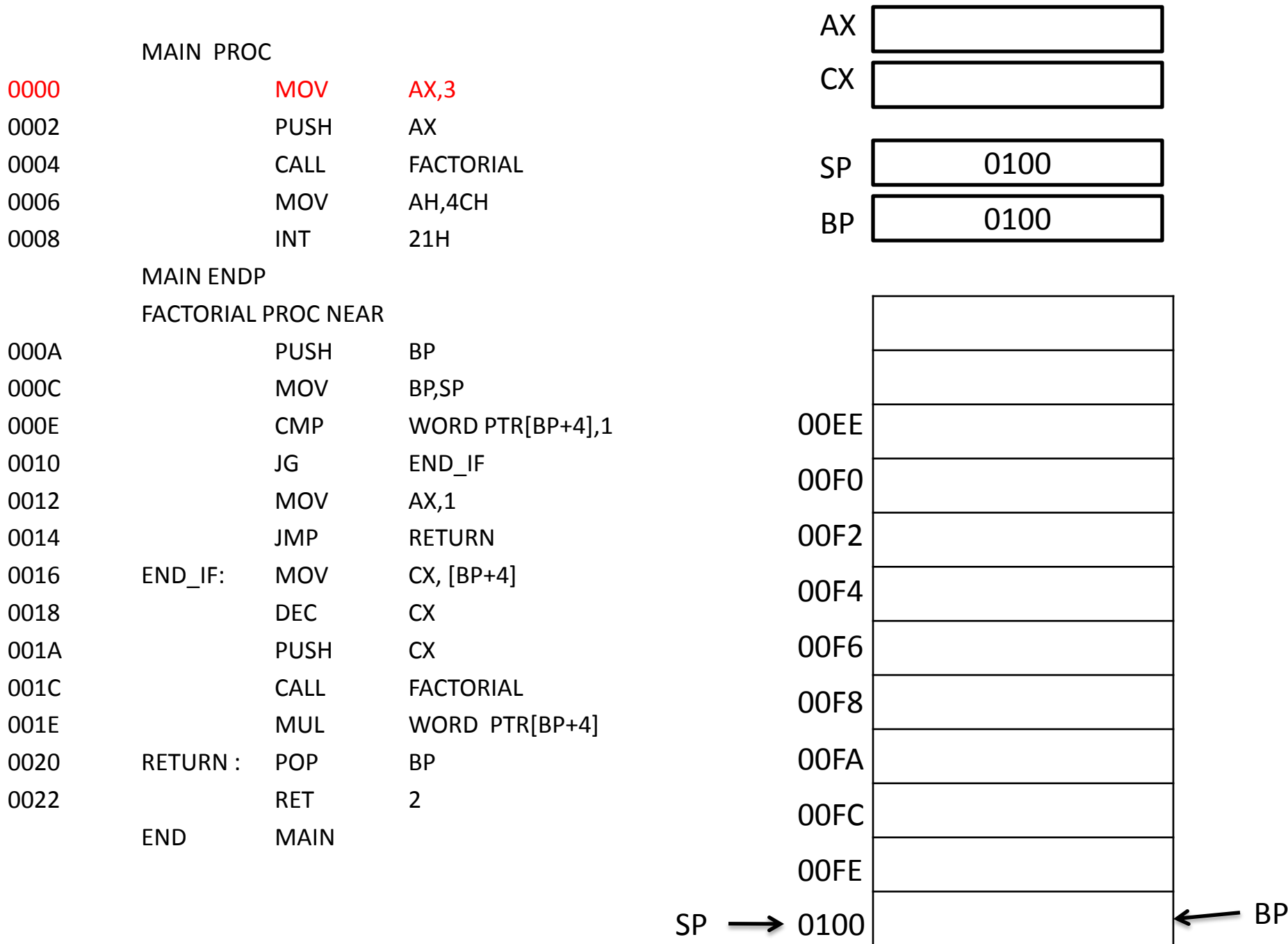
Activation Record

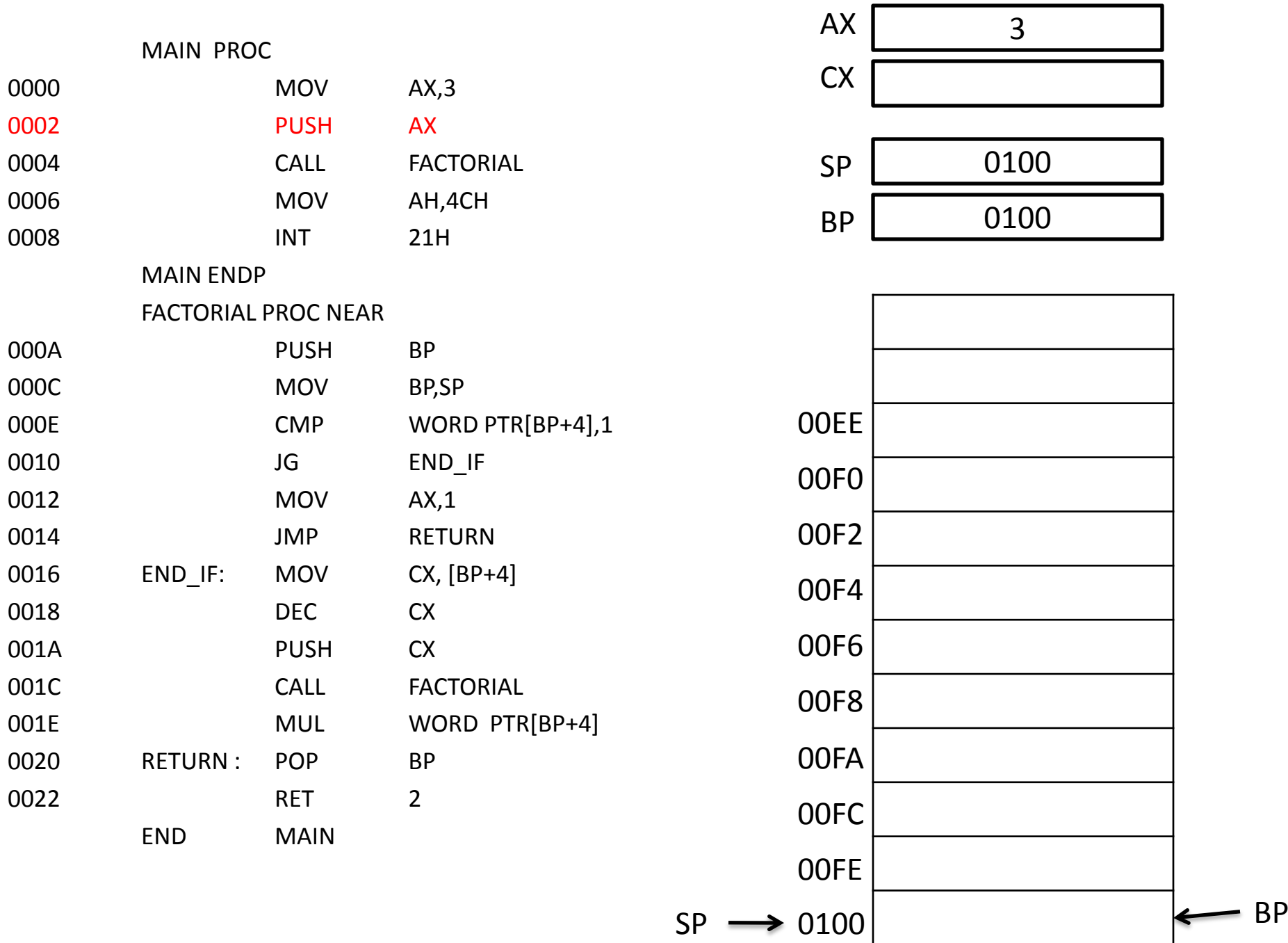
- Record to be stored before CALLing a recursive function

Factorial

- $\text{Factorial}(n) = n! = n * (n-1) * (n-2) * \dots * 3 * 2 * 1$

$$\text{Factorial}(n) = n * \text{Factorial}(n-1)$$

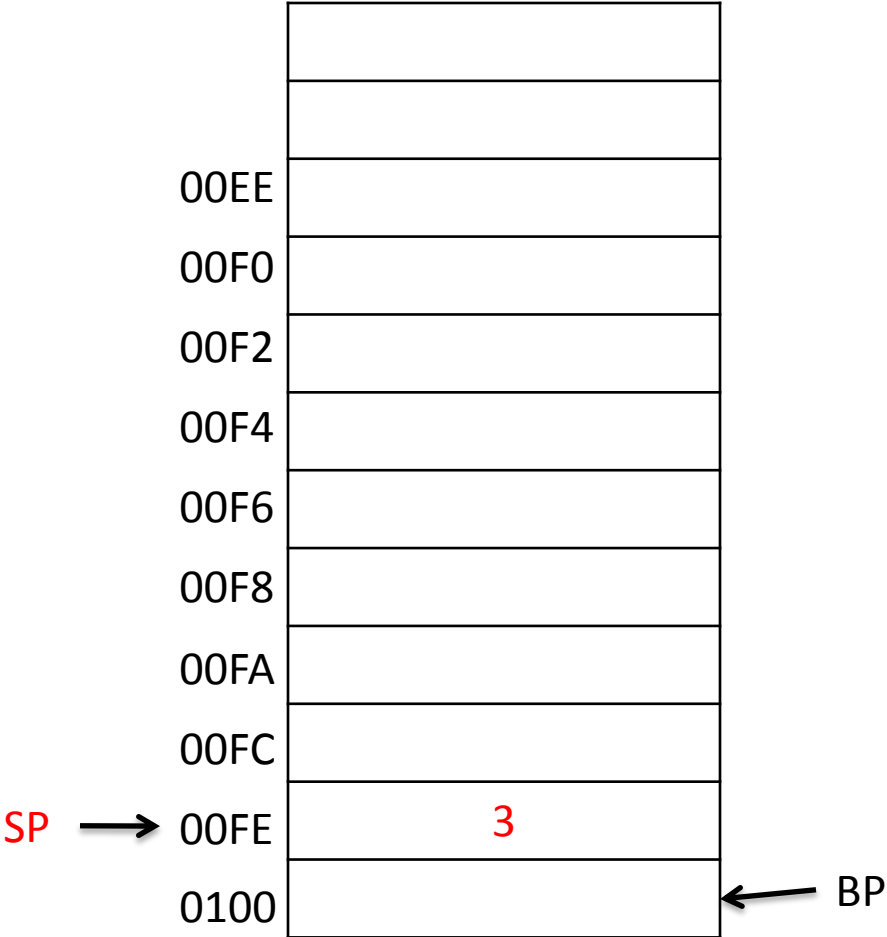
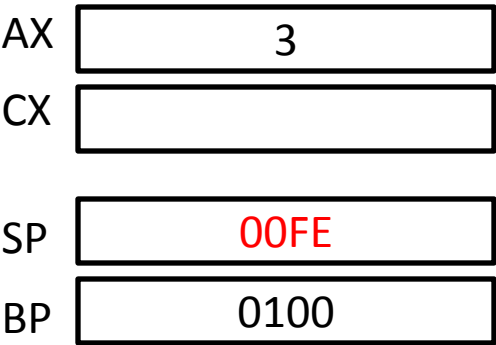




```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

MAIN ENDP
FACTORIAL PROC NEAR

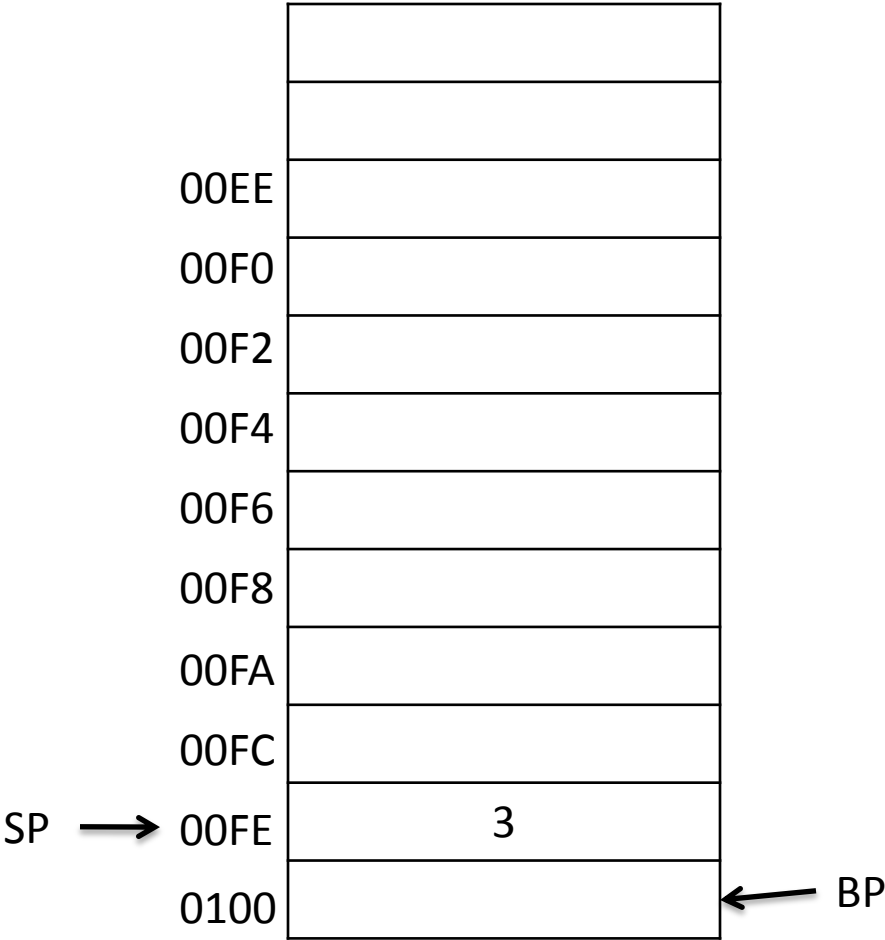
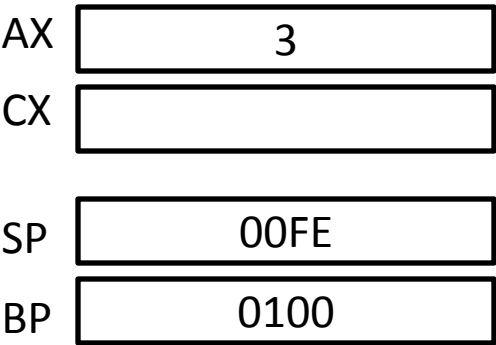
```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

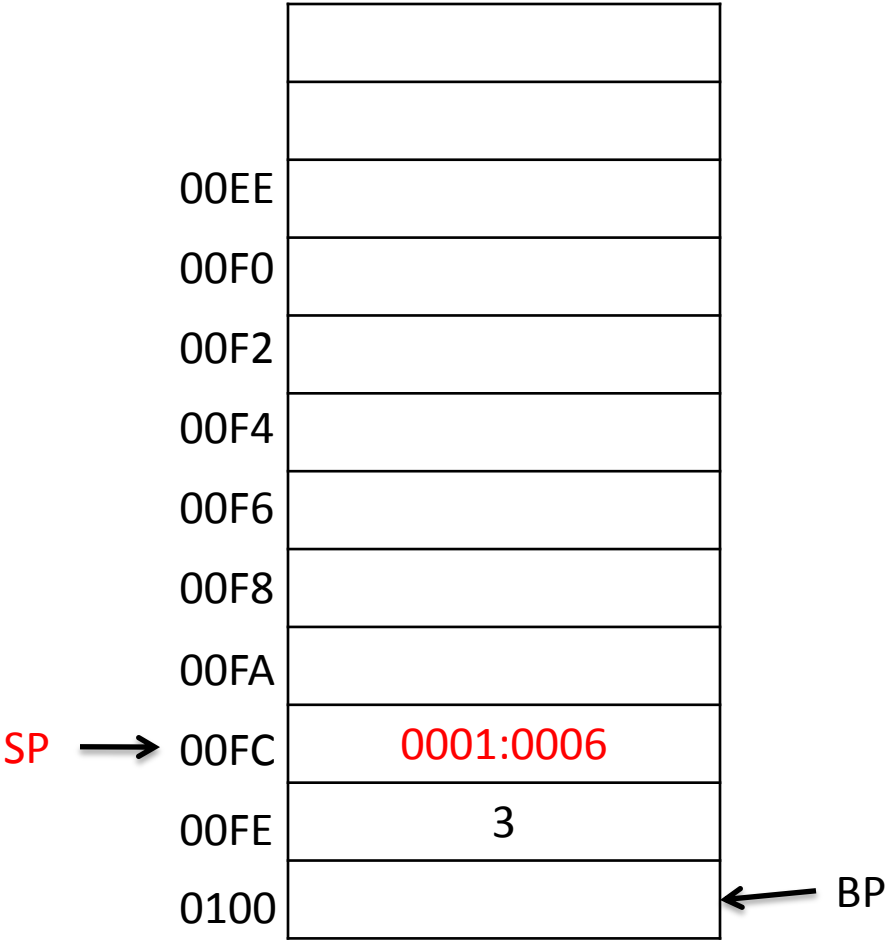
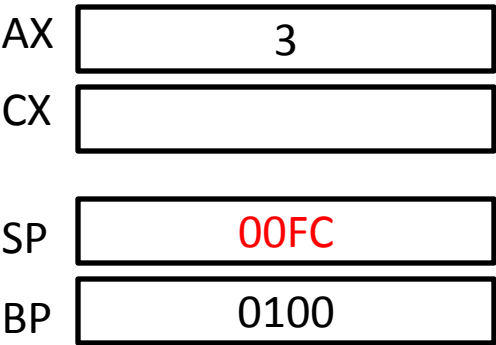
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

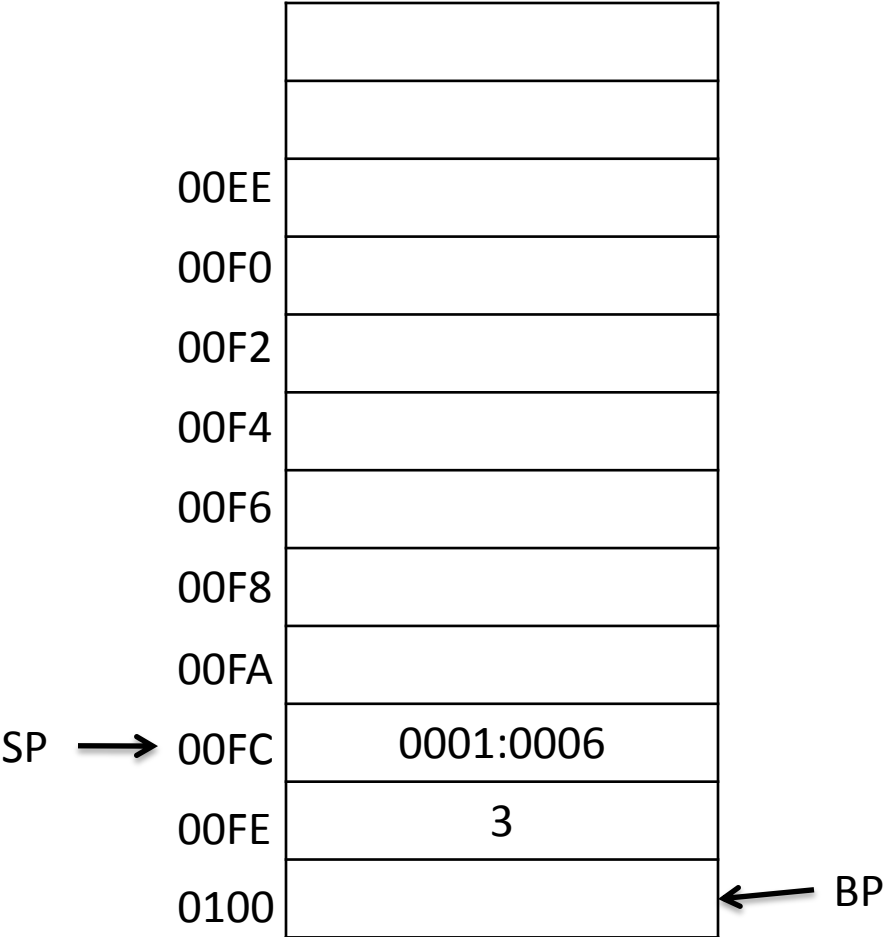
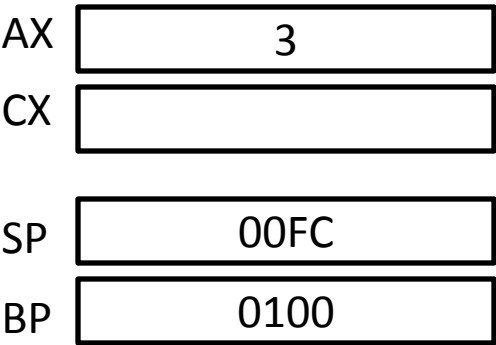
END       MAIN
```




```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX, [BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

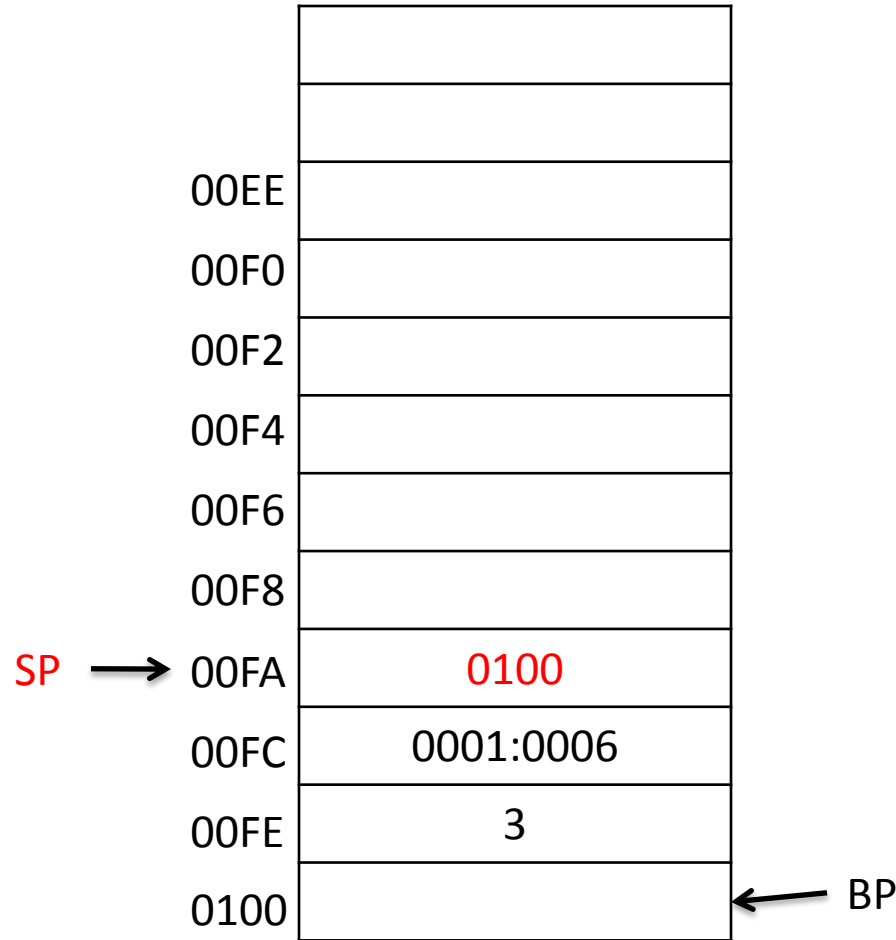
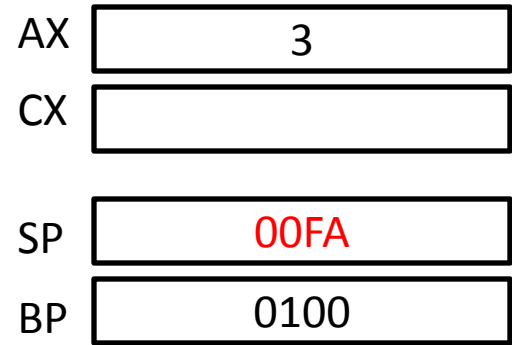
END      MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX, [BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

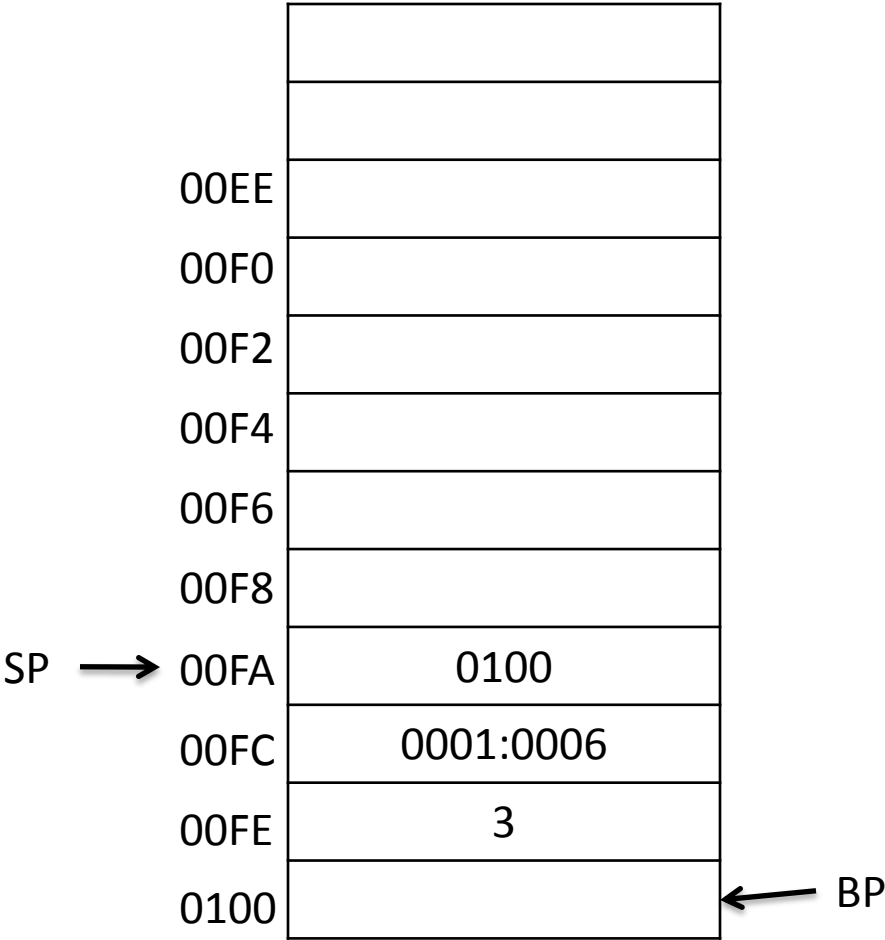
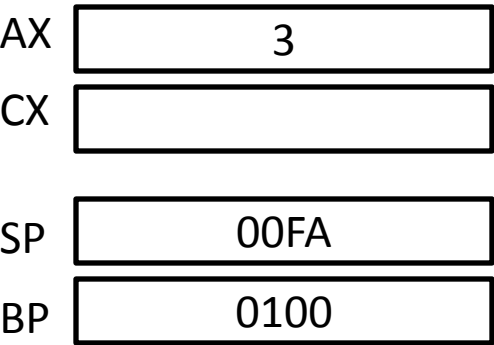
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

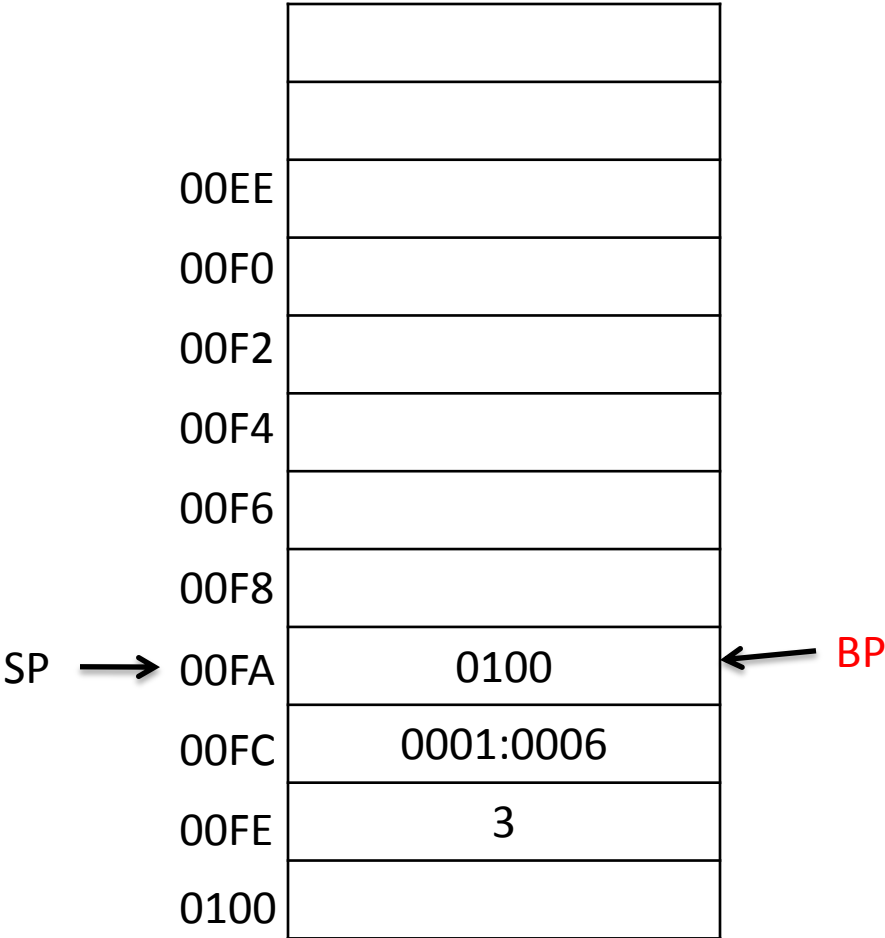
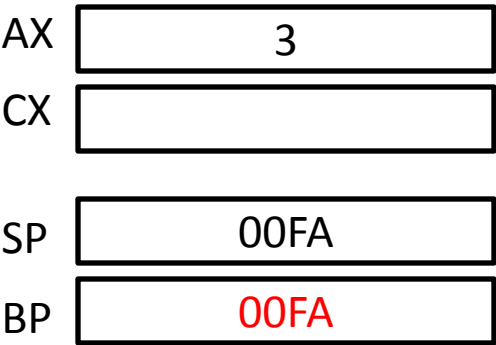
END      MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX, [BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

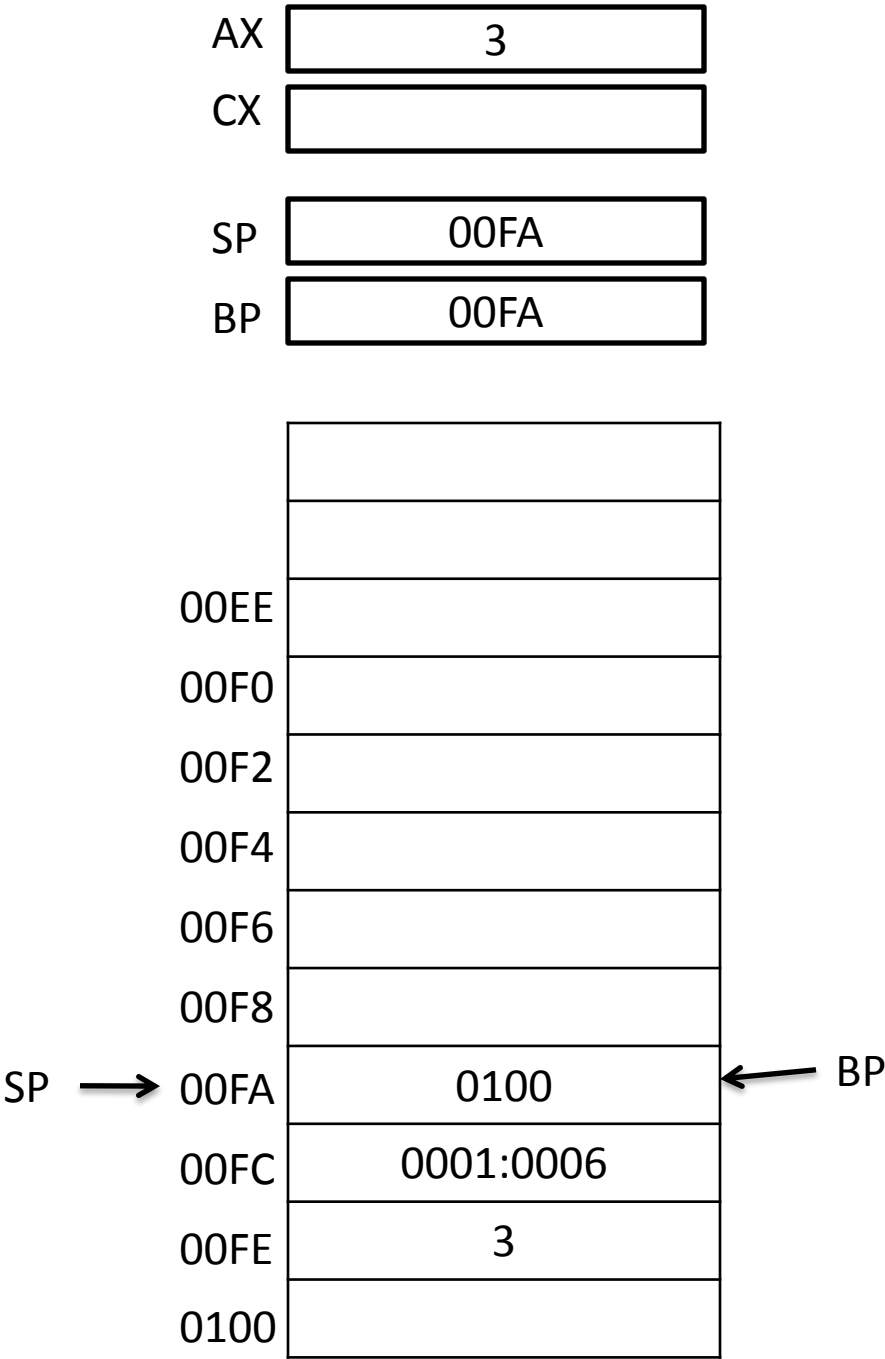
END      MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX, [BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

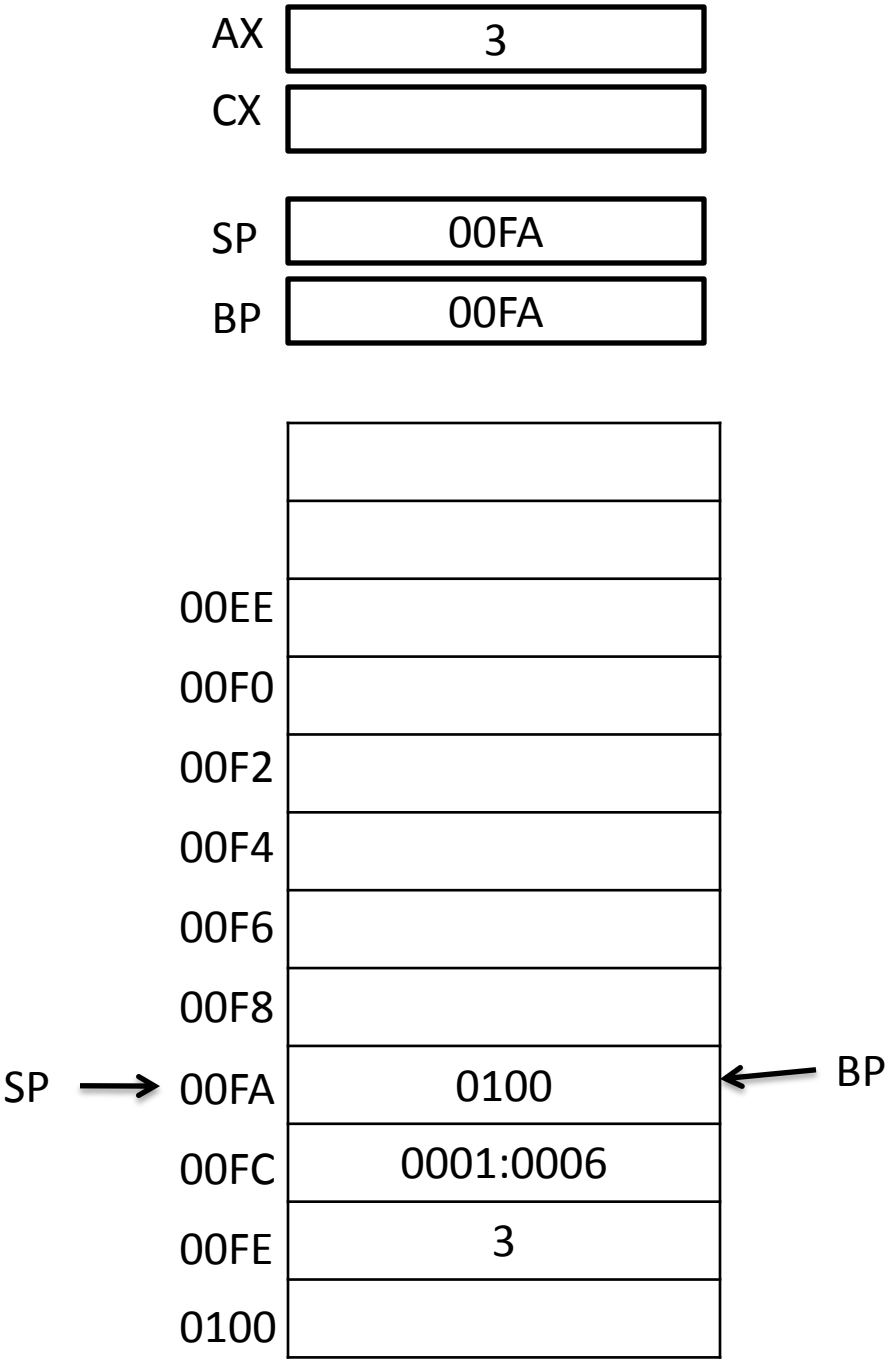
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

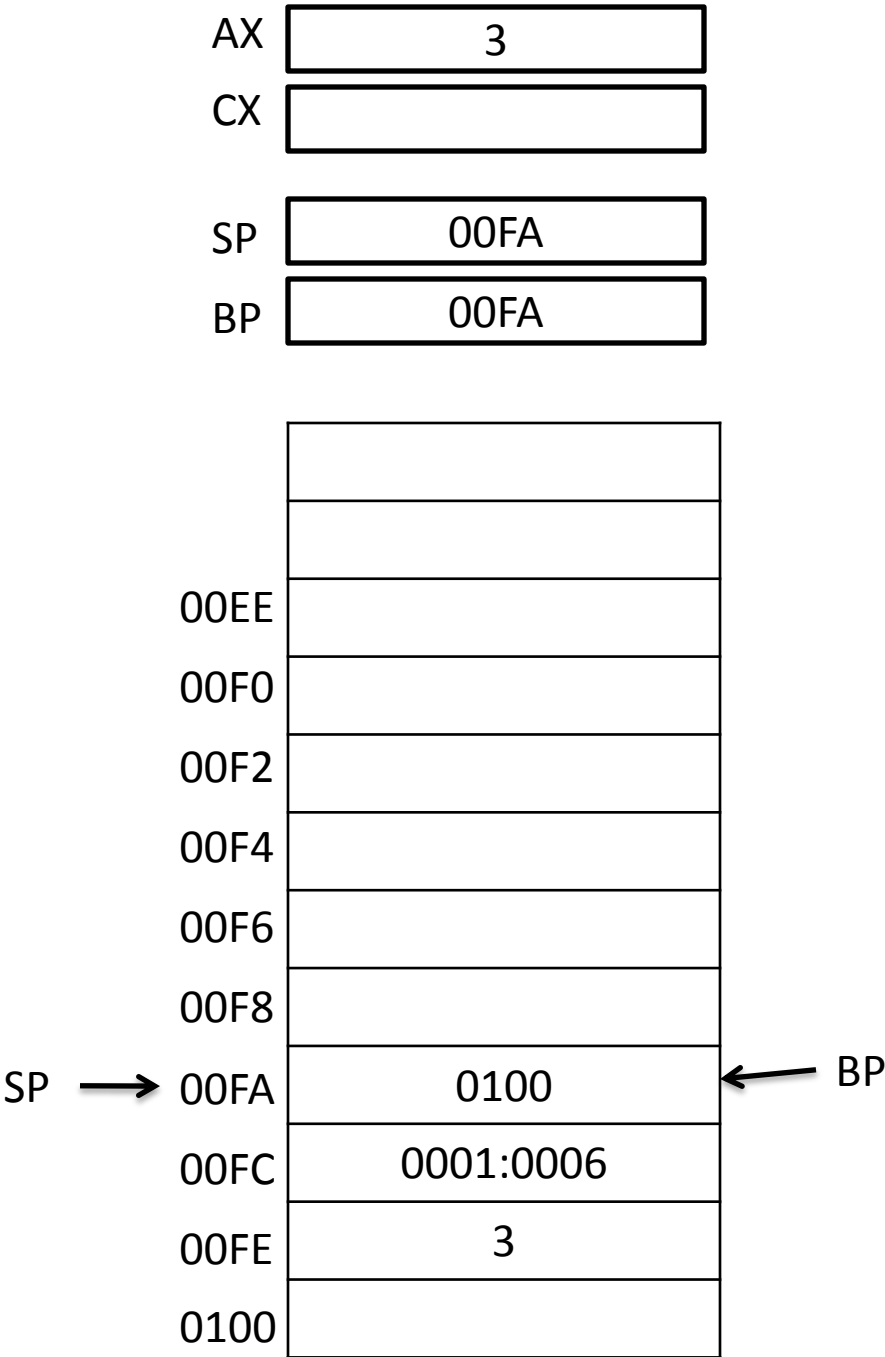
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

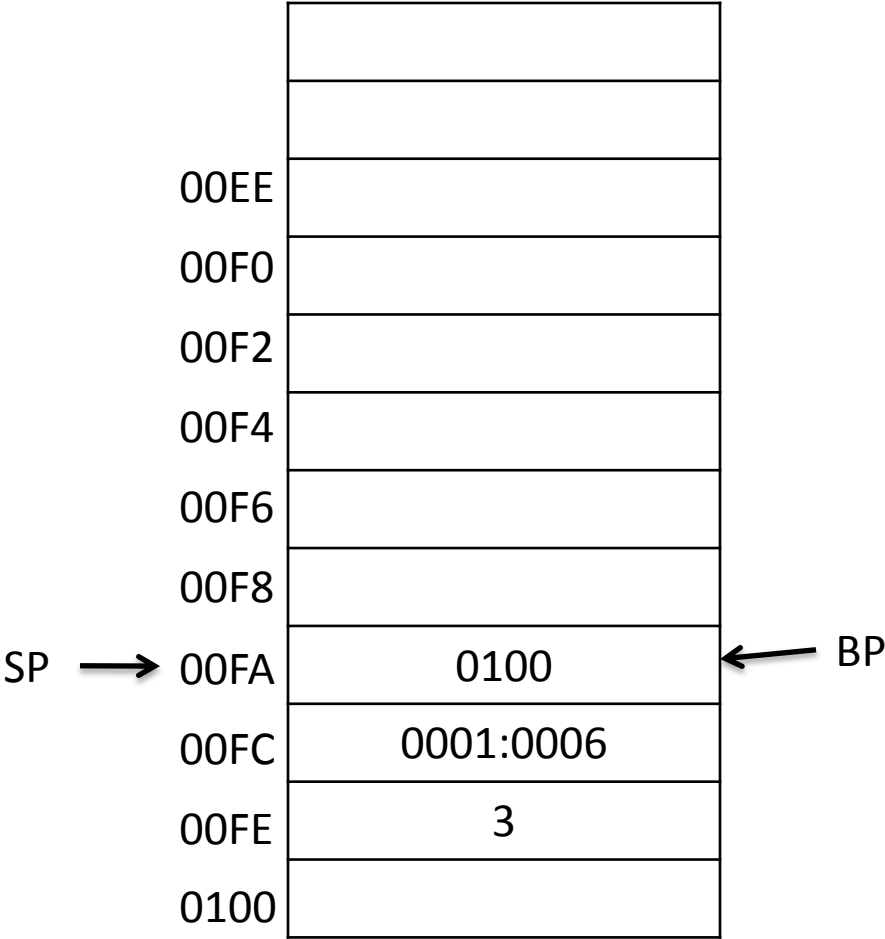
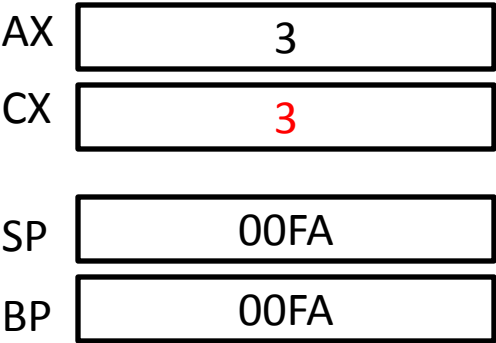
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

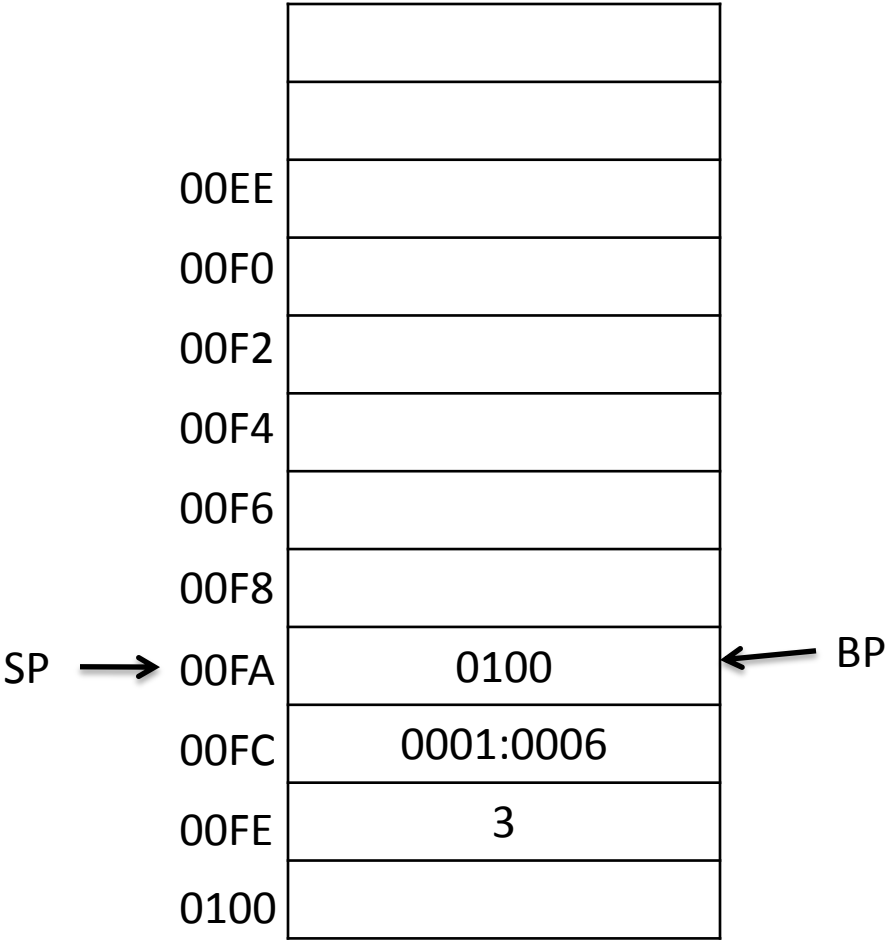
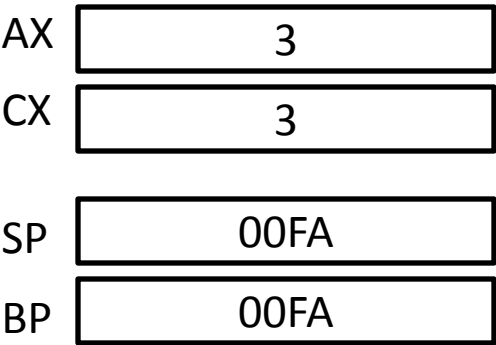
END      MAIN
```




```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

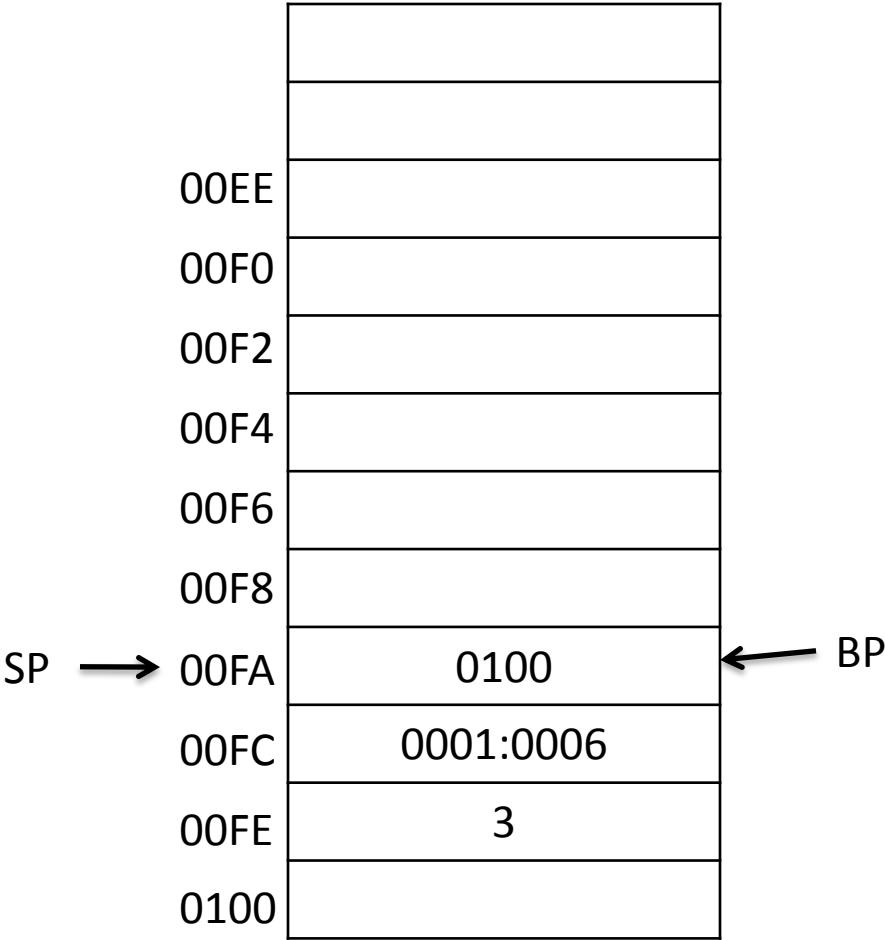
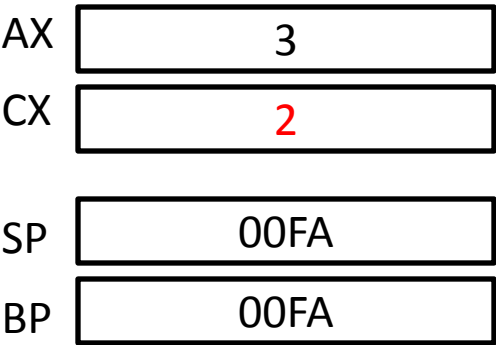
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

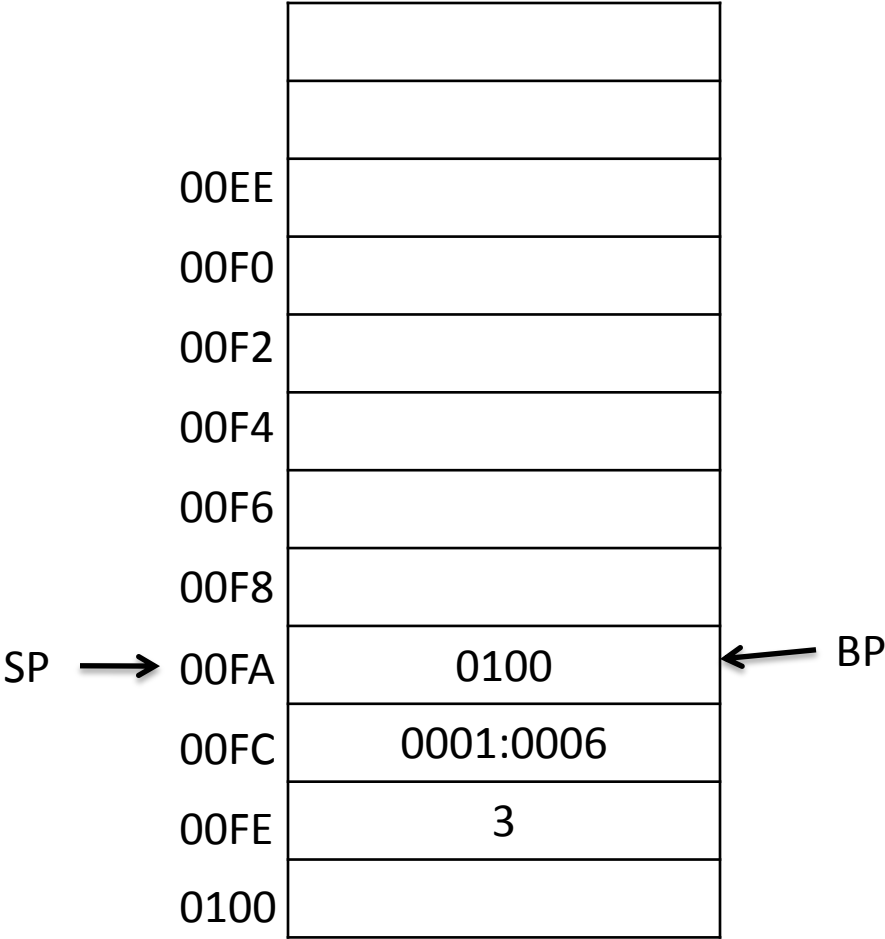
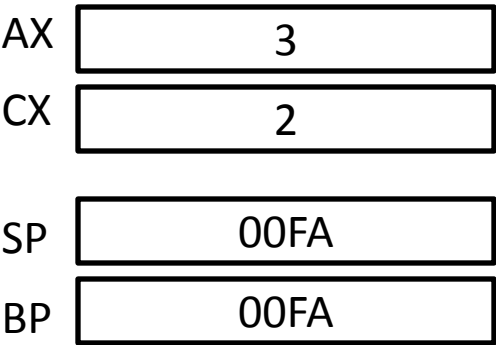
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX, [BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

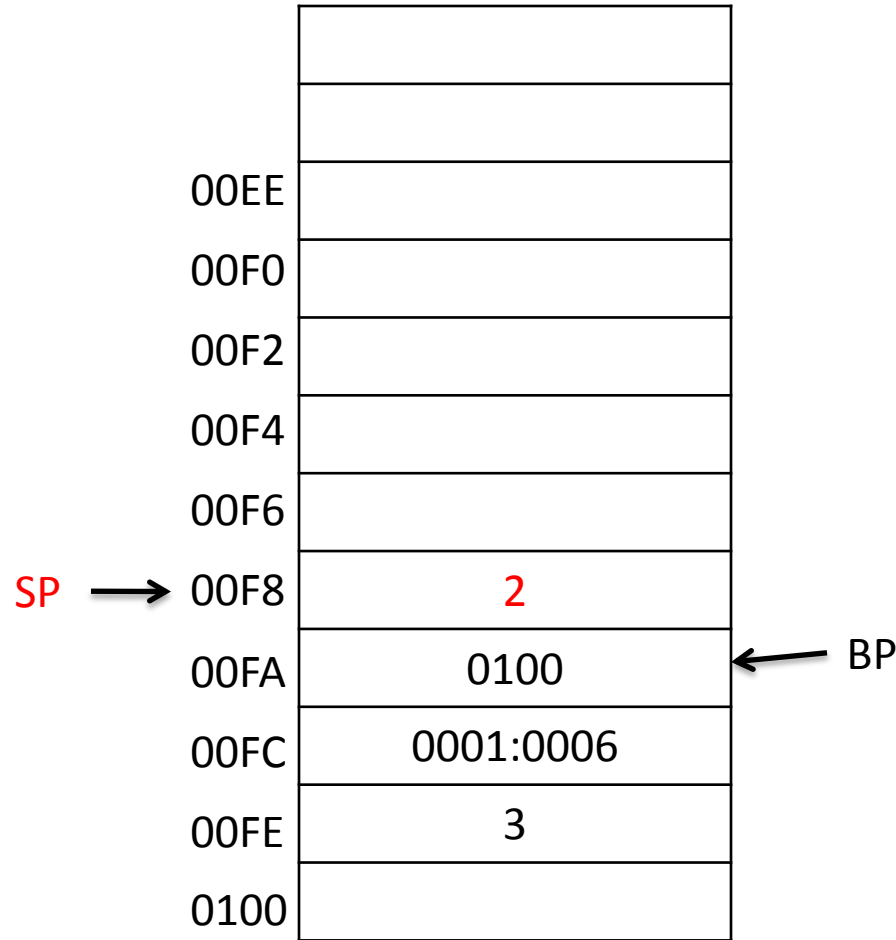
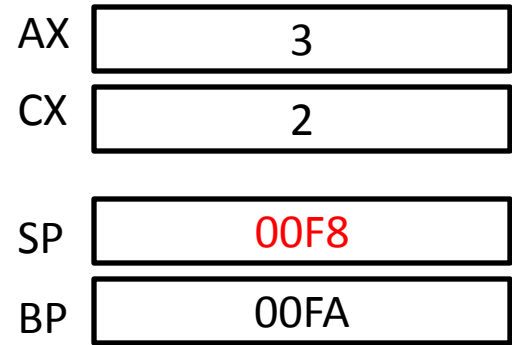
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

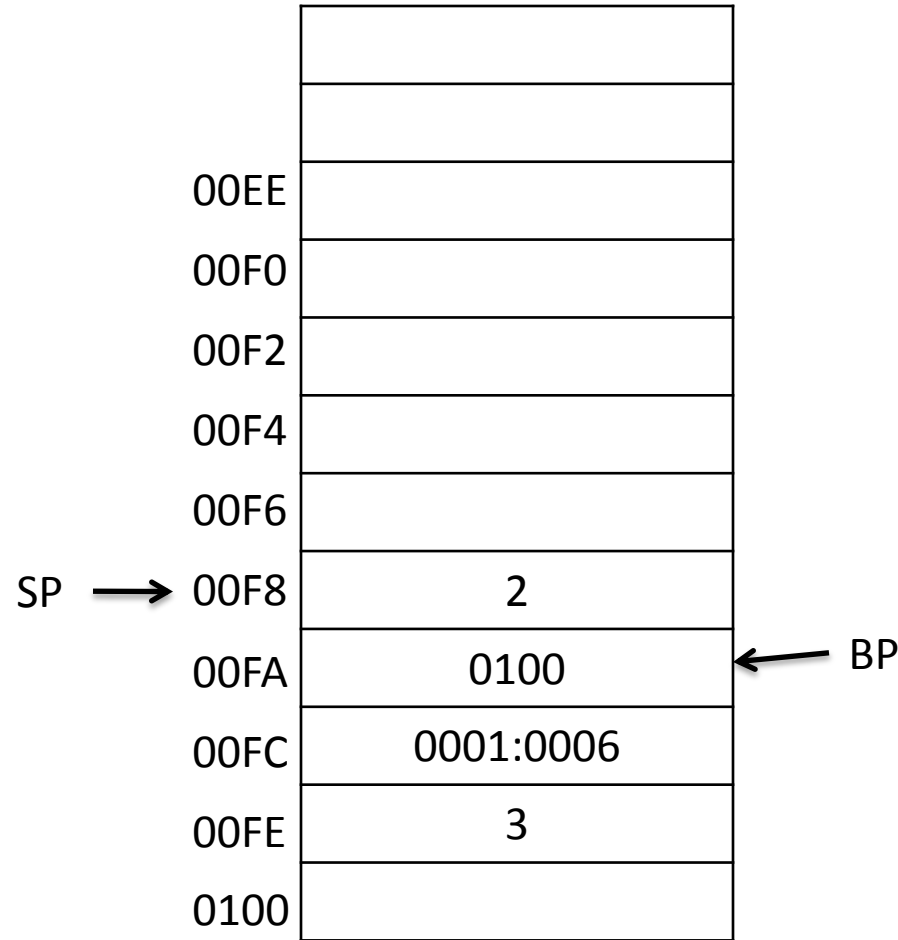
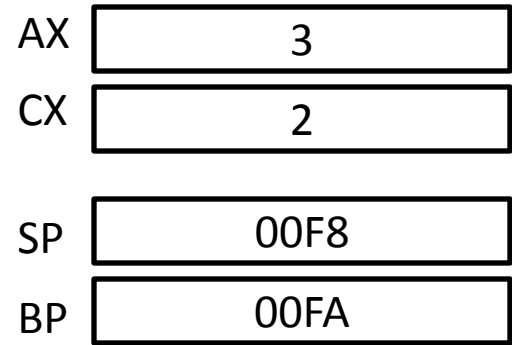
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C  CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN
```



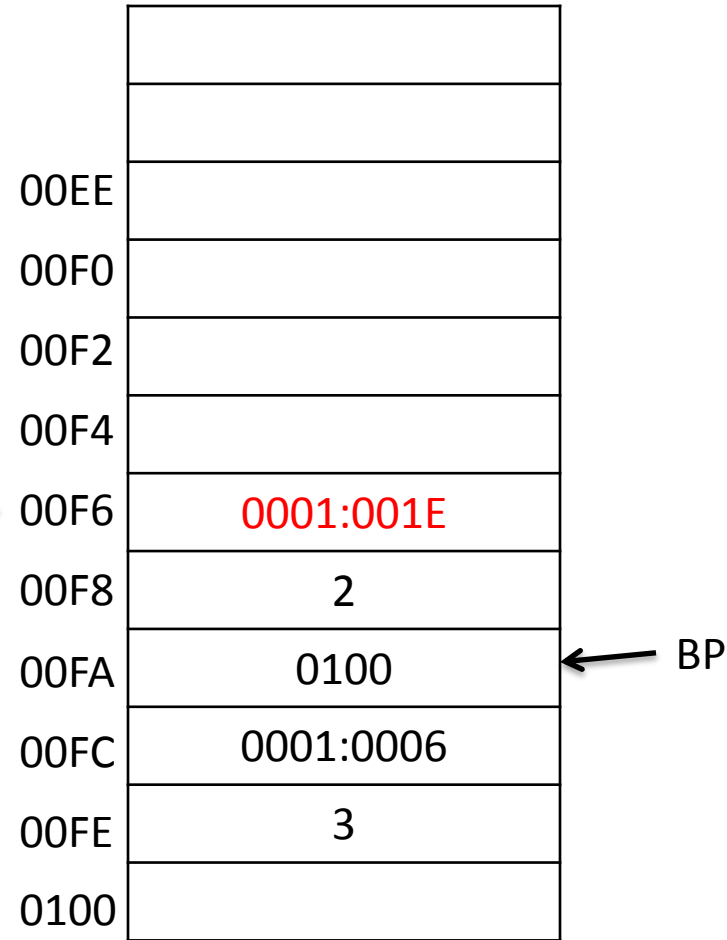
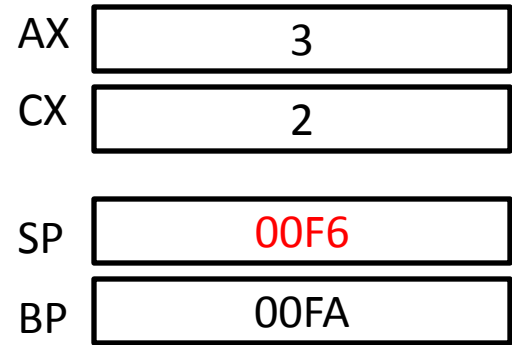
```

MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END       MAIN

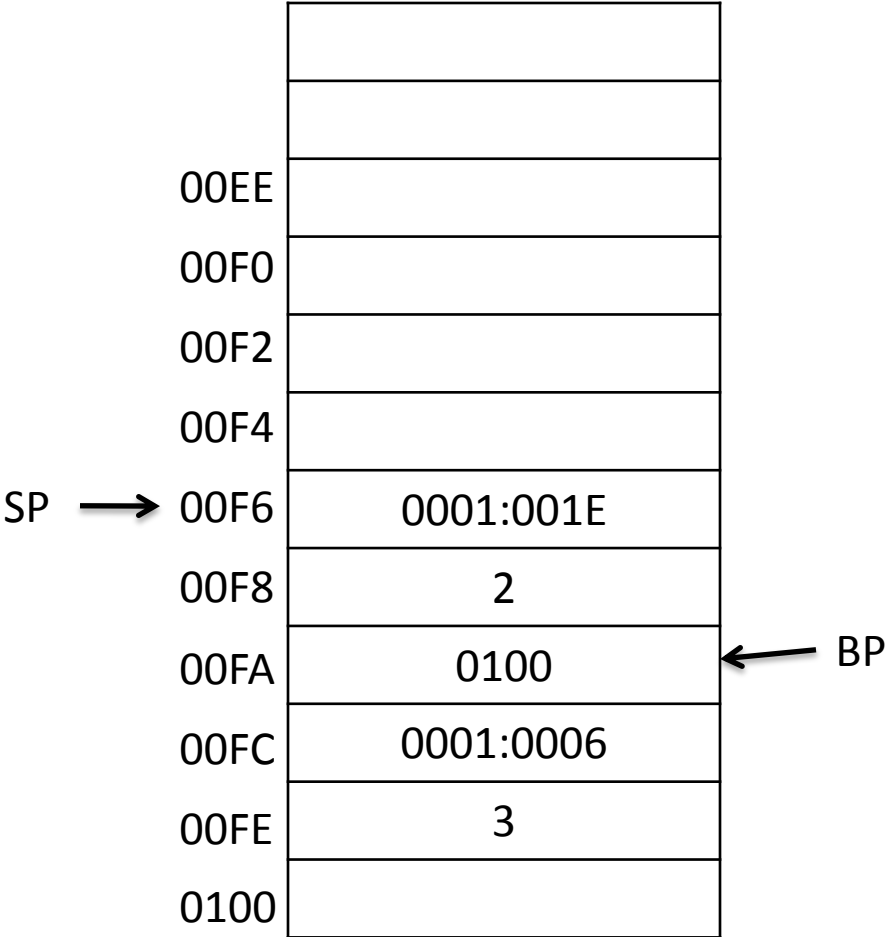
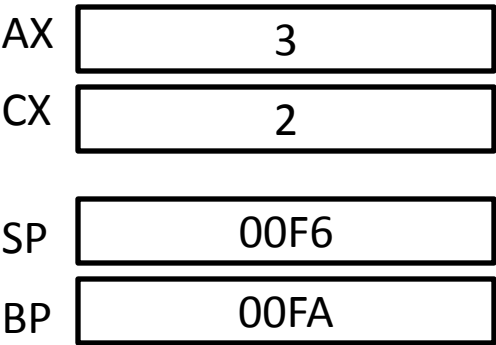
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

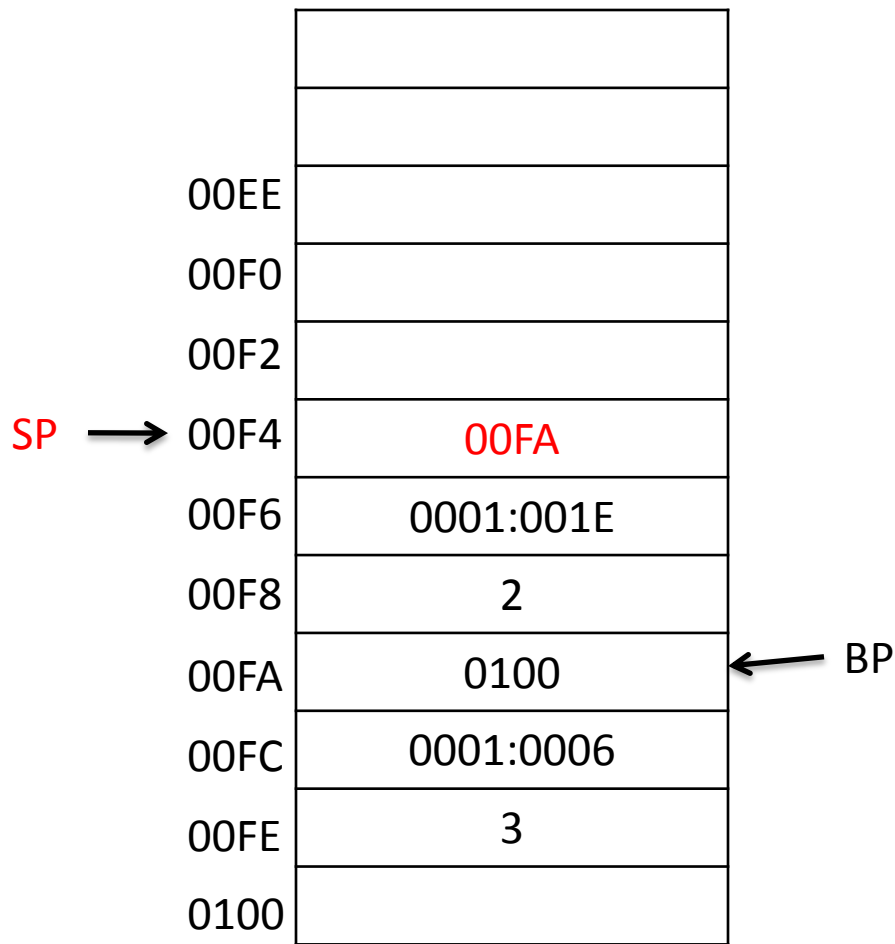
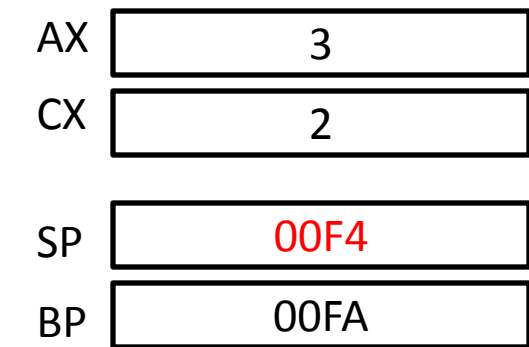
```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX,[BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

END      MAIN
```

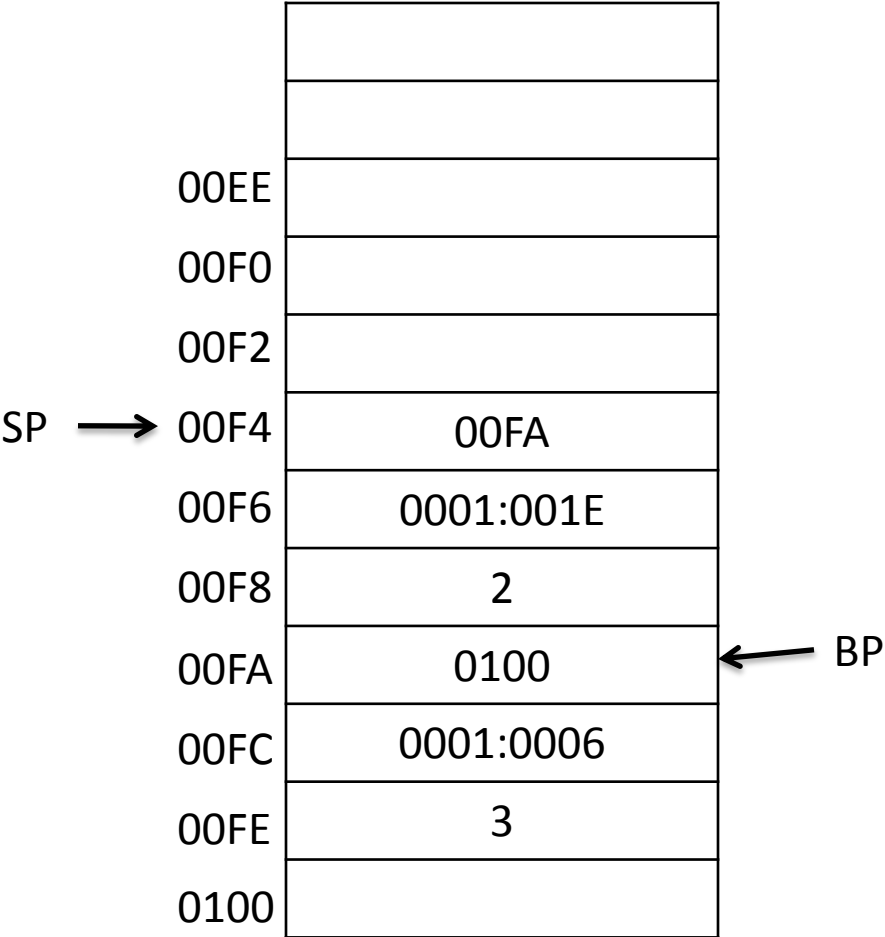
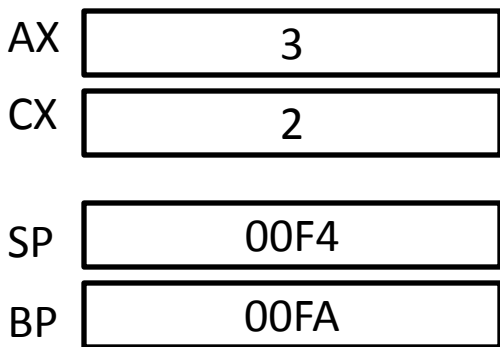



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

MAIN ENDP

FACTORIAL PROC NEAR

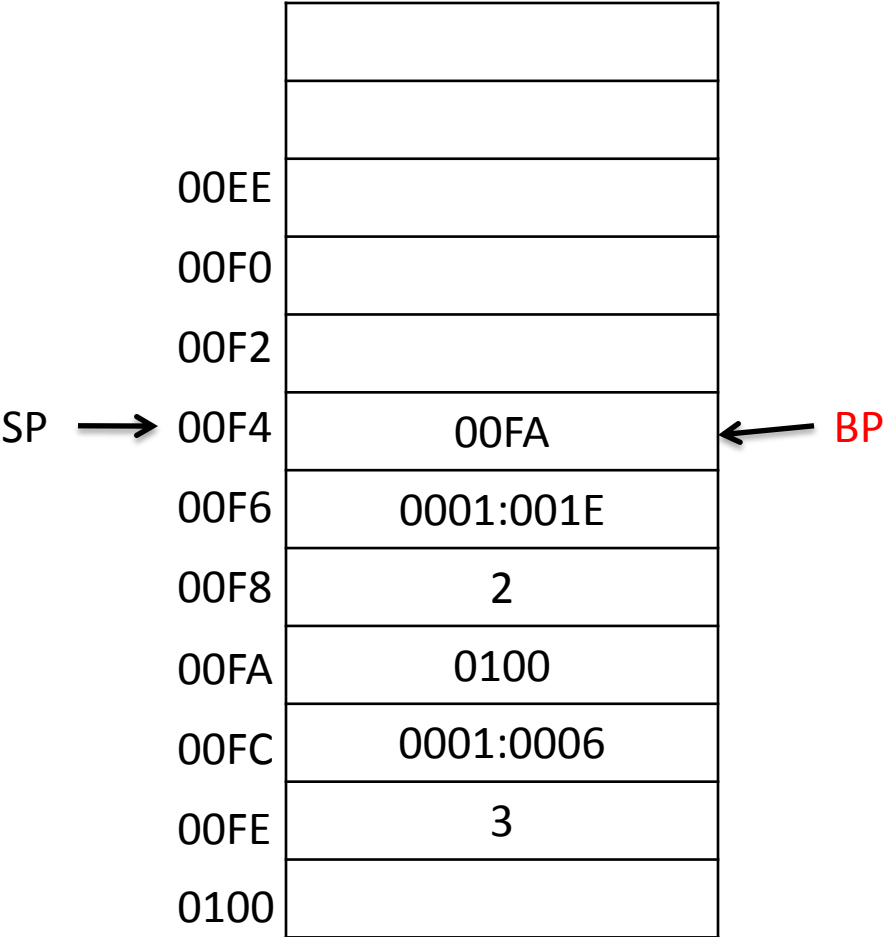
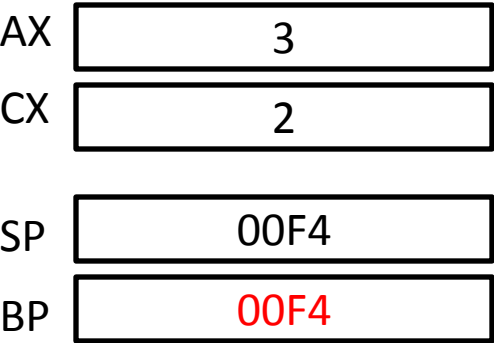
```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX,[BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

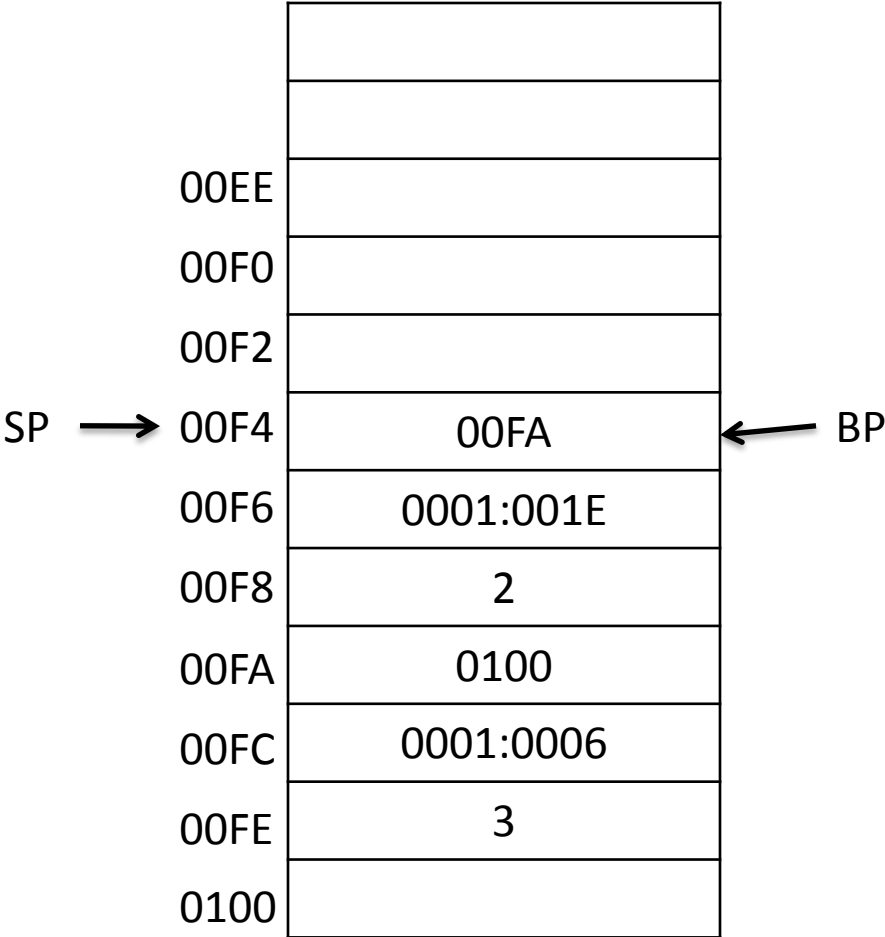
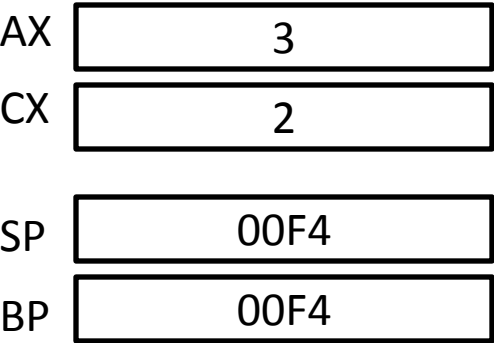
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

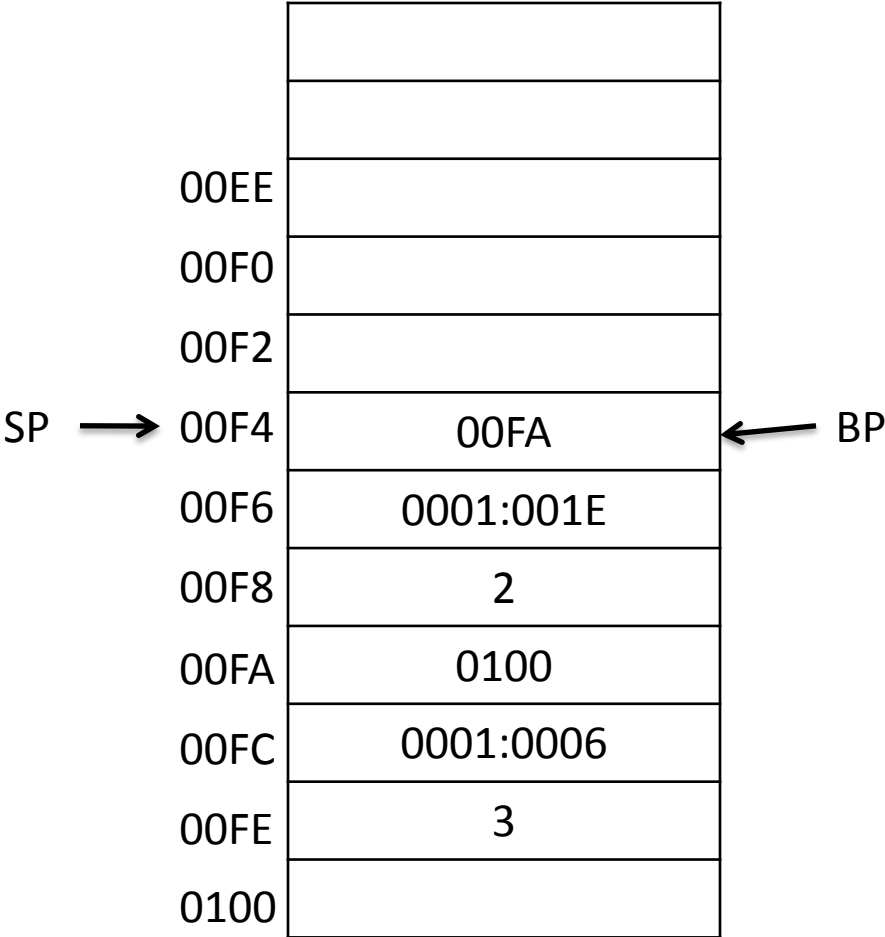
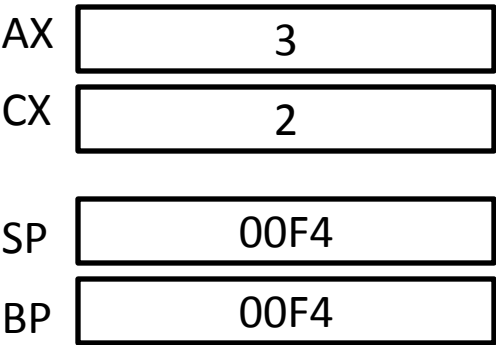
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

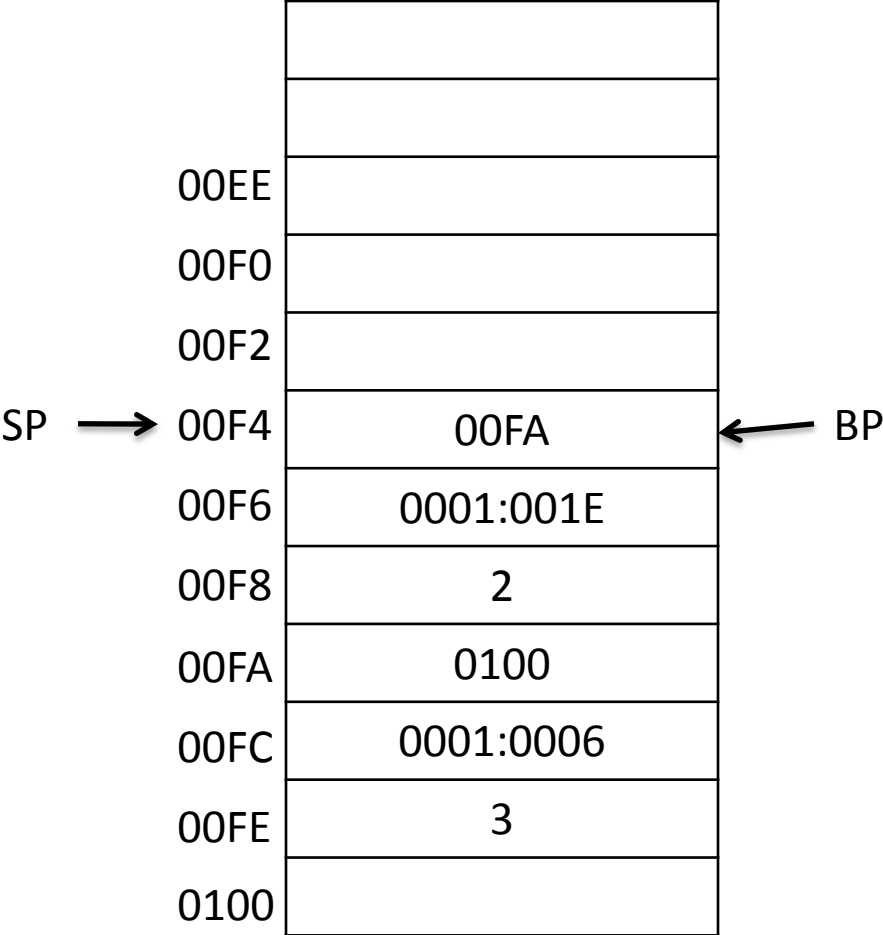
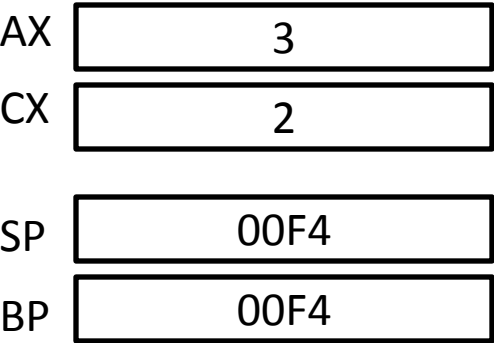
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

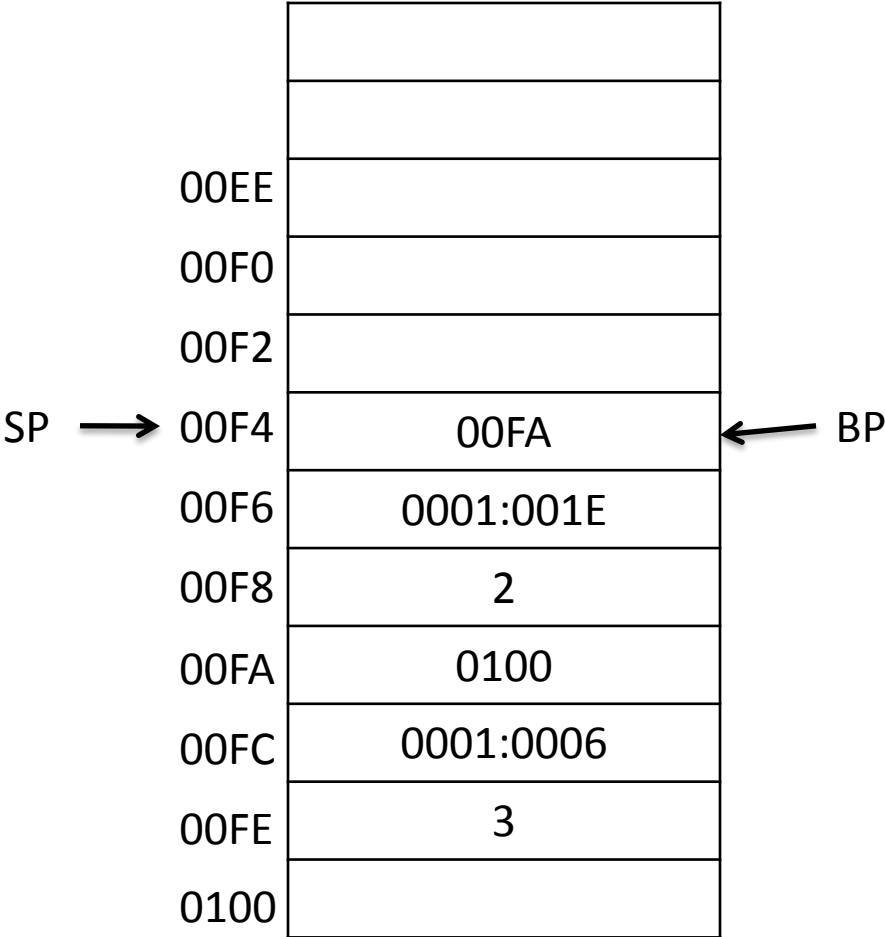
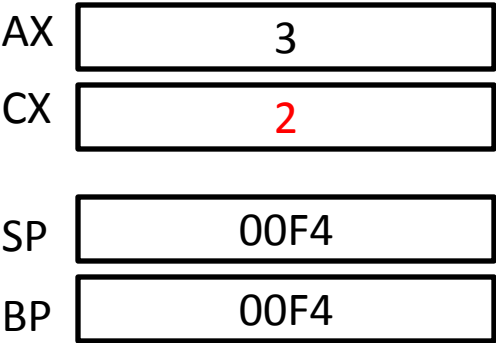
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

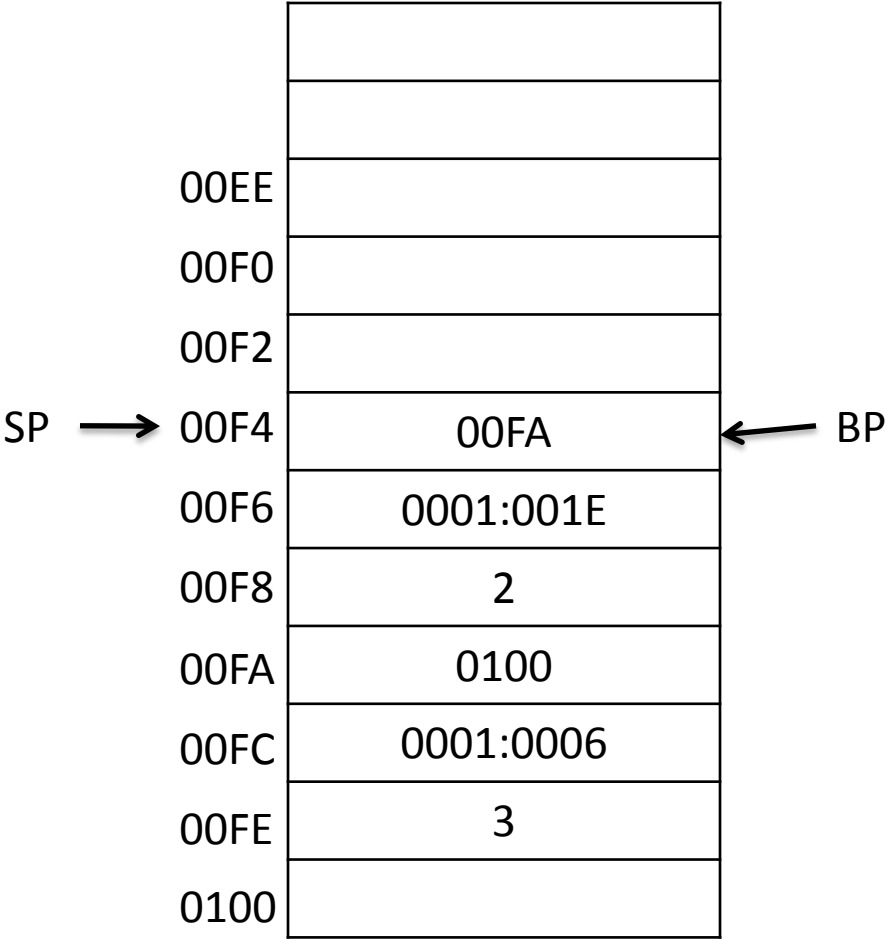
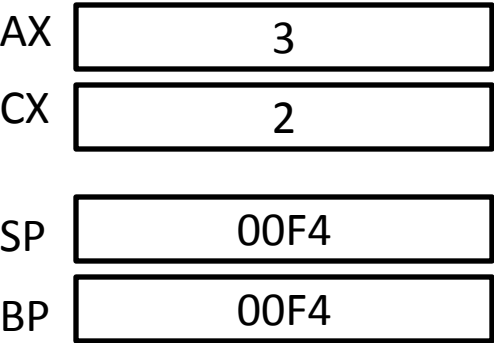
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

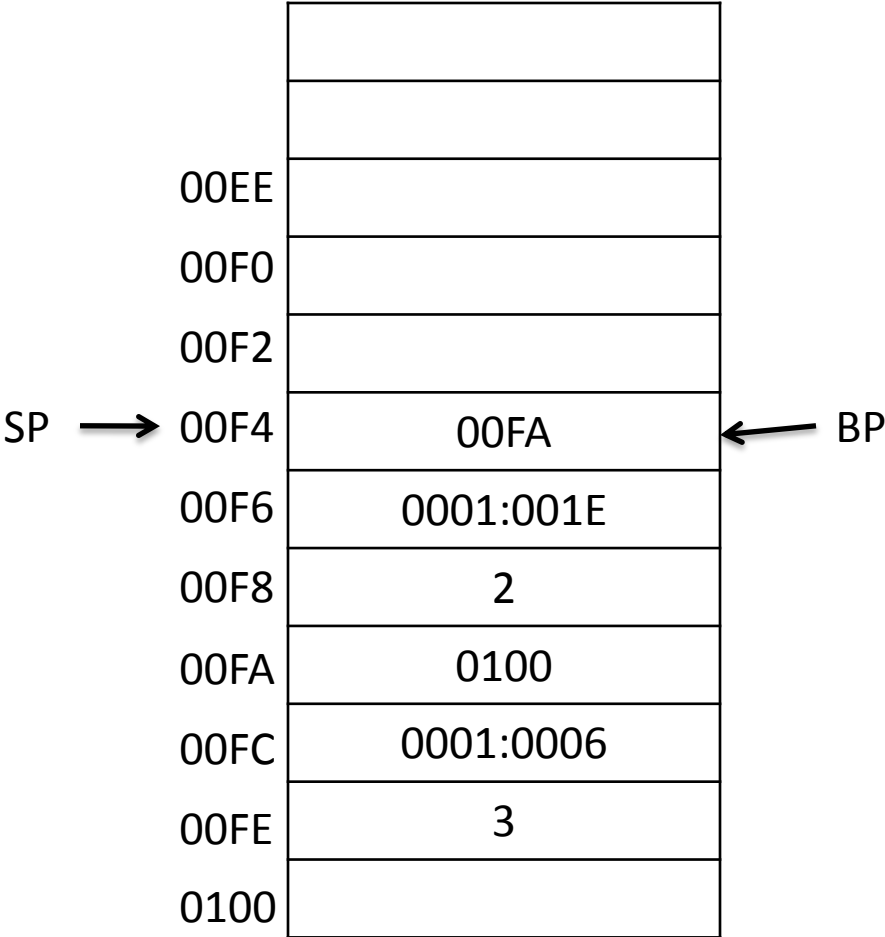
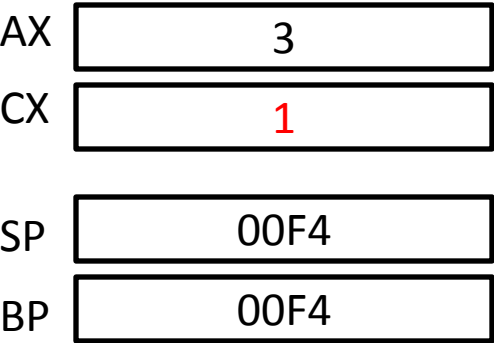
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

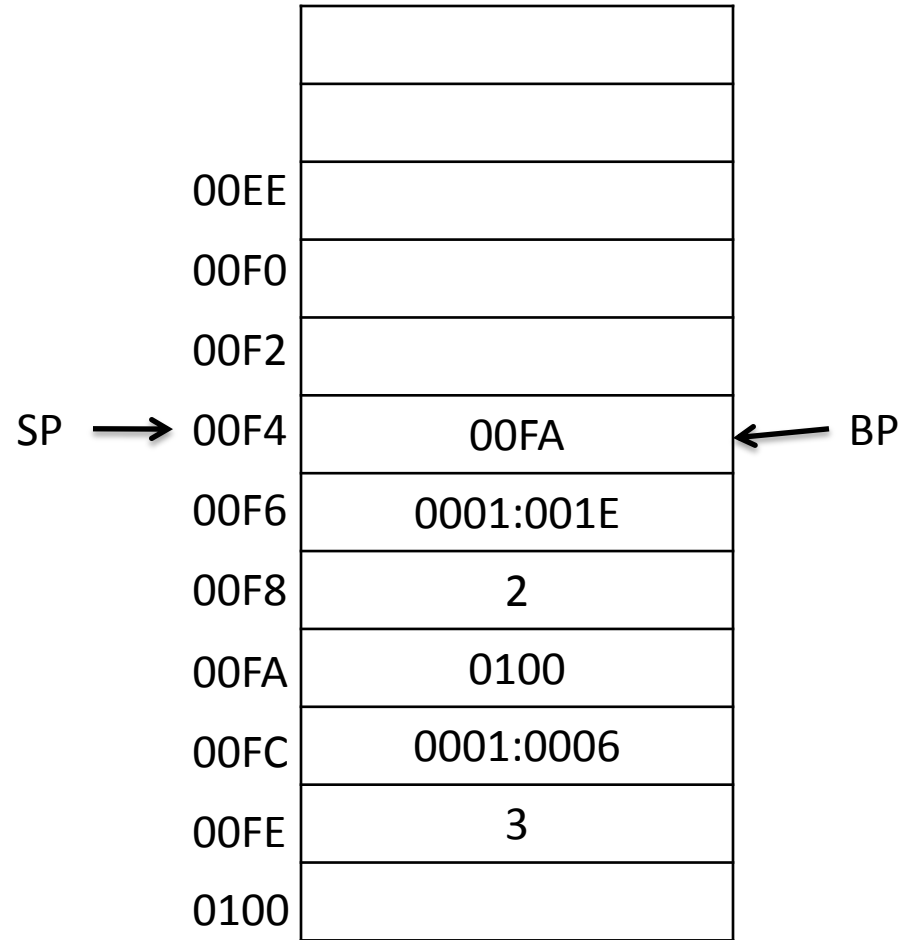
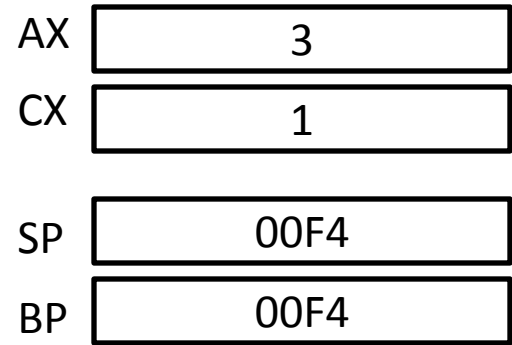
END       MAIN
```




```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

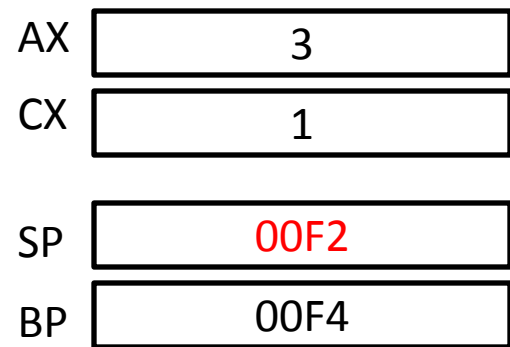
END      MAIN
```



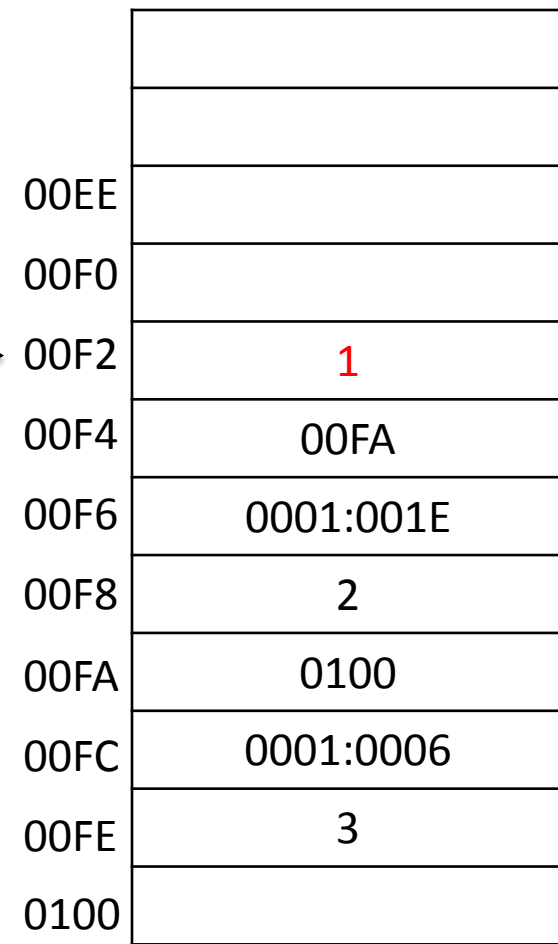
```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016  END_IF:  MOV      CX,[BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020  RETURN : POP      BP
0022      RET      2

END      MAIN
```



SP →



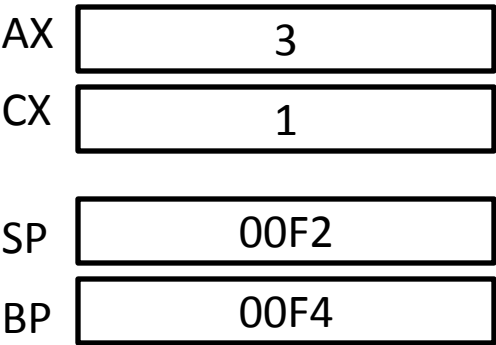
← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

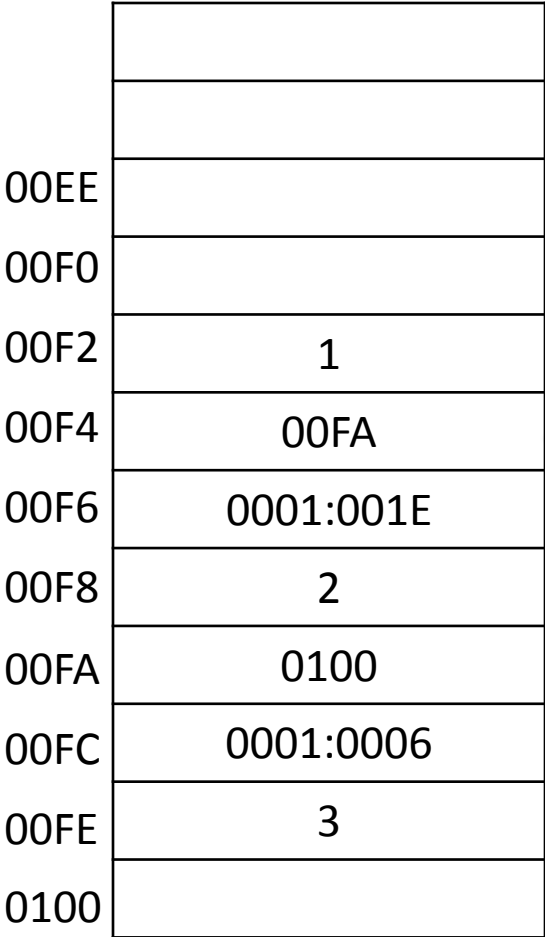
MAIN ENDP

FACTORIAL PROC NEAR

```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



SP →

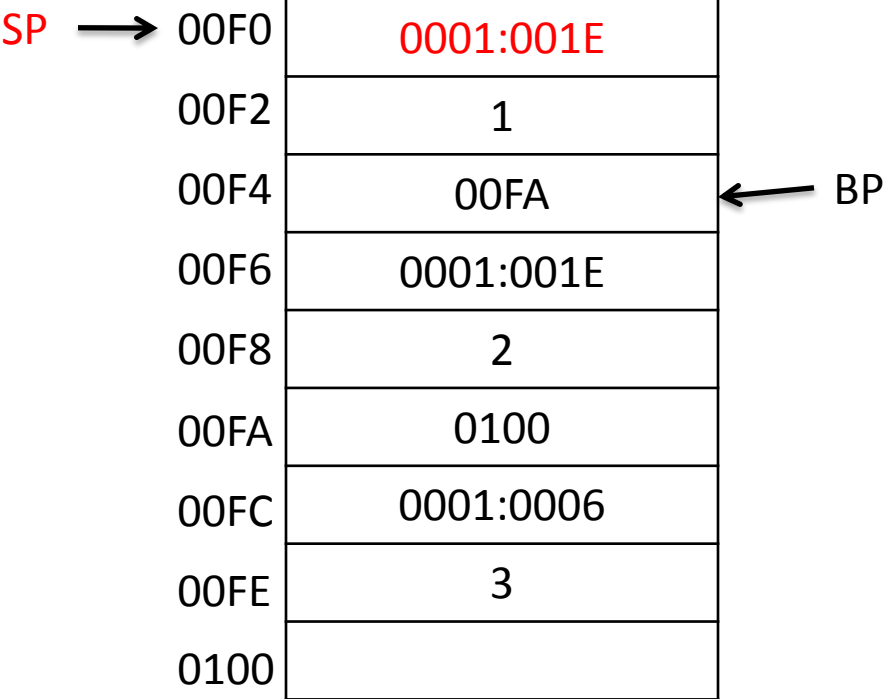
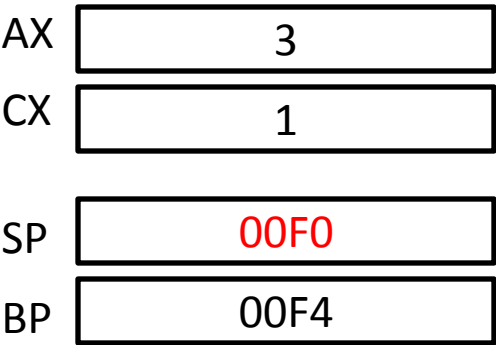


← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

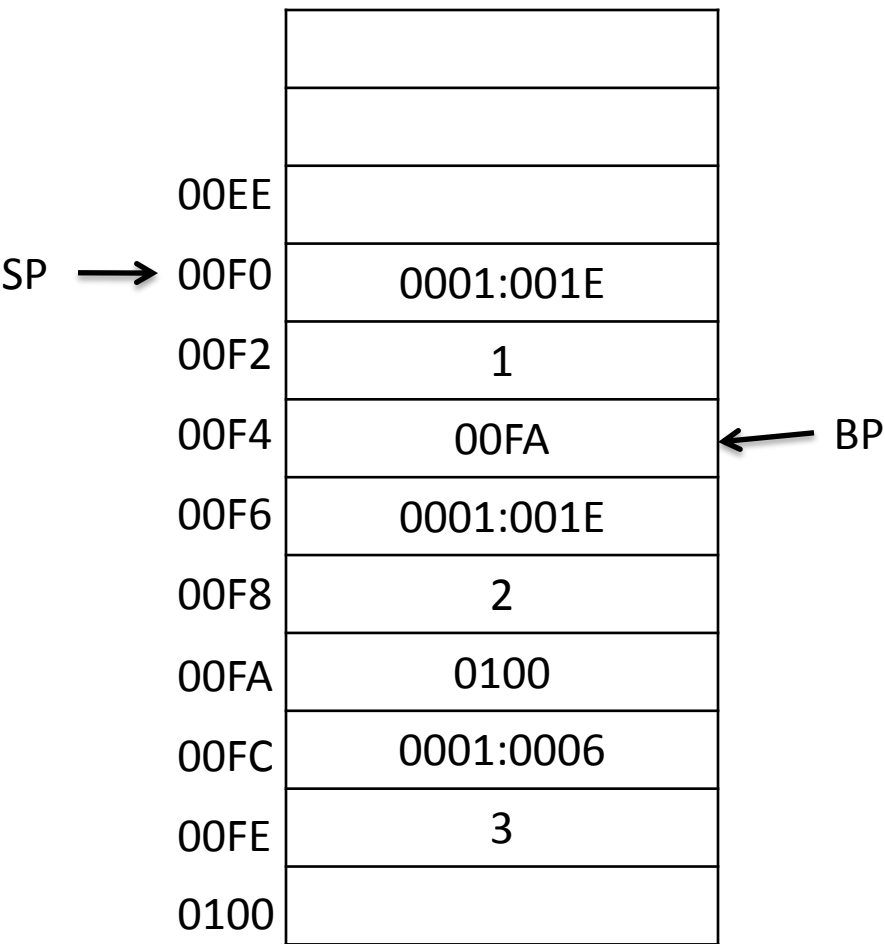
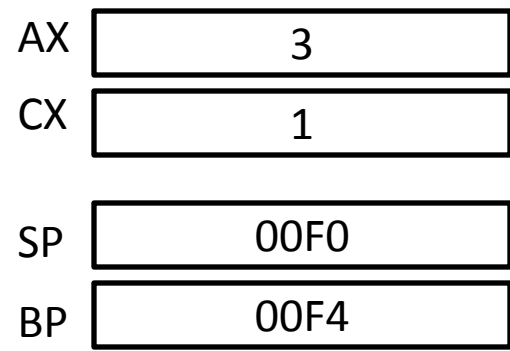
```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

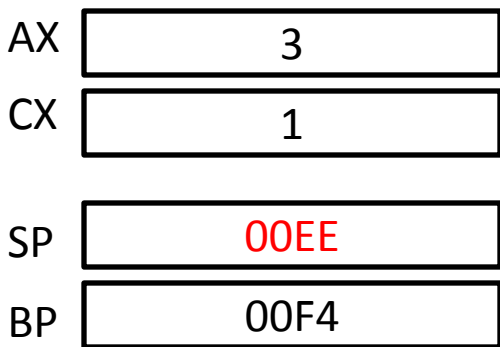
```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



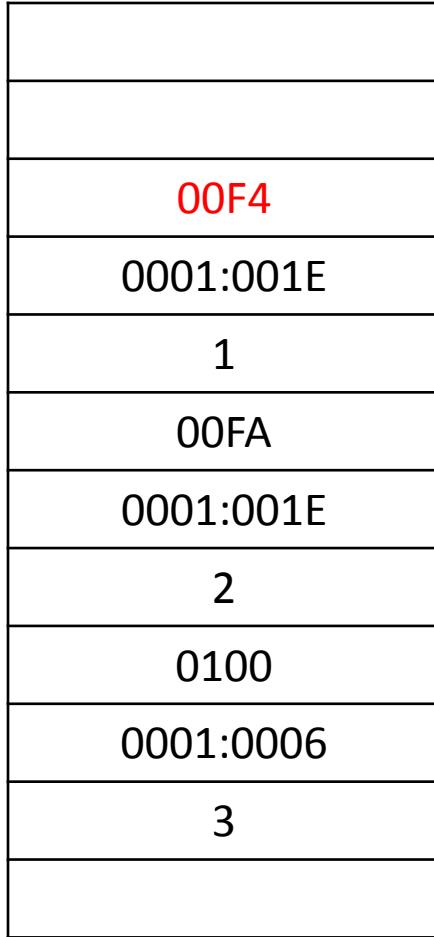
```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



SP → 00EE



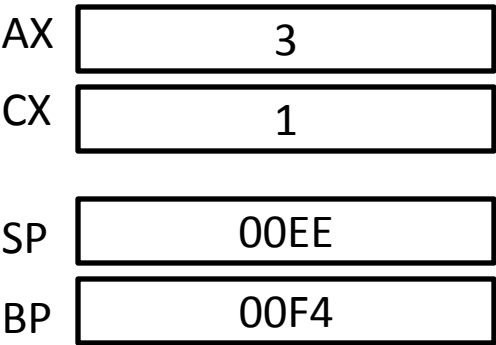
← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

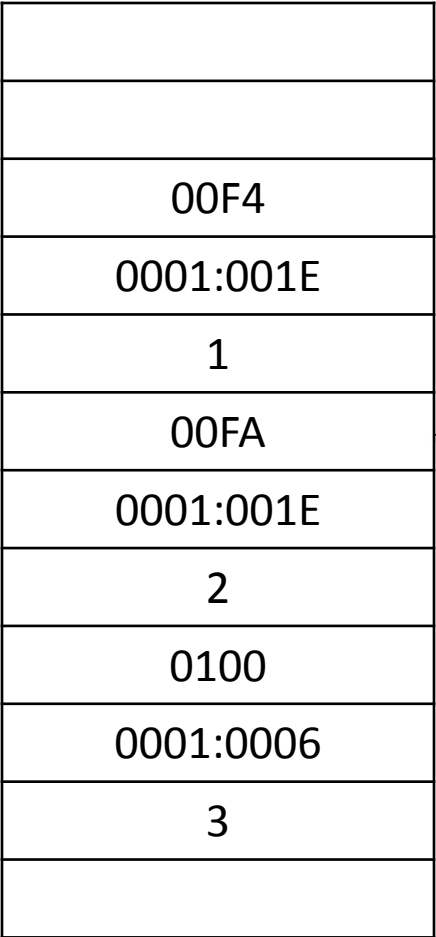
MAIN ENDP

FACTORIAL PROC NEAR

```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```



SP → 00EE

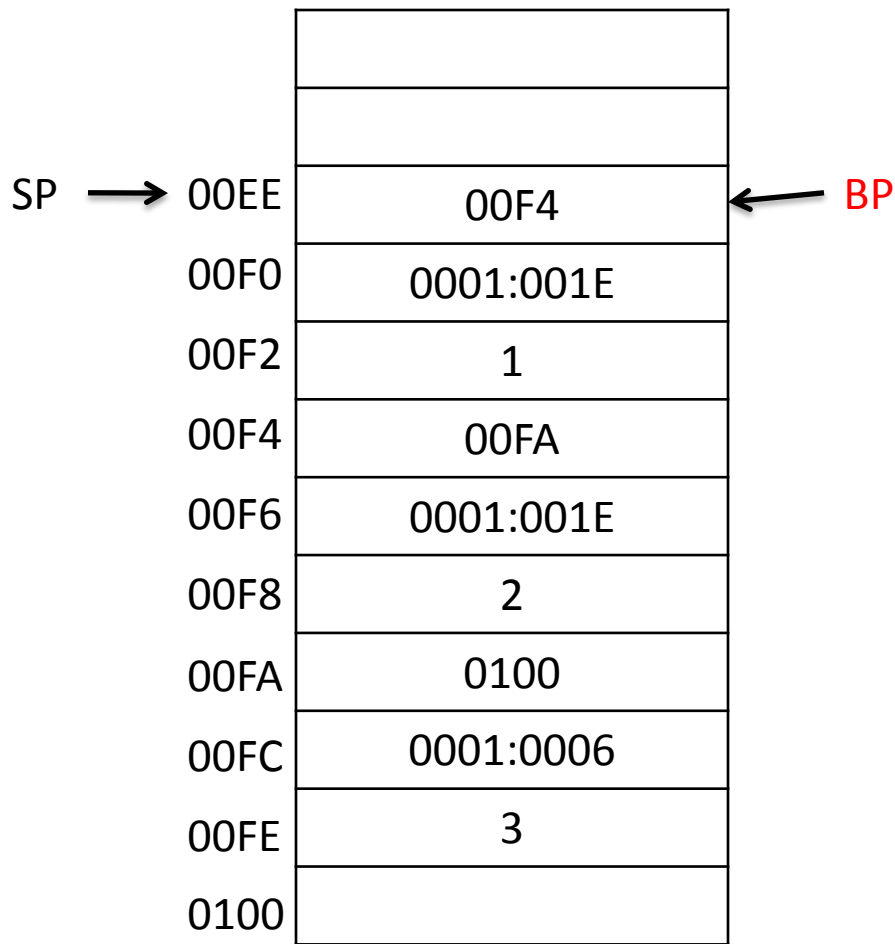
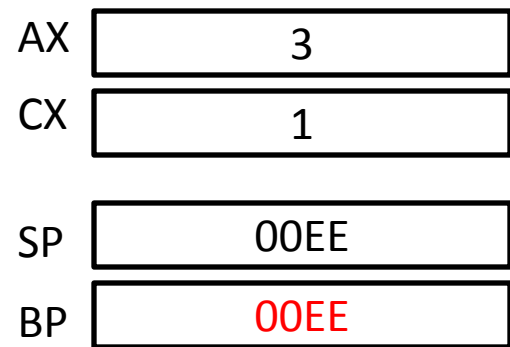


← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

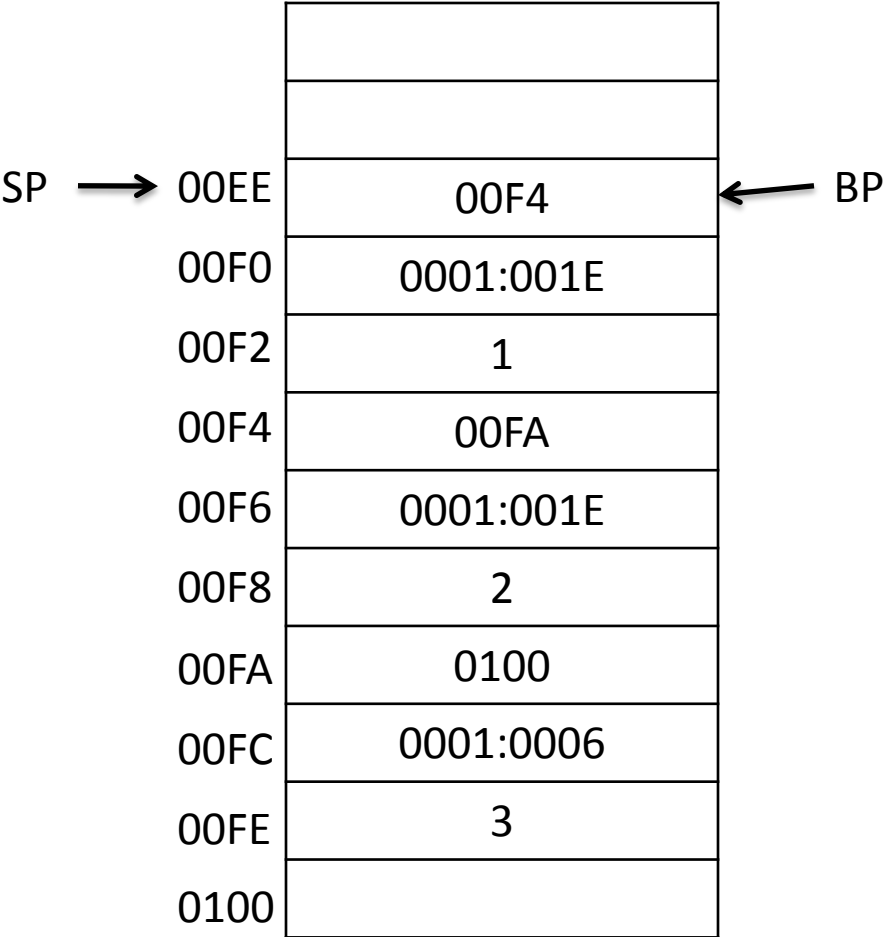
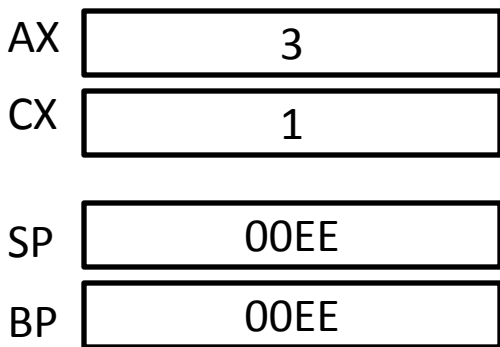
```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2
END MAIN
```




```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

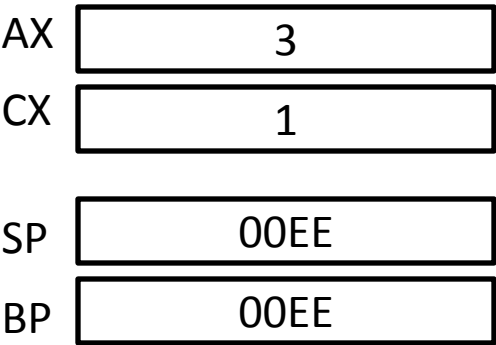
```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2
END MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

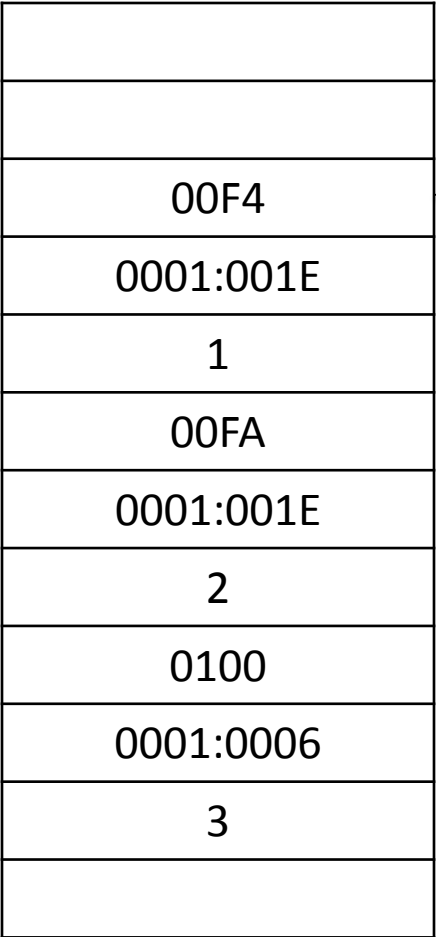
```
MAIN ENDP
FACTORIAL PROC NEAR
```

```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2
END MAIN
```



SP → 00EE

← BP



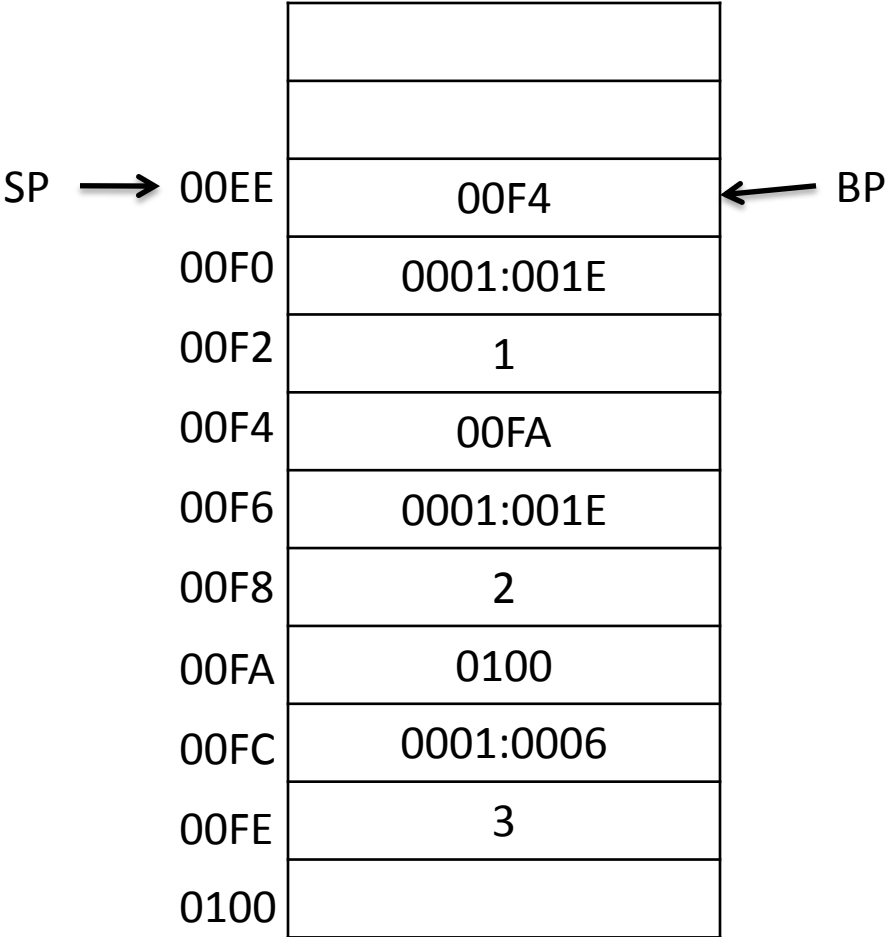
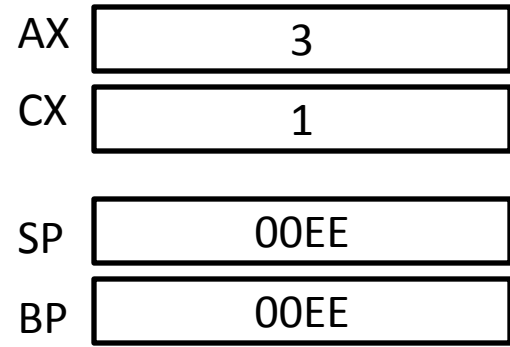
```

MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

END       MAIN

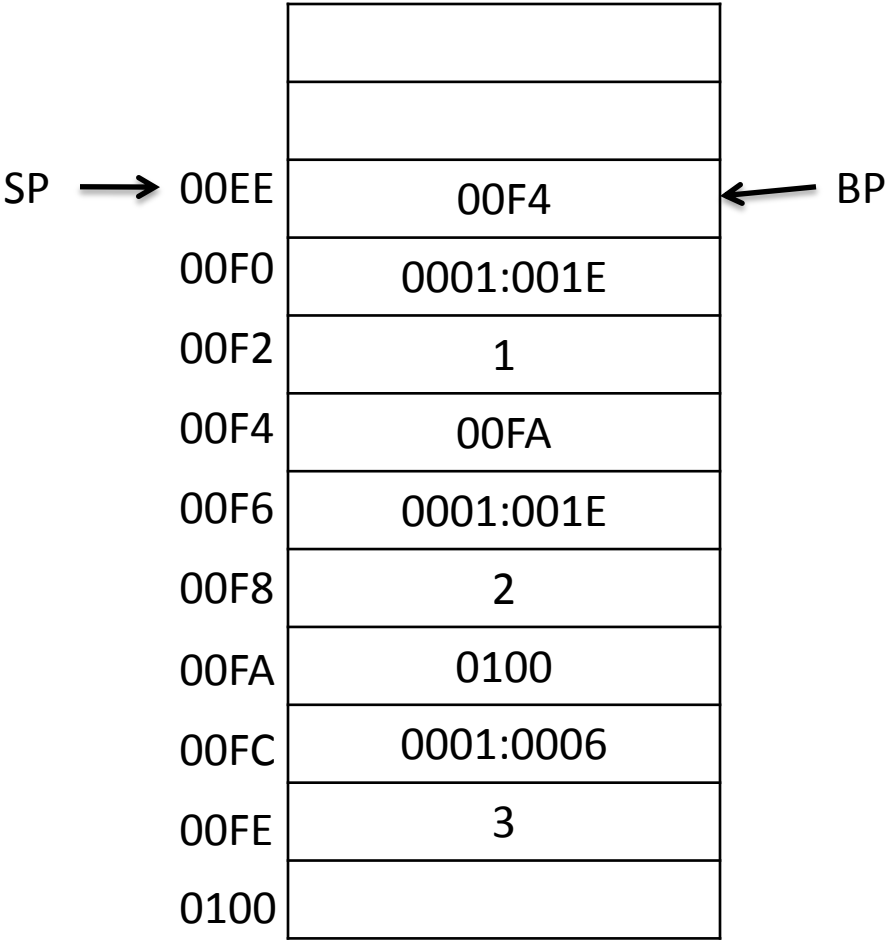
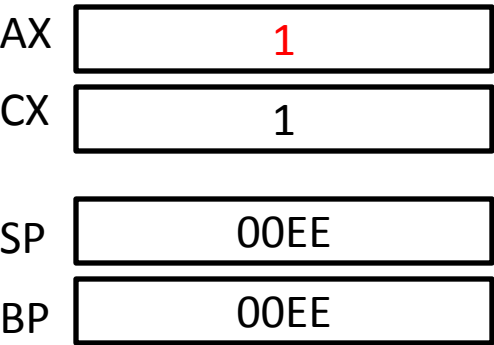
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

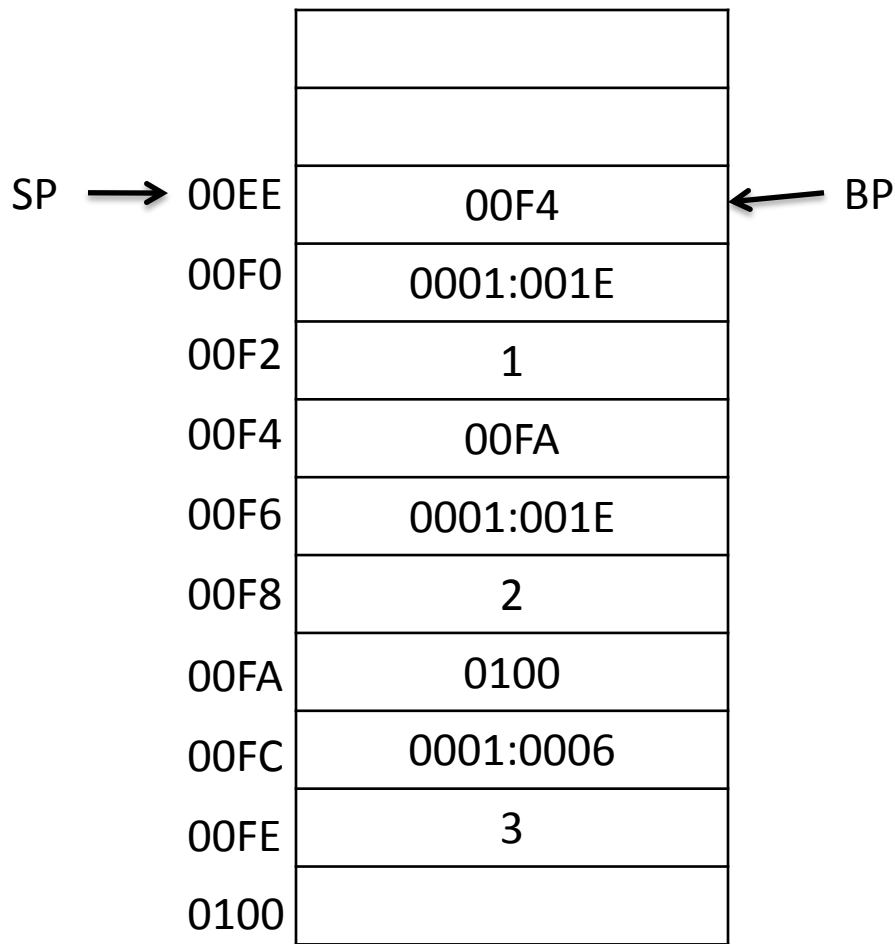
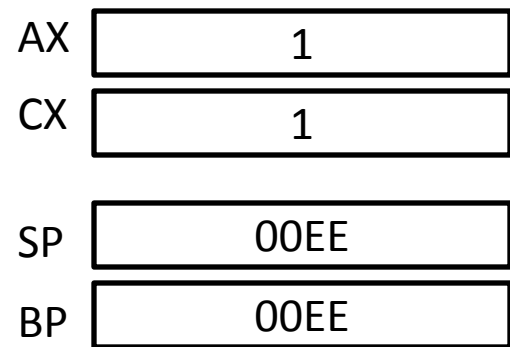
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

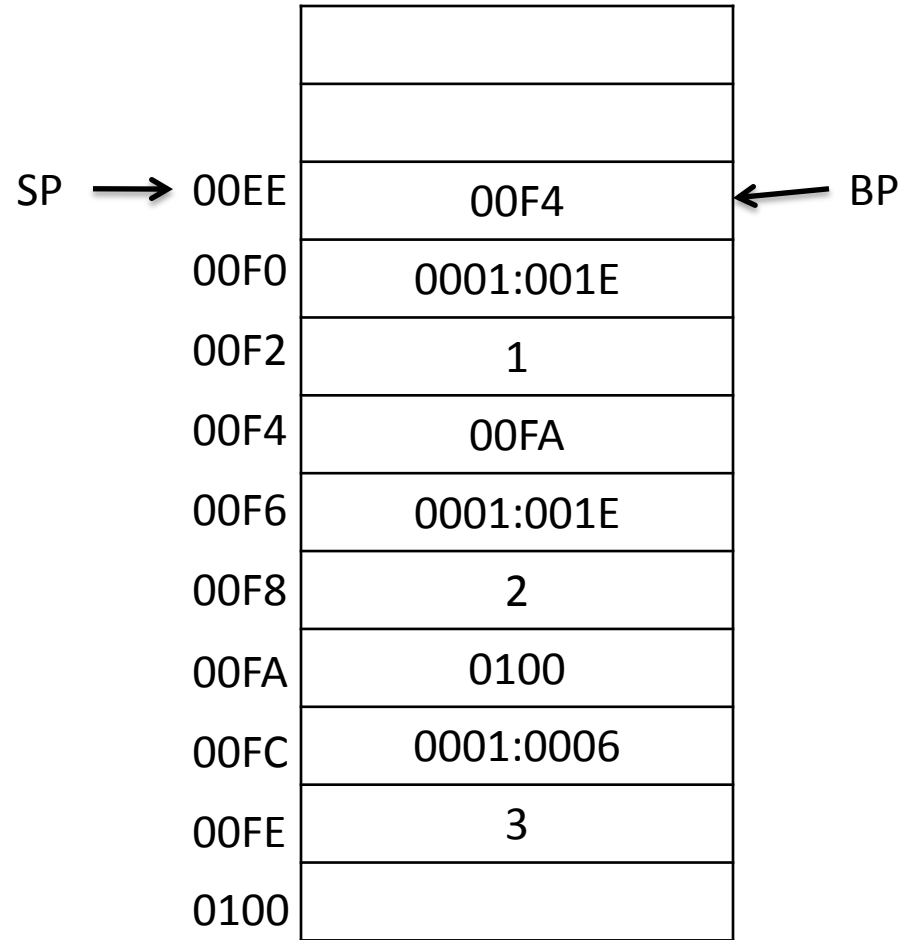
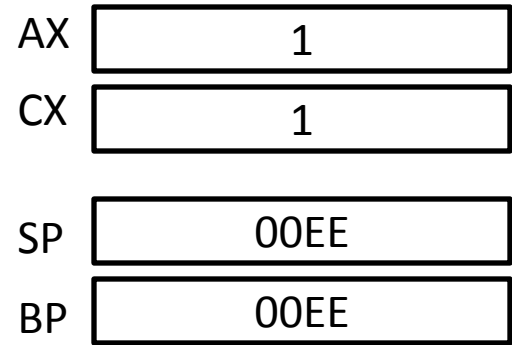
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

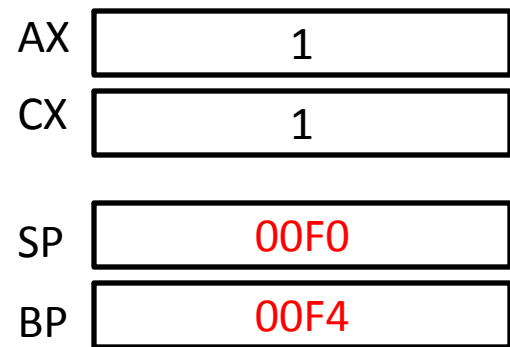
END      MAIN
```



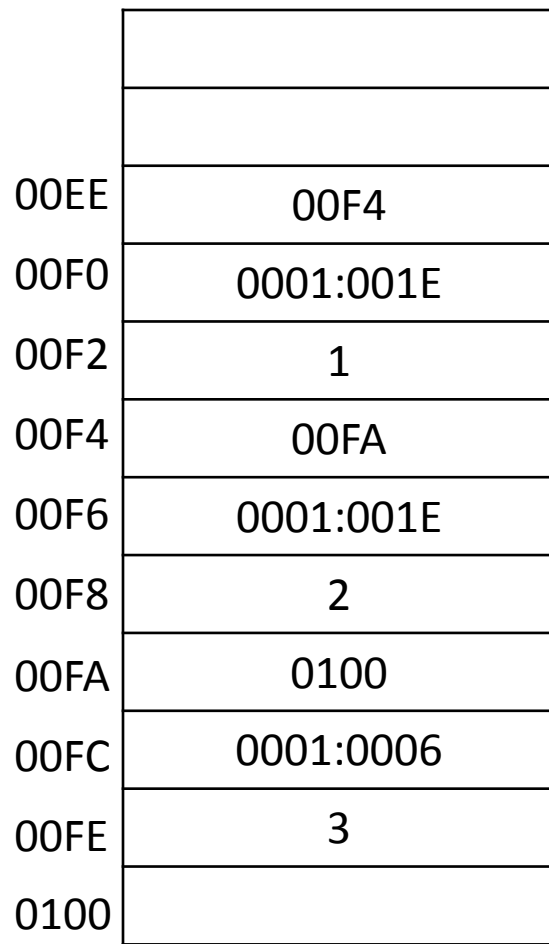
```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN
```



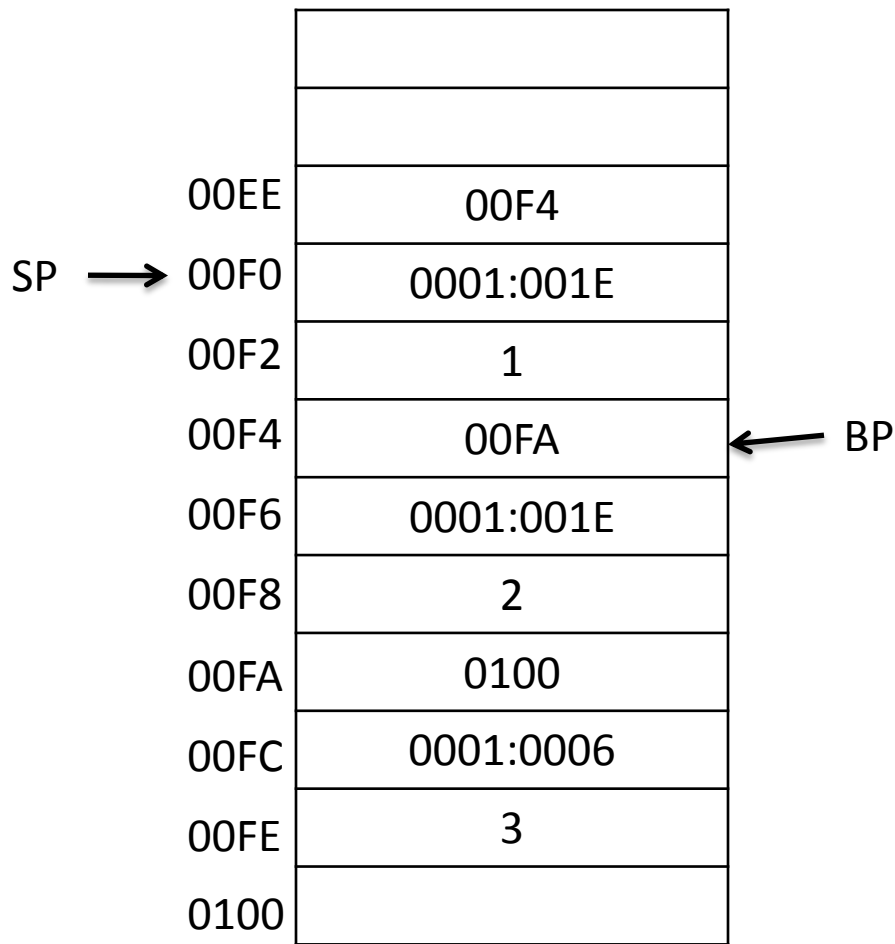
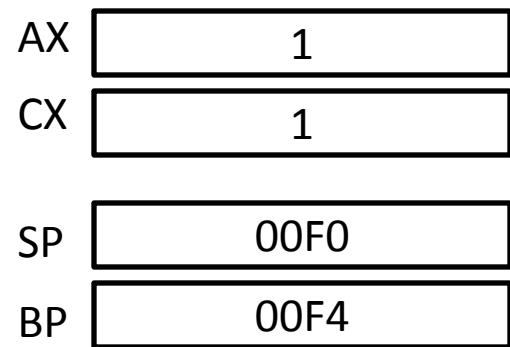
SP →



← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END      MAIN
```



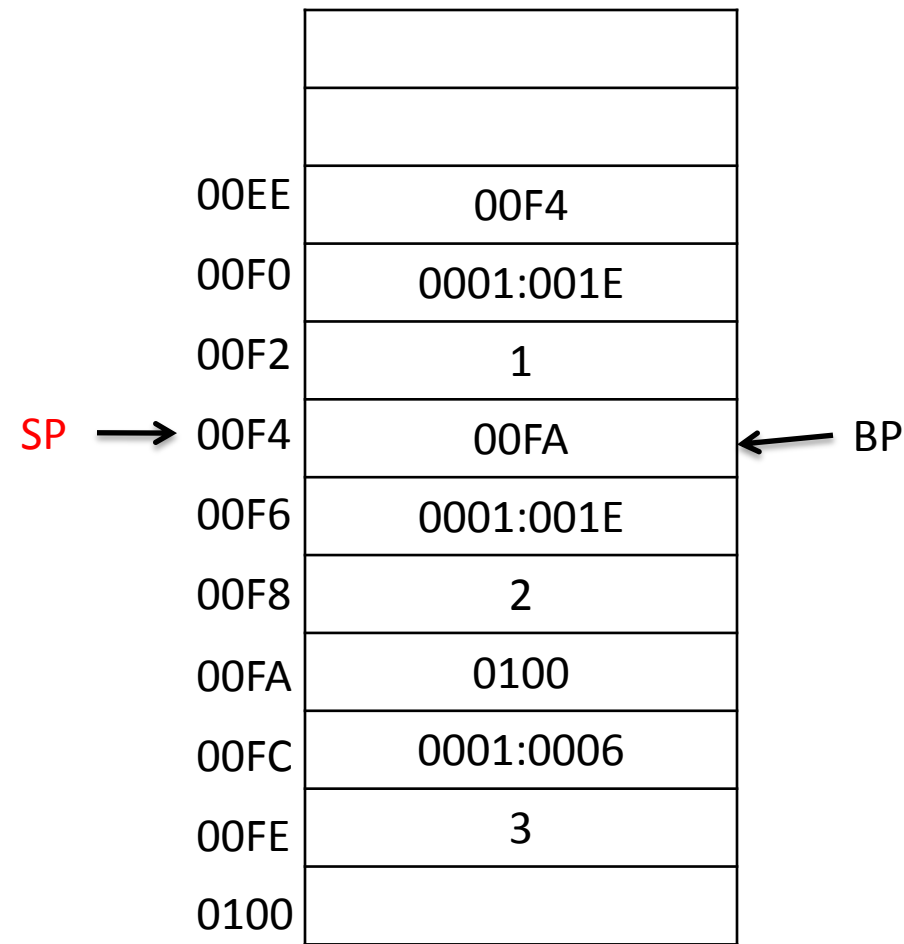
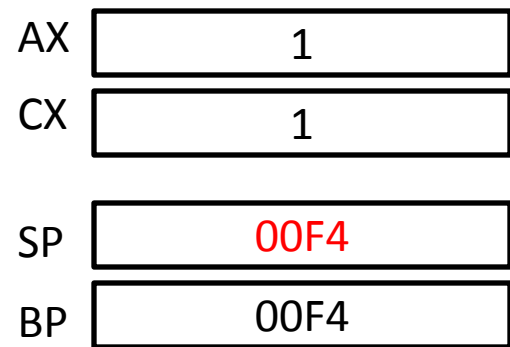

```

MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016  END_IF:  MOV      CX,[BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020  RETURN : POP      BP
0022      RET      2

END      MAIN

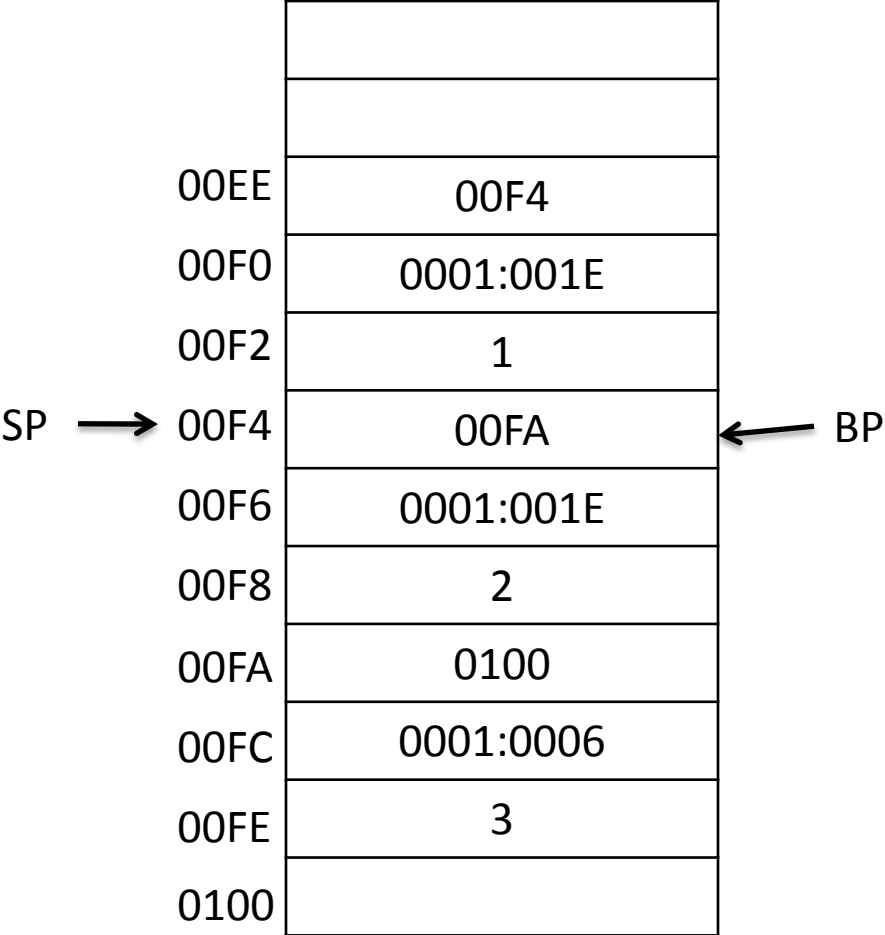
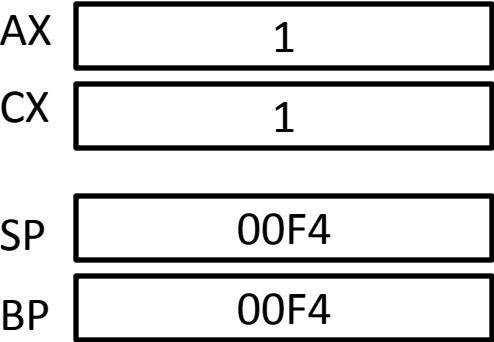
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

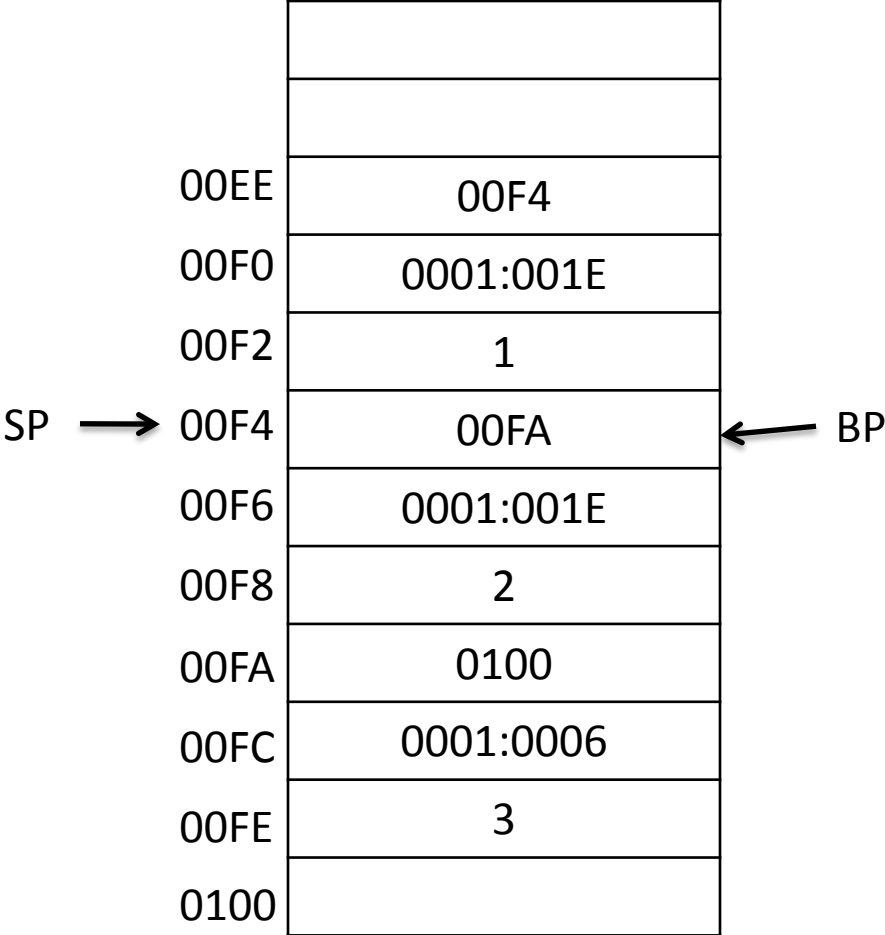
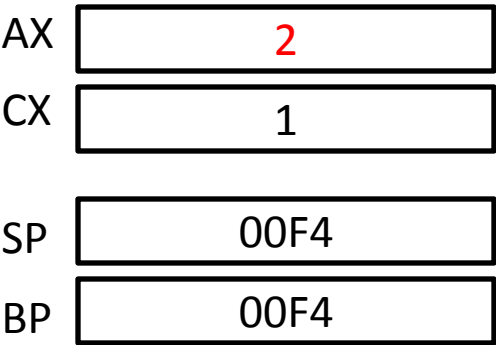
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN
```



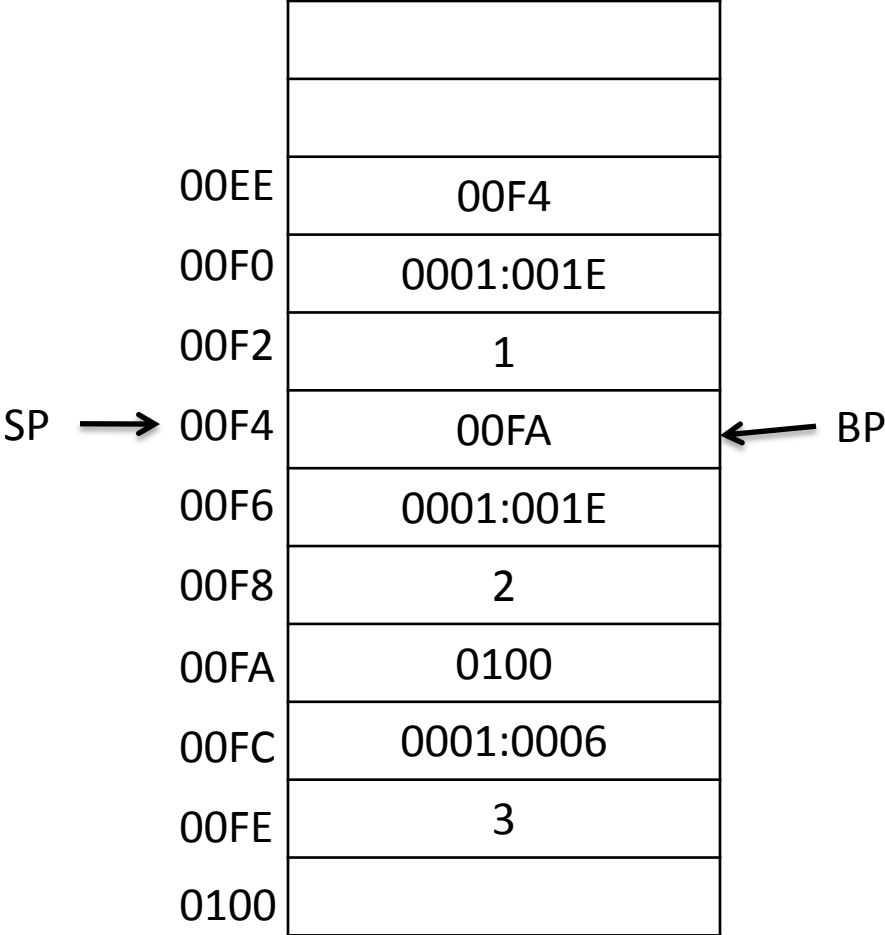
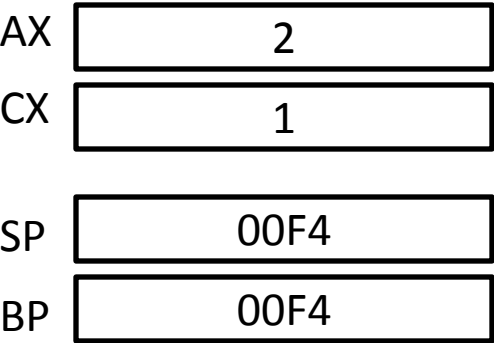
```

MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN

```



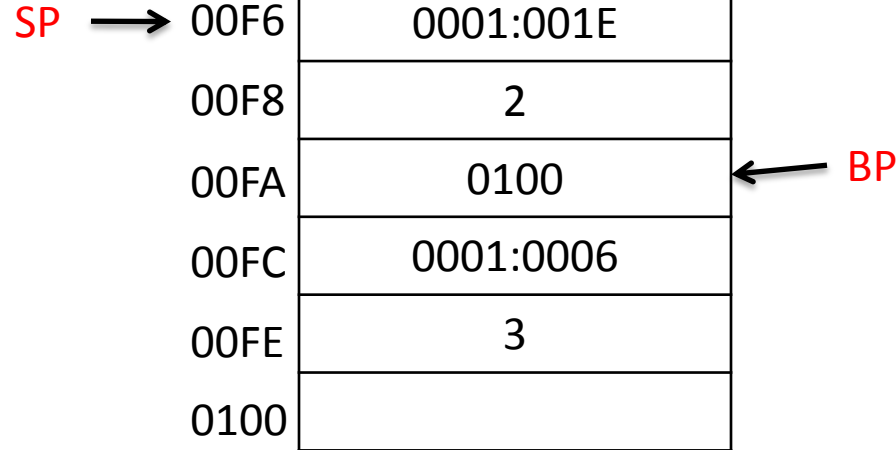
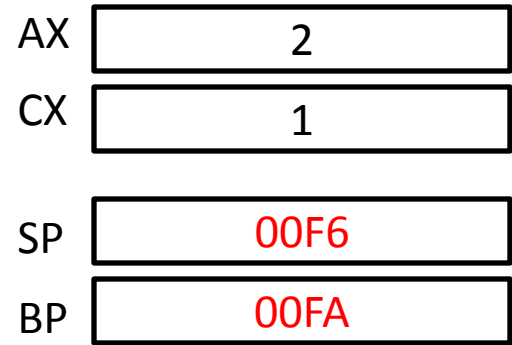
```

MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

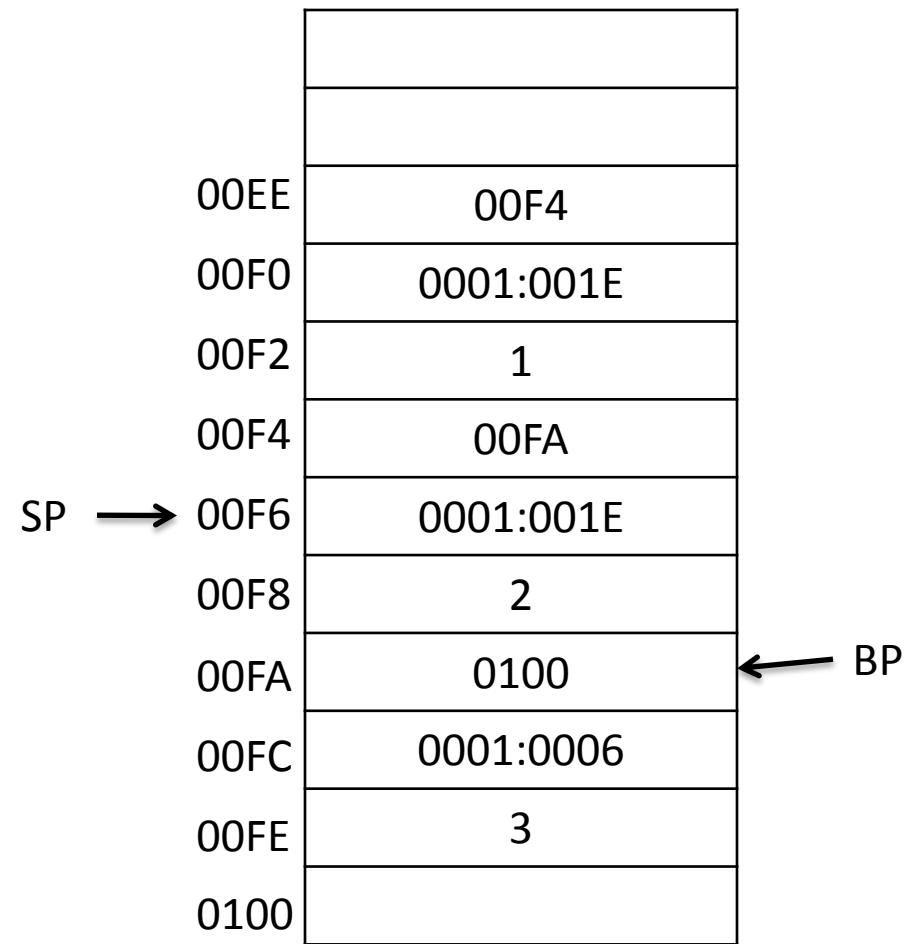
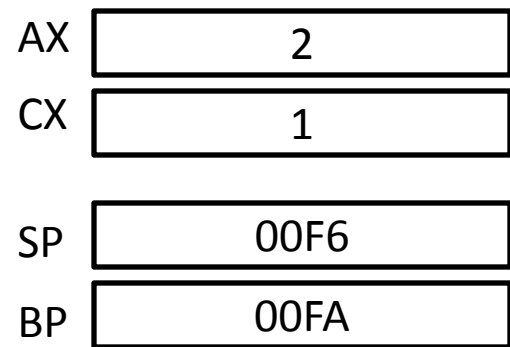
END      MAIN

```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

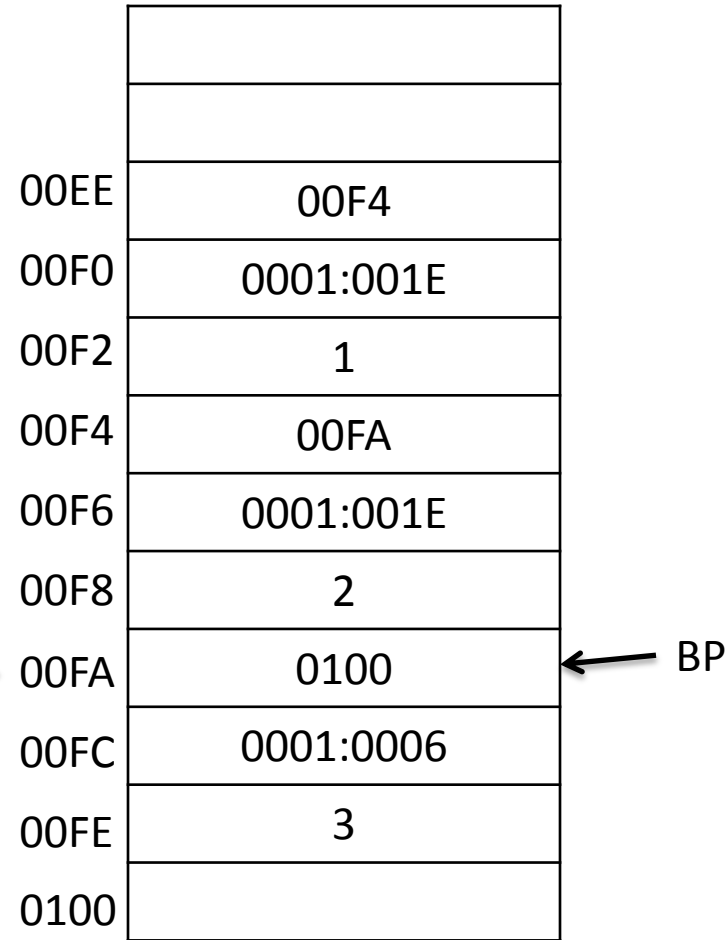
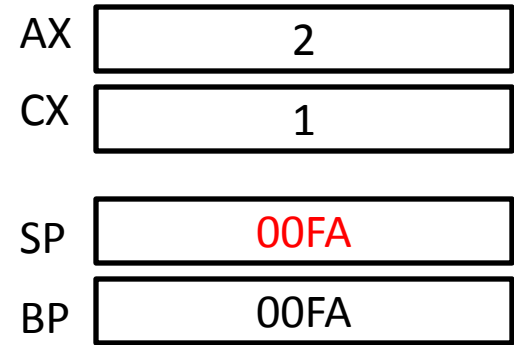
MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END      MAIN
```



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

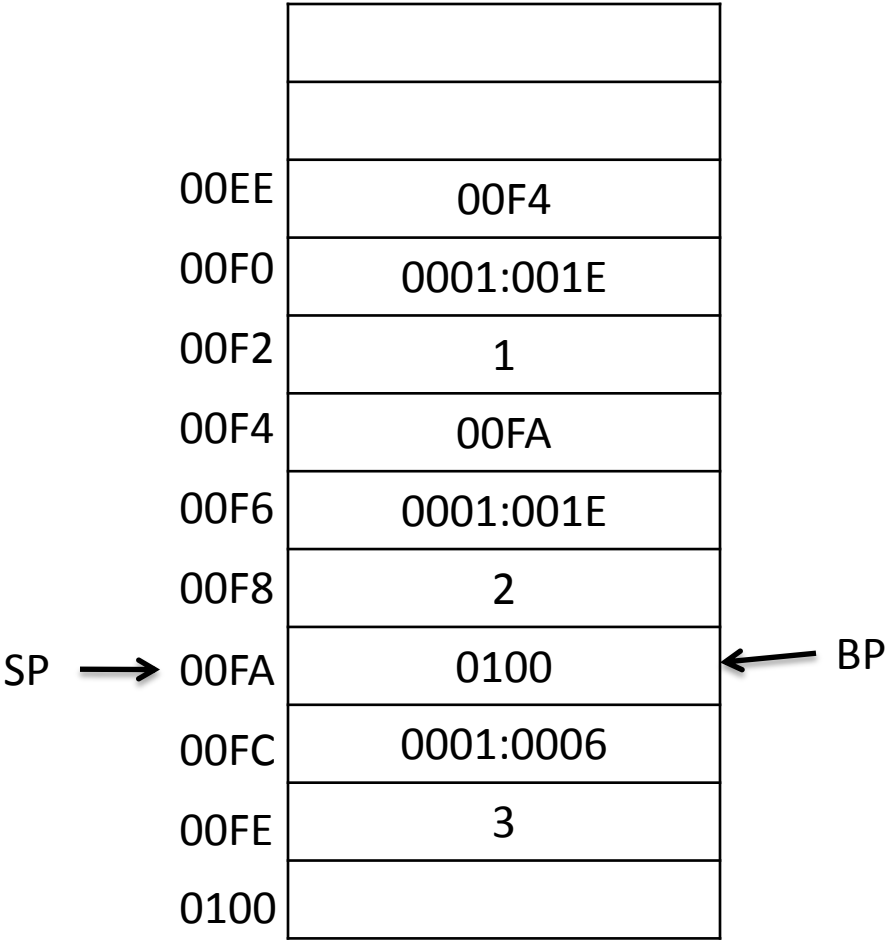
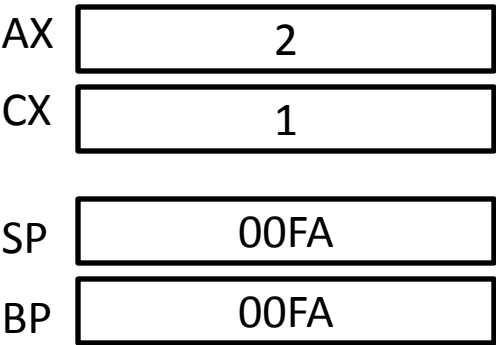
END      MAIN
```



```
MAIN PROC
0000      MOV      AX,3
0002      PUSH     AX
0004      CALL     FACTORIAL
0006      MOV      AH,4CH
0008      INT      21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH     BP
000C      MOV      BP,SP
000E      CMP      WORD PTR[BP+4],1
0010      JG       END_IF
0012      MOV      AX,1
0014      JMP      RETURN
0016      END_IF:  MOV      CX,[BP+4]
0018      DEC      CX
001A      PUSH     CX
001C      CALL     FACTORIAL
001E      MUL      WORD PTR[BP+4]
0020      RETURN : POP      BP
0022      RET      2

END      MAIN
```




```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN
```

AX

6

CX

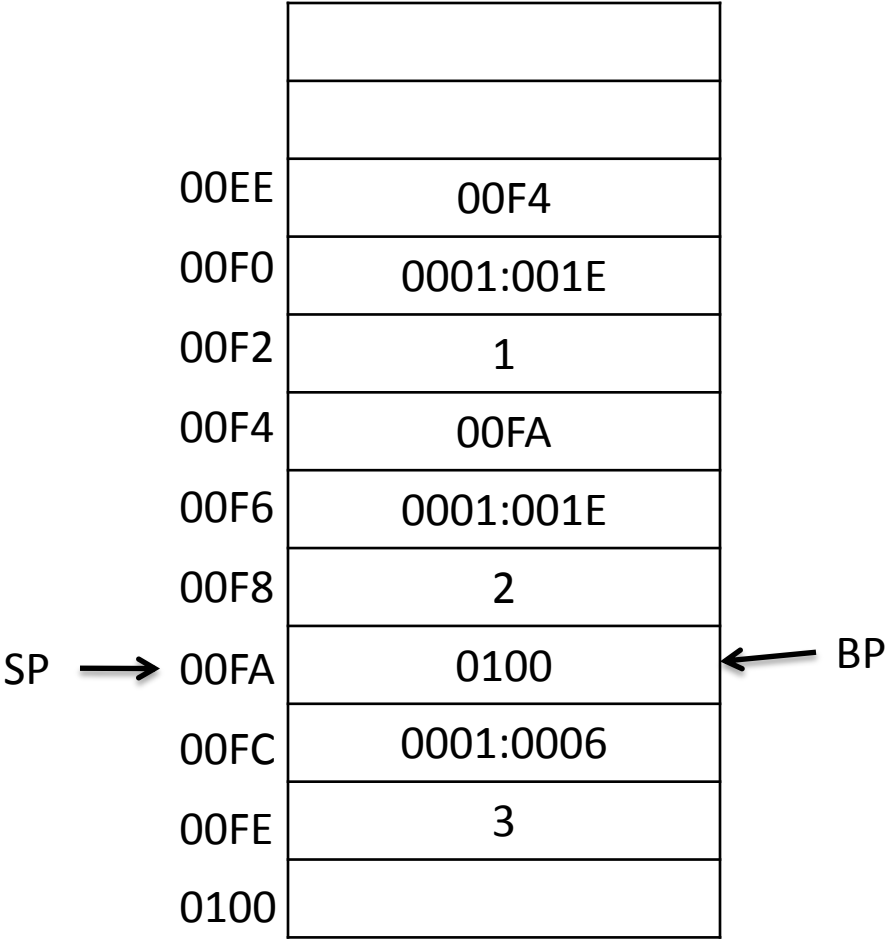
1

SP

00FA

BP

00FA

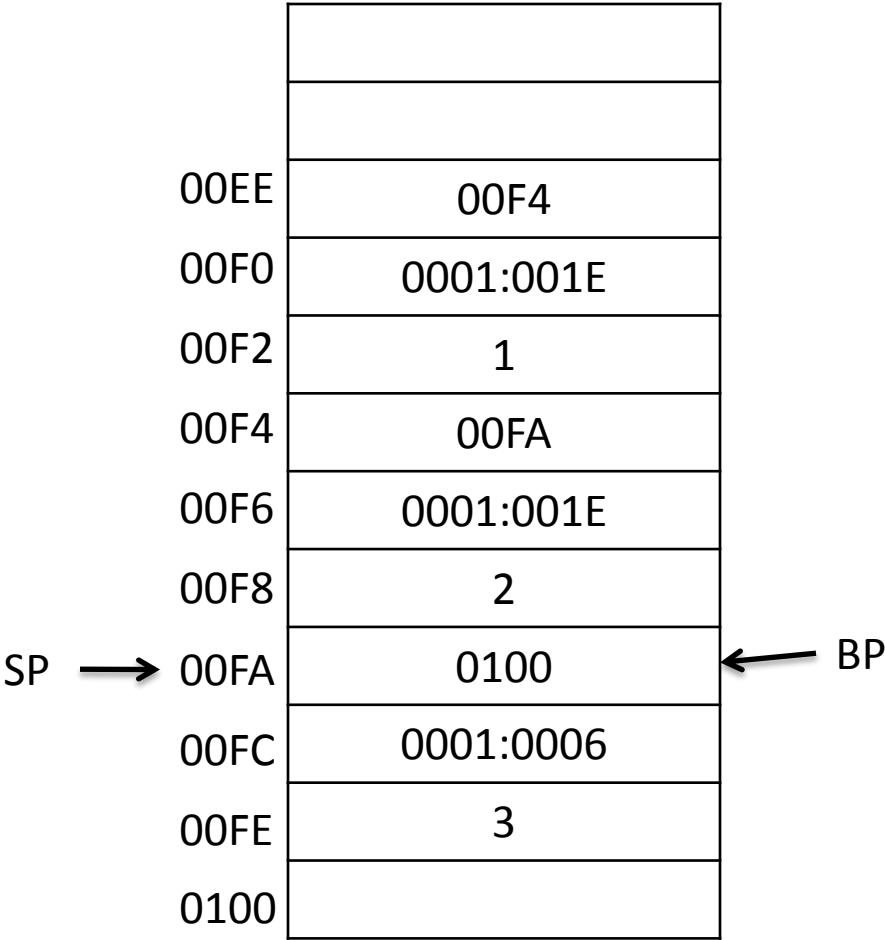


```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END      MAIN
```

AX	6
CX	1
SP	00FA
BP	00FA



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2

END MAIN
```

AX

6

CX

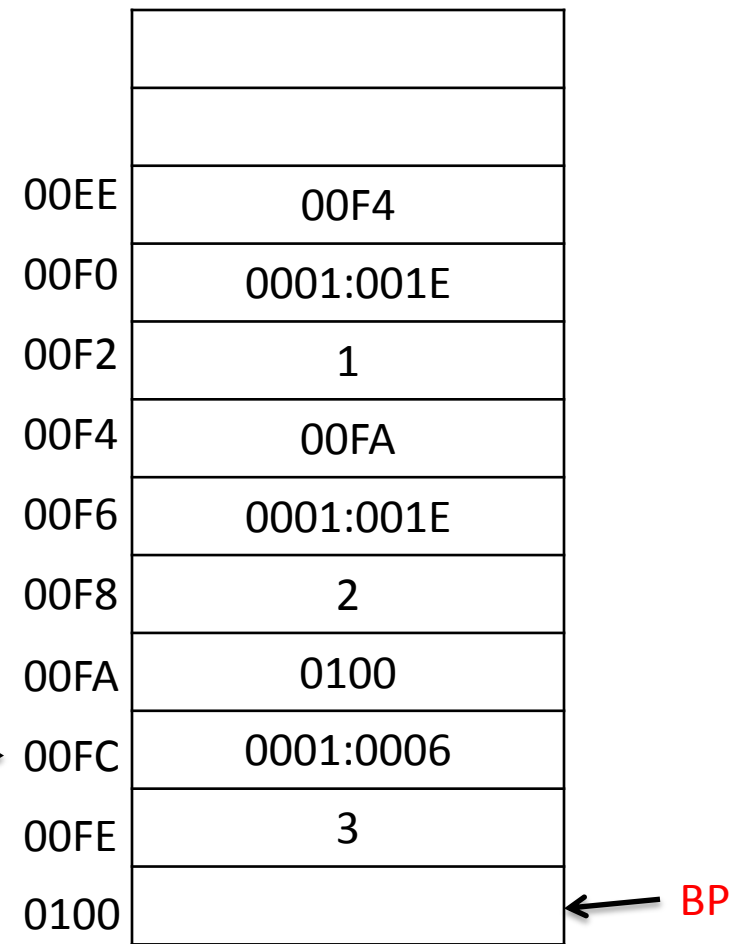
1

SP

00FC

BP

0100



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022  RET     2
END MAIN
```

AX	6
CX	1
SP	00FC
BP	0100

00EE	00F4
00F0	0001:001E
00F2	1
00F4	00FA
00F6	0001:001E
00F8	2
00FA	0100
00FC	0001:0006
00FE	3
0100	

SP →

← BP

```
MAIN PROC
0000      MOV     AX,3
0002      PUSH    AX
0004      CALL    FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

```
000A      PUSH    BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH    CX
001C      CALL    FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END MAIN
```

AX	6
CX	1
SP	0100
BP	0100

00EE	00F4
00F0	0001:001E
00F2	1
00F4	00FA
00F6	0001:001E
00F8	2
00FA	0100
00FC	0001:0006
00FE	3
0100	



```
MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H
```

```
MAIN ENDP
FACTORIAL PROC NEAR
```

```
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2
END MAIN
```

AX	6
CX	1
SP	0100
BP	0100

00EE	00F4
00F0	0001:001E
00F2	1
00F4	00FA
00F6	0001:001E
00F8	2
00FA	0100
00FC	0001:0006
00FE	3



```

MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

MAIN ENDP
FACTORIAL PROC NEAR
000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016      END_IF: MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020      RETURN : POP     BP
0022      RET     2

END       MAIN

```

AX	6
CX	1
SP	0100
BP	0100

00EE	00F4
00F0	0001:001E
00F2	1
00F4	00FA
00F6	0001:001E
00F8	2
00FA	0100
00FC	0001:0006
00FE	3
0100	

SP → 0100

← BP

```

MAIN PROC
0000      MOV     AX,3
0002      PUSH   AX
0004      CALL   FACTORIAL
0006      MOV     AH,4CH
0008      INT     21H

```

MAIN ENDP

FACTORIAL PROC NEAR

```

000A      PUSH   BP
000C      MOV     BP,SP
000E      CMP     WORD PTR[BP+4],1
0010      JG      END_IF
0012      MOV     AX,1
0014      JMP     RETURN
0016  END_IF:  MOV     CX,[BP+4]
0018      DEC     CX
001A      PUSH   CX
001C      CALL   FACTORIAL
001E      MUL     WORD PTR[BP+4]
0020  RETURN : POP     BP
0022      RET     2
END        MAIN

```

AX	6
CX	1
SP	00FE
BP	0100

SP →

00EE	00F4
00F0	0001:001E
00F2	1
00F4	00FA
00F6	0001:001E
00F8	2
00FA	0100
00FC	0001:0006
00FE	0001:000A
0100	

← BP