Chapter 8

The Stack and Introduction to Procedures

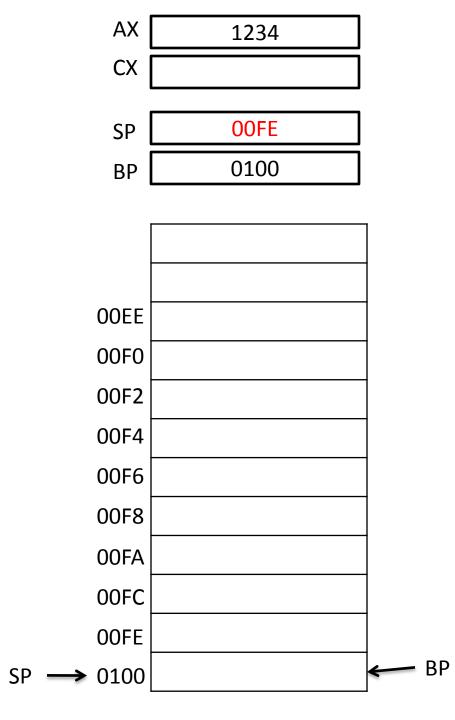
PUSH source

•source is a 16-bit register / memory word

AX	1234	
CX		
·		
SP	0100	
ВР	0100	
		_
]
00EE		1
00F0		1
00F2		1
00F4		
00F6		
00F8		
00FA		
00FC		
00FE		
SP → 0100		← BF

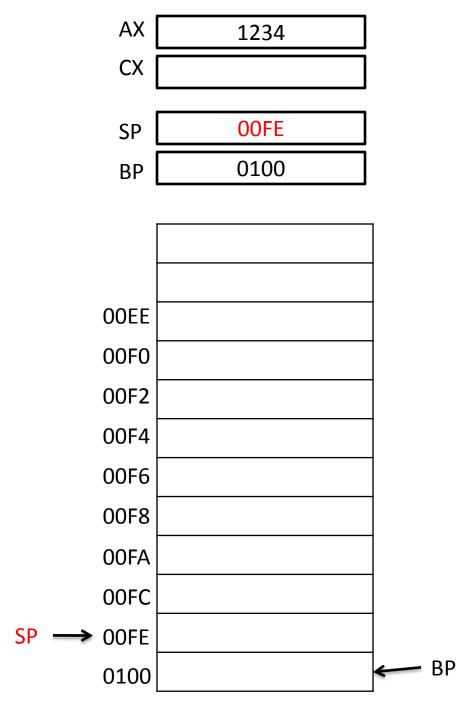
PUSH source

•source is a 16-bit register / memory word



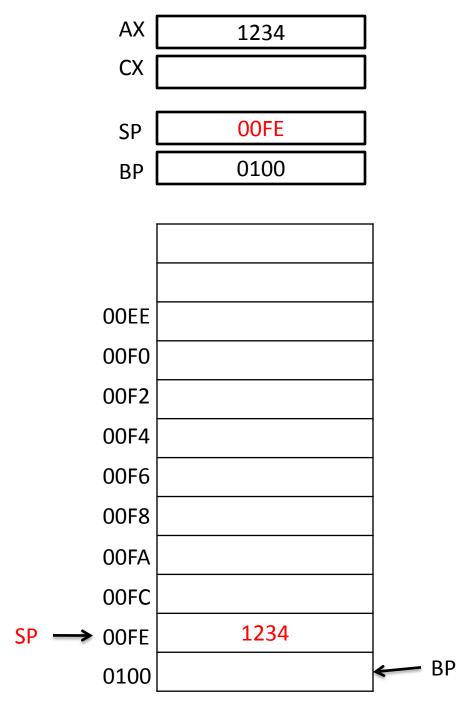
PUSH source

•source is a 16-bit register / memory word

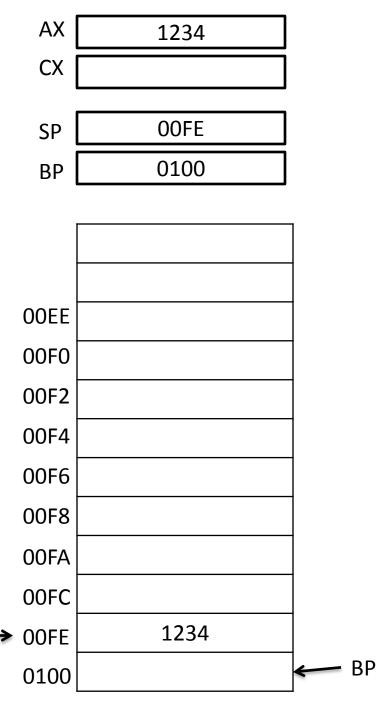


PUSH source

•source is a 16-bit register / memory word

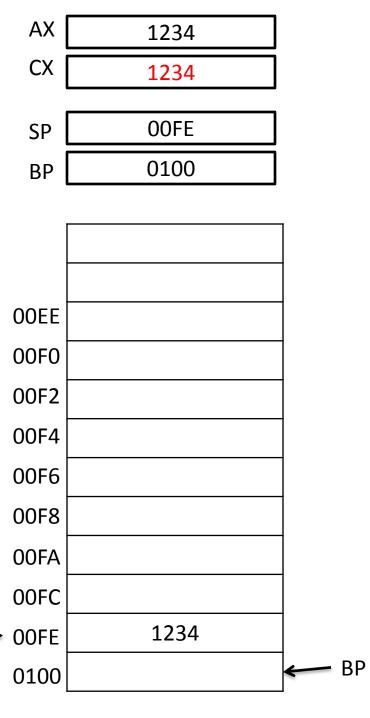


POP POP destination •destination is a 16-bit register(except IP register) / memory word POP CX



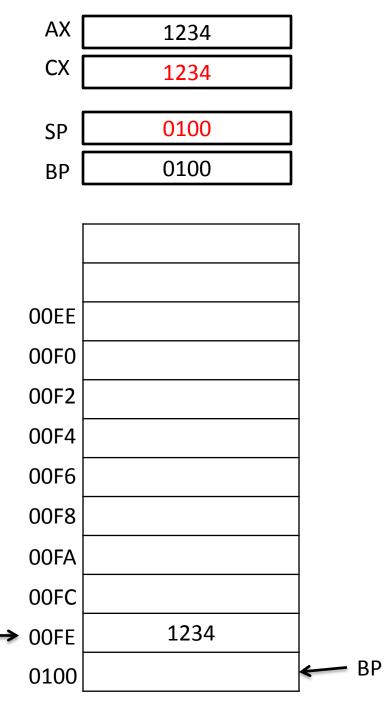
SP

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



SP

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



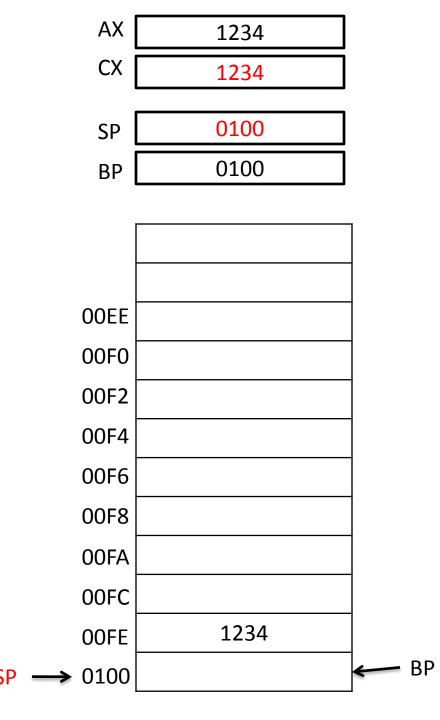
SP

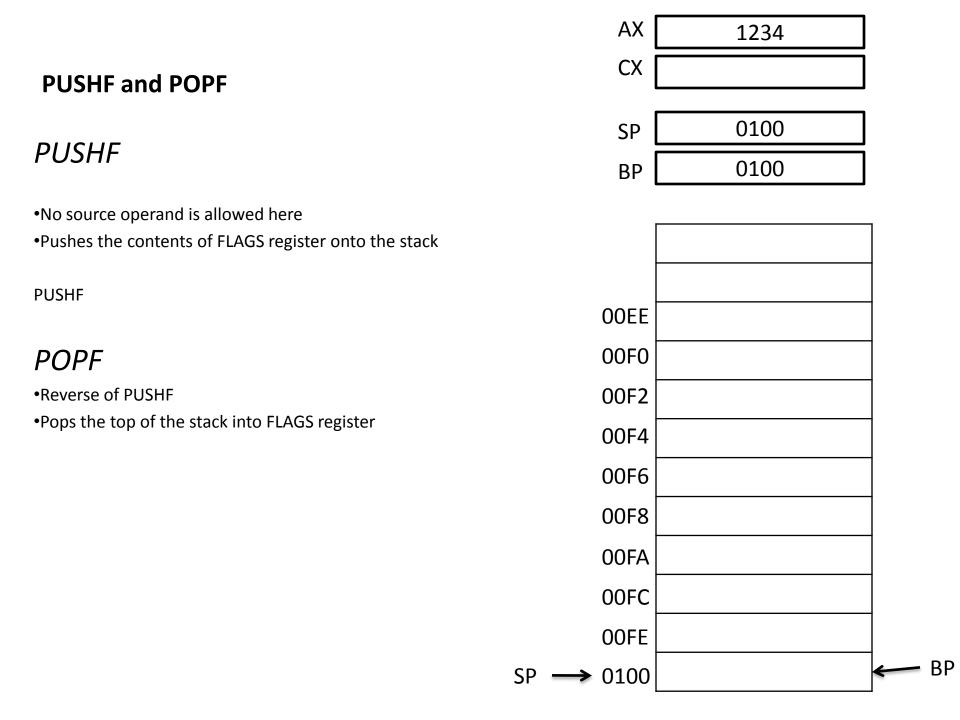
POP

POP destination

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

POP CX





Order of PUSH and POP

PUSH A

PUSH B

PUSH C

• • •

•••

POP C

POP B

POP C

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 $% \left\{ \left(1\right) \right\} =\left\{ \left(1\right) \right\} =\left\{$

RET

name ENDP

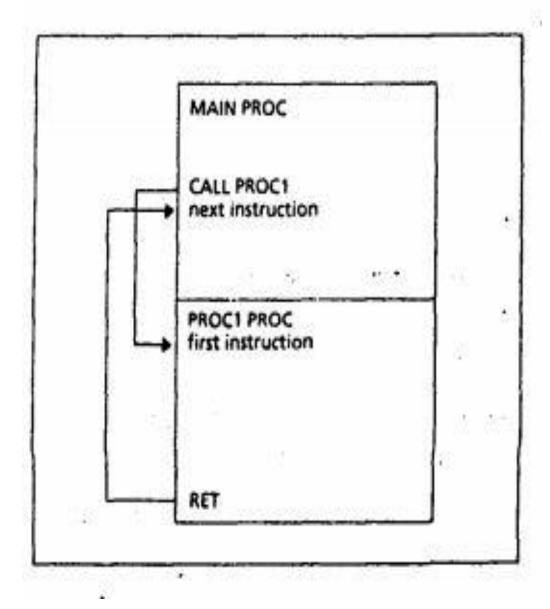
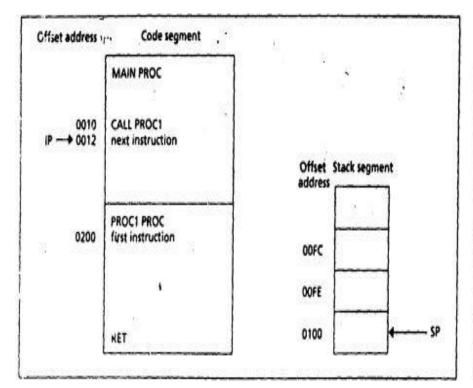


Fig: Procedure CALL and RET

CALL



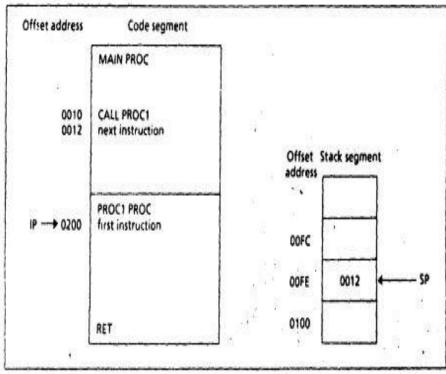
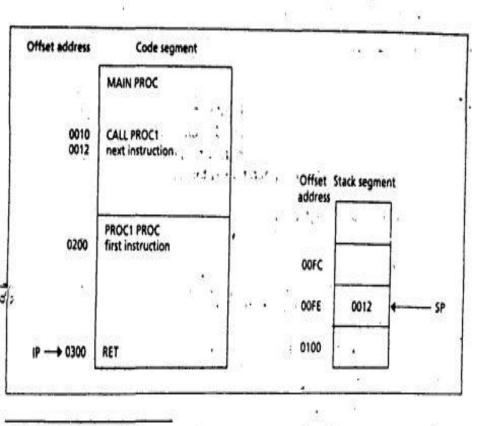


Figure 8.4A Before CALL

Figure 8.4B After CALL

RET



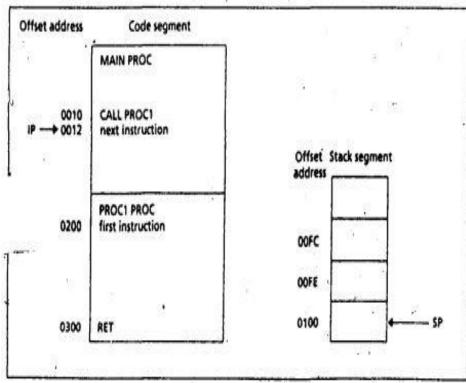


Figure 8.5A Before RET

Figure 8.58 After RET

Chapter 17

Recursion

Activation Record

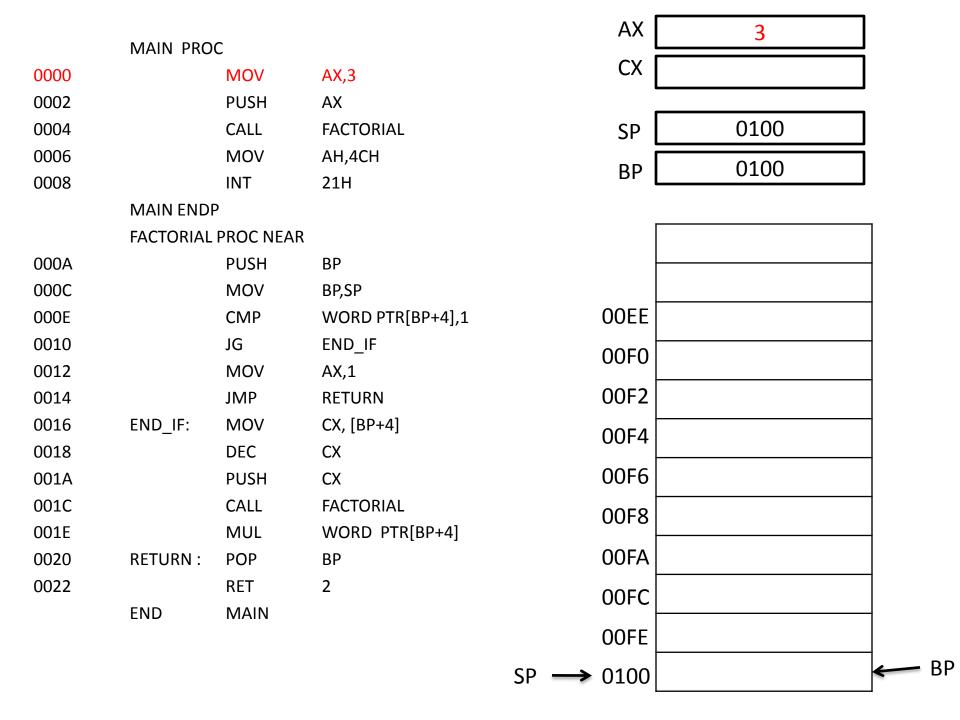
Record to be stored before CALLing a recursive function

Factorial

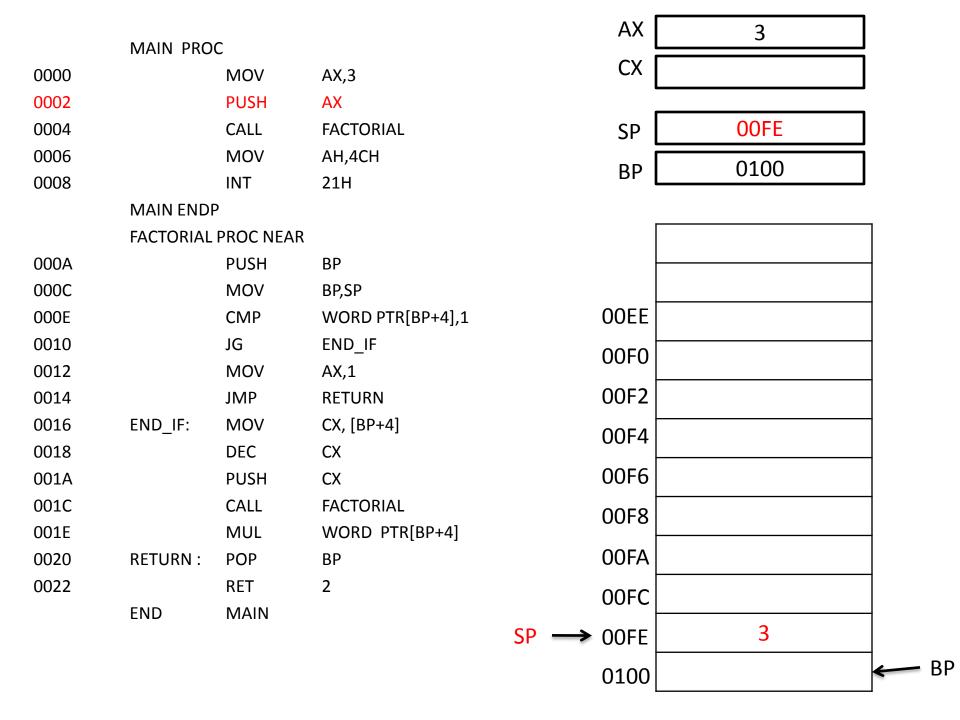
• Factorial(n)=n! =n*(n-1)*(n-2)*.....*3*2*1

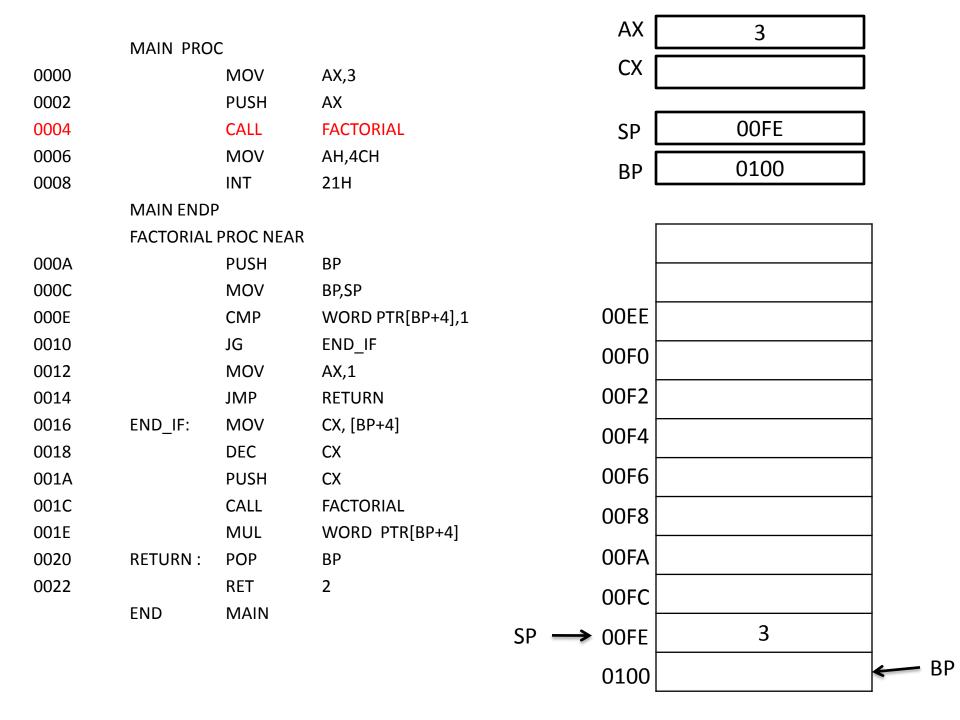
Factorial(n)=n*Factorial(n-1)

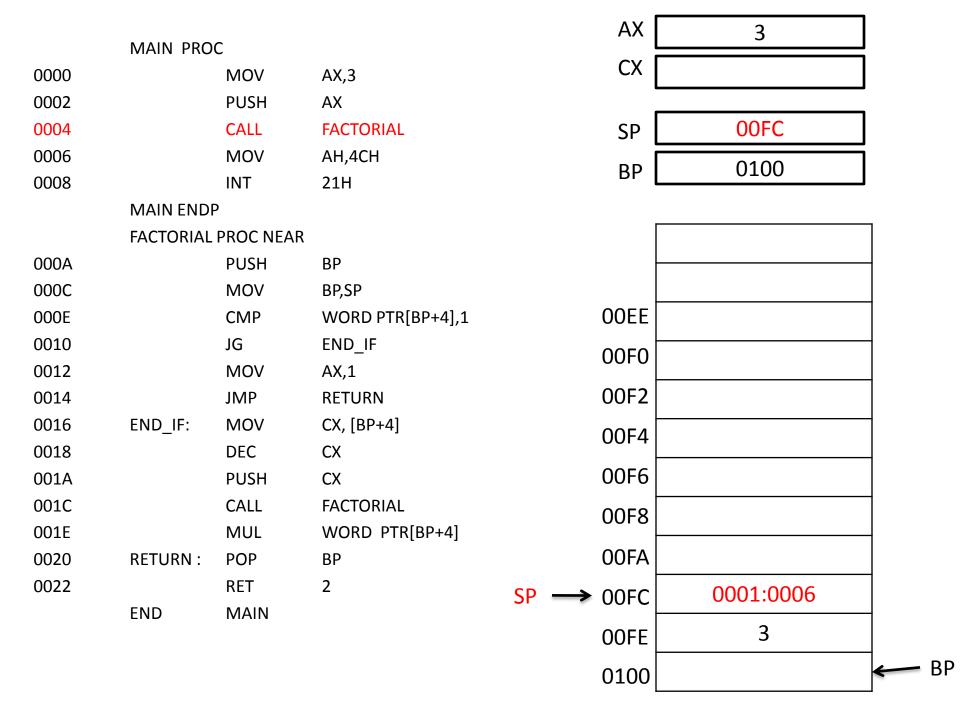
				AX		
	MAIN PRO	C				
0000		MOV	AX,3	CX		
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	0100	
0006		MOV	AH,4CH	أمم	0100	
8000		INT	21H	BP	0100	
	MAIN END	Р				
	FACTORIAL	. PROC NEAF	3]
000A		PUSH	ВР			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		1
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	00F4		1
0018		DEC	CX			
001A		PUSH	CX	00F6		
001C		CALL	FACTORIAL	00F8		1
001E		MUL	WORD PTR[BP+4]			1
0020	RETURN:	POP	ВР	00FA		
0022		RET	2	00FC		
	END	MAIN		ľ		
				00FE		
				SP → 0100		← BP



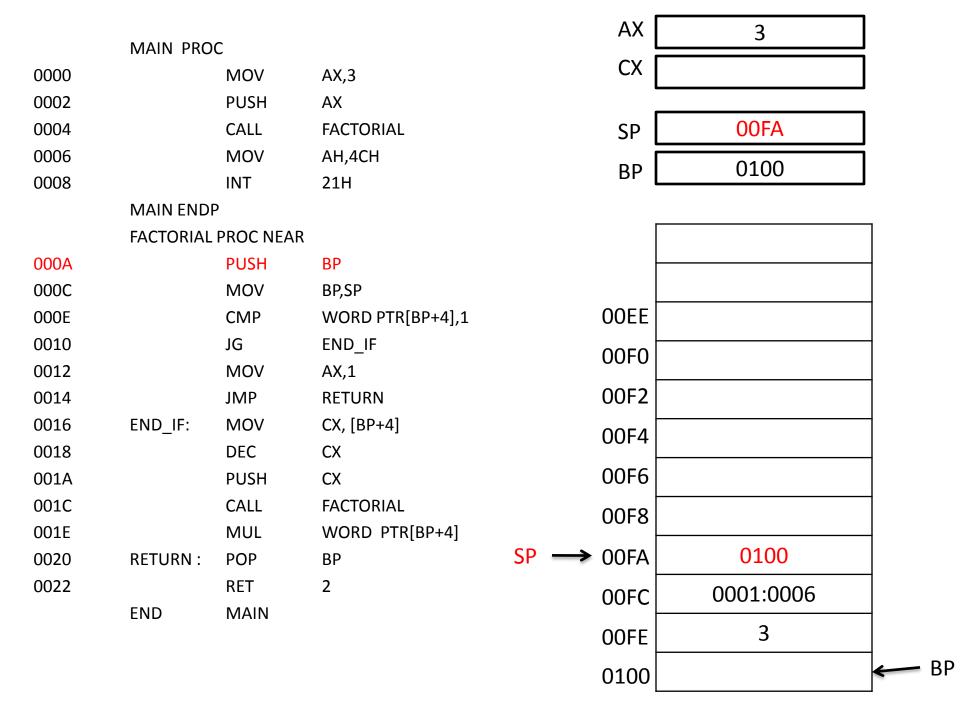
				AX	3	
	MAIN PRO			CX		
0000		MOV	AX,3	CA		
0002		PUSH	AX	1		
0004		CALL	FACTORIAL	SP	0100	
0006		MOV	AH,4CH	ВР	0100	
8000		INT	21H	DP	0100	
	MAIN END	Р				
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		1
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	00F4		-
0018		DEC	CX	0014		
001A		PUSH	CX	00F6		
001C		CALL	FACTORIAL	00F8		1
001E		MUL	WORD PTR[BP+4]	0018		-
0020	RETURN:	POP	BP	00FA		
0022		RET	2	00FC		
	END	MAIN		OUFC		-
				00FE		
				SP → 0100		← BP



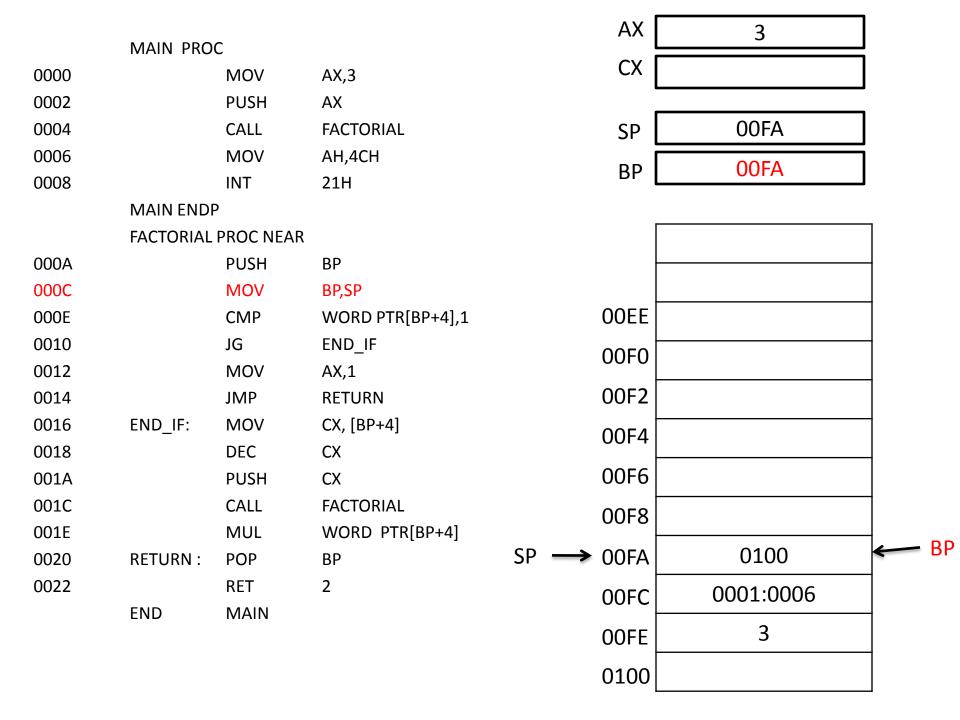


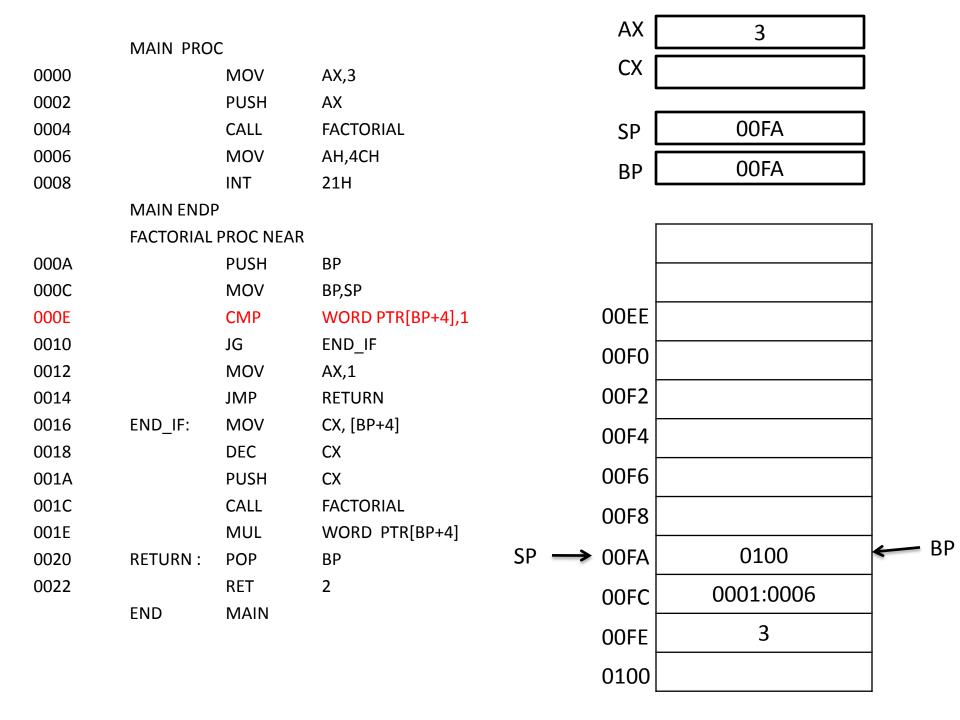


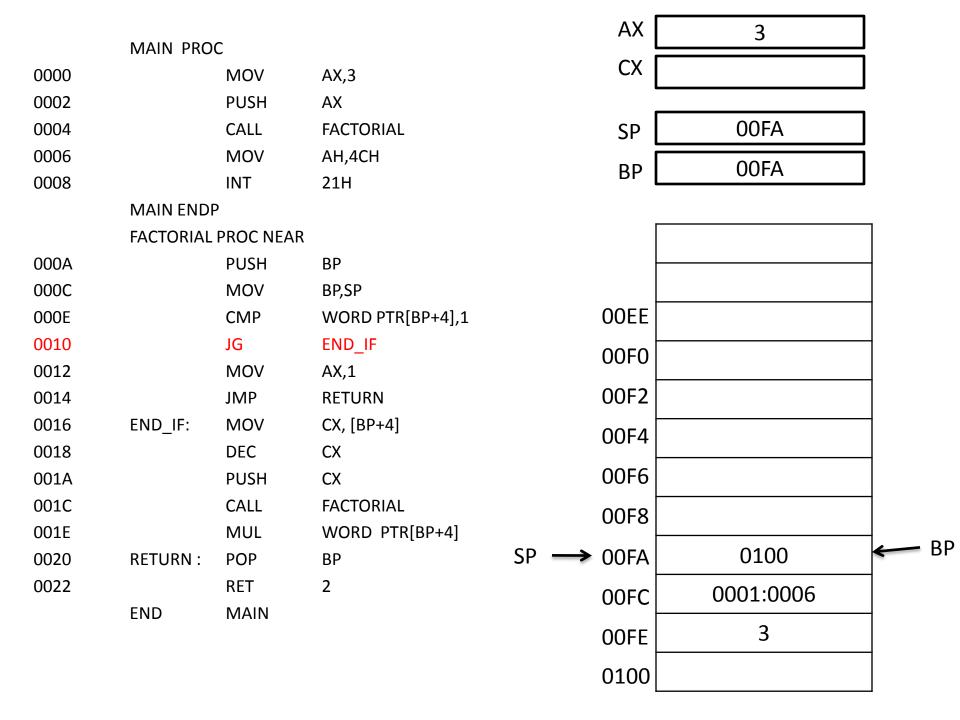
				AV !		
	MAIN PRO)C		AX [3	
0000		MOV	AX,3	CX		
0002		PUSH	AX	•		
0004		CALL	FACTORIAL	SP	00FC	
0006		MOV	AH,4CH	i	0100	
8000		INT	21H	BP (0100	
	MAIN END	Р				
	FACTORIAL	PROC NEAR]
000A		PUSH	BP			┦
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		†
0012		MOV	AX,1			_
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	00F4		1
0018		DEC	CX			-
001A		PUSH	CX	00F6		
001C		CALL	FACTORIAL	00F8		7
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA		_
0022		RET	2	SP → 00FC	0001:0006	
	END	MAIN			3	┪
				00FE	<u> </u>	
				0100		← BP
				·		

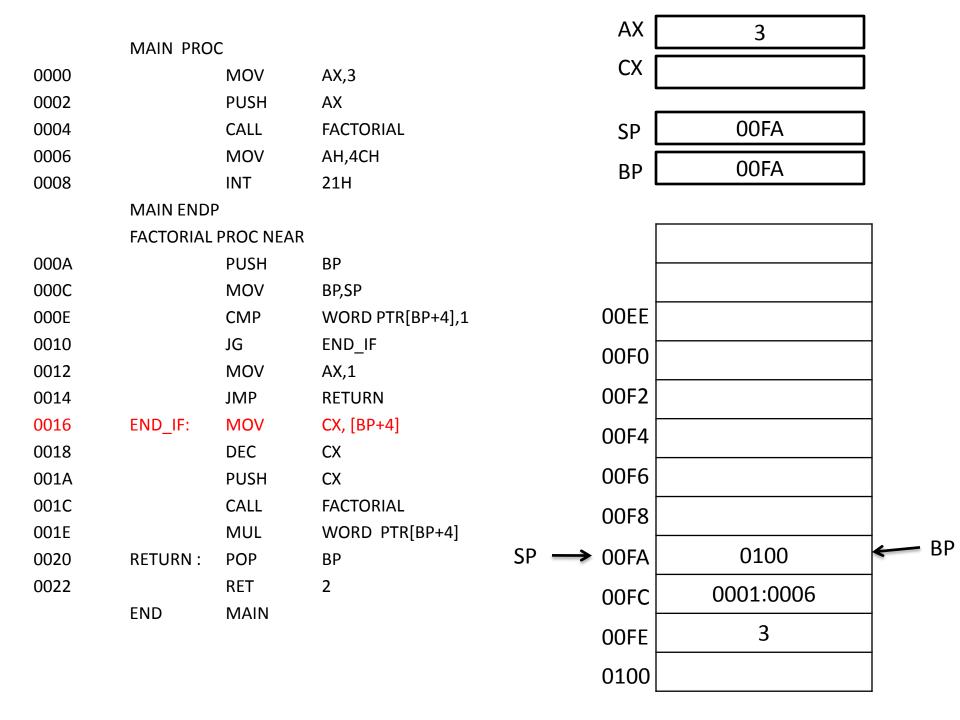


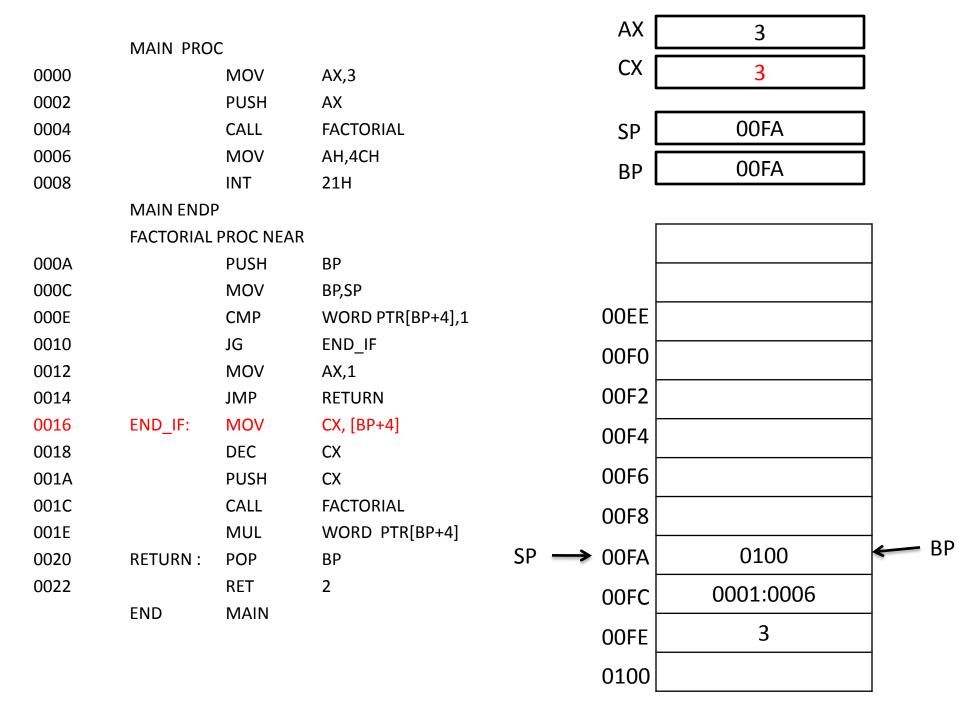
				AX	3	
	MAIN PRO	С		CV		
0000		MOV	AX,3	CX		
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	00FA	
0006		MOV	AH,4CH	D.D.	0100	
8000		INT	21H	ВР	0100	
	MAIN END	Р				
	FACTORIAL	PROC NEAR]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	0050		-
0012		MOV	AX,1	00F0		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	0054		-
0018		DEC	CX	00F4		
001A		PUSH	CX	00F6		
001C		CALL	FACTORIAL	00F8		1
001E		MUL	WORD PTR[BP+4]	0018		_
0020	RETURN:	POP	ВР	SP → 00FA	0100	
0022		RET	2	00FC	0001:0006	1
	END	MAIN		UUFC		-
				00FE	3	
				0100		← BP
						_

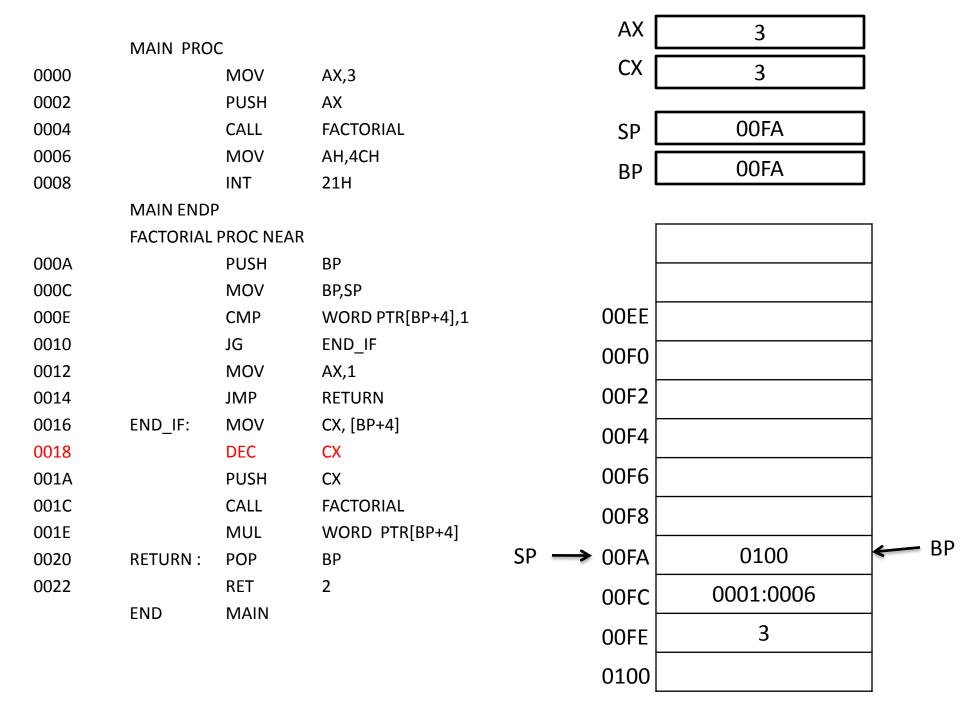


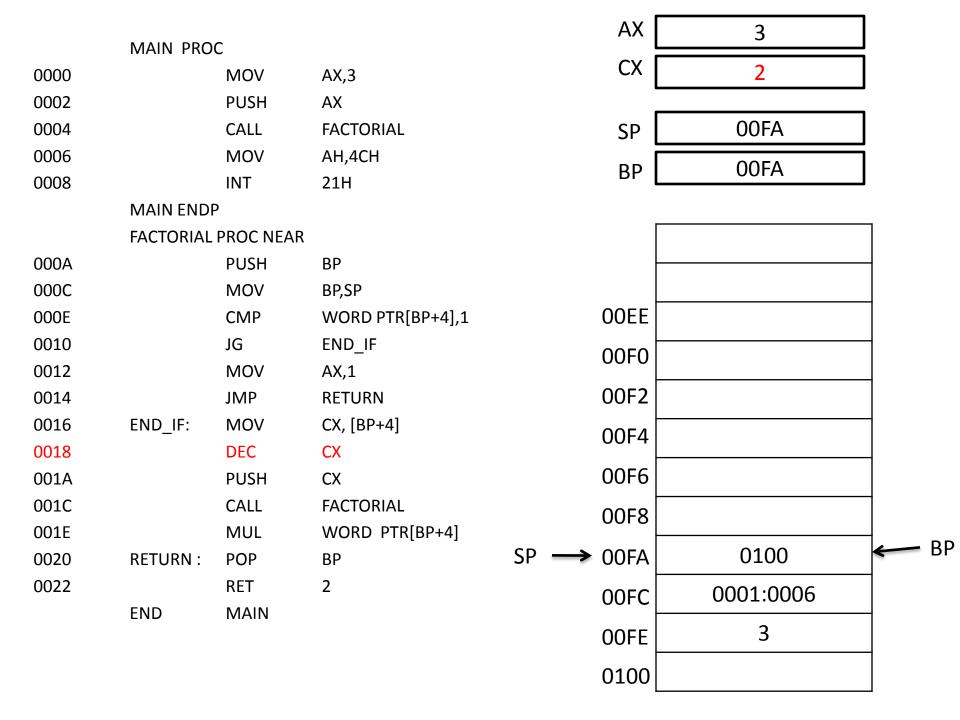


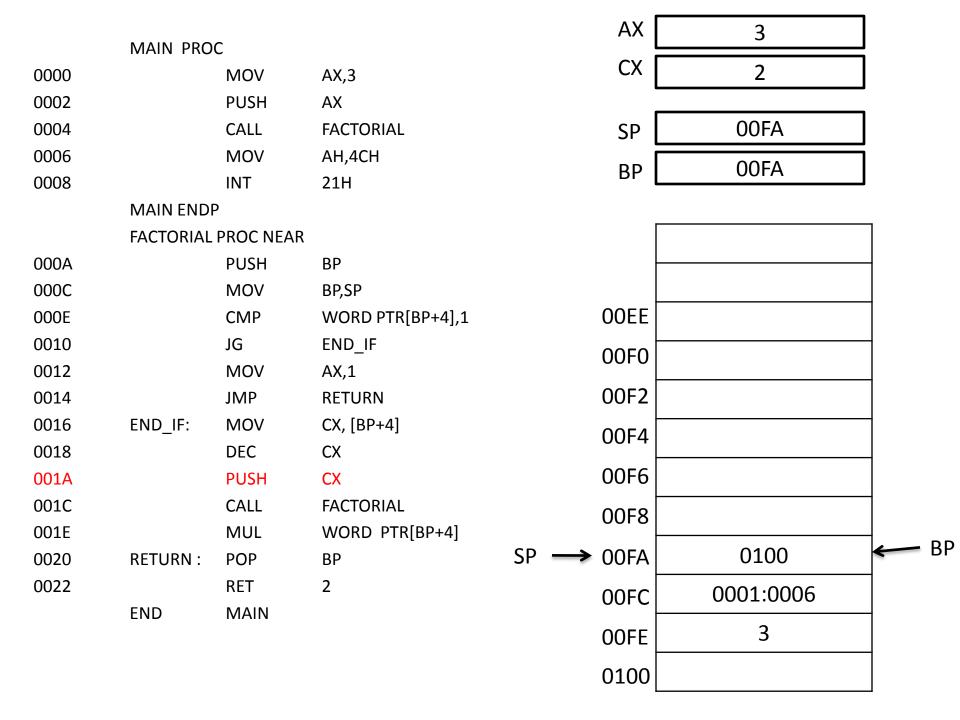


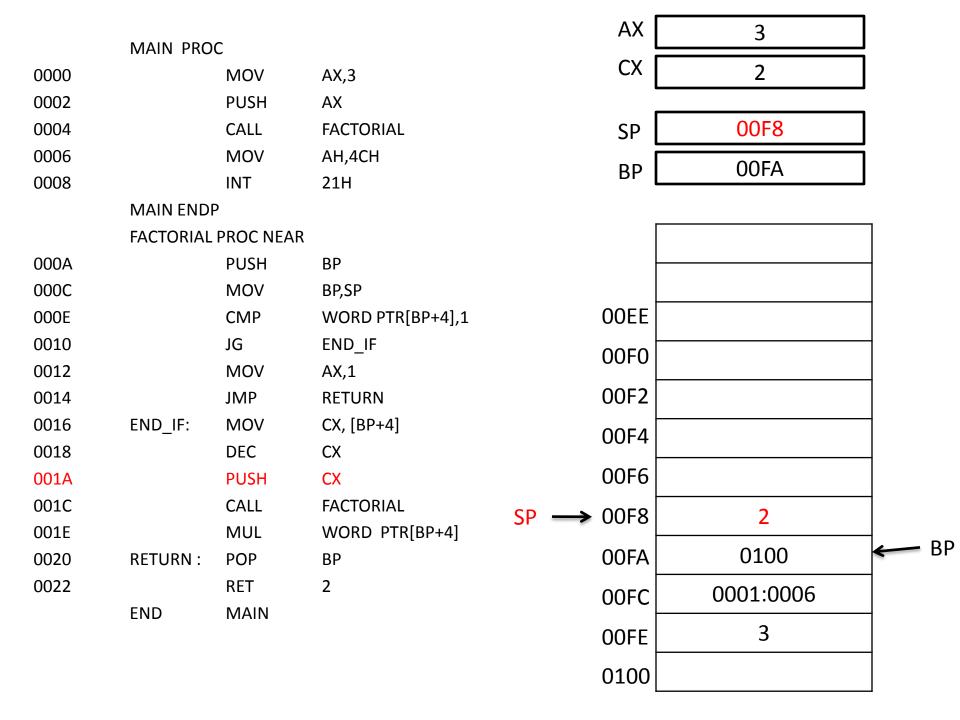


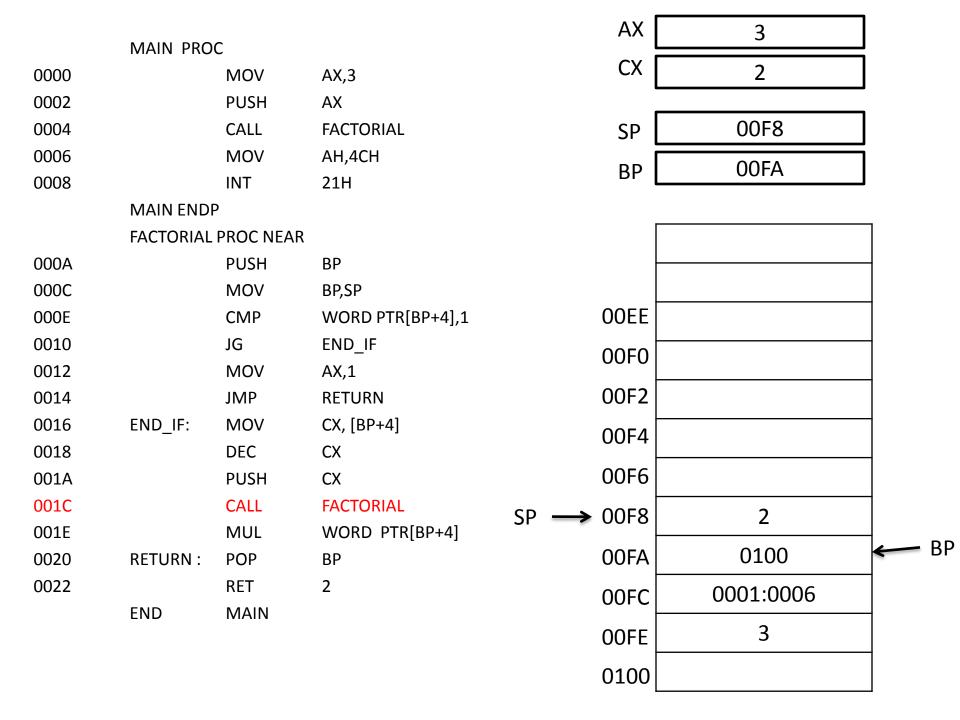


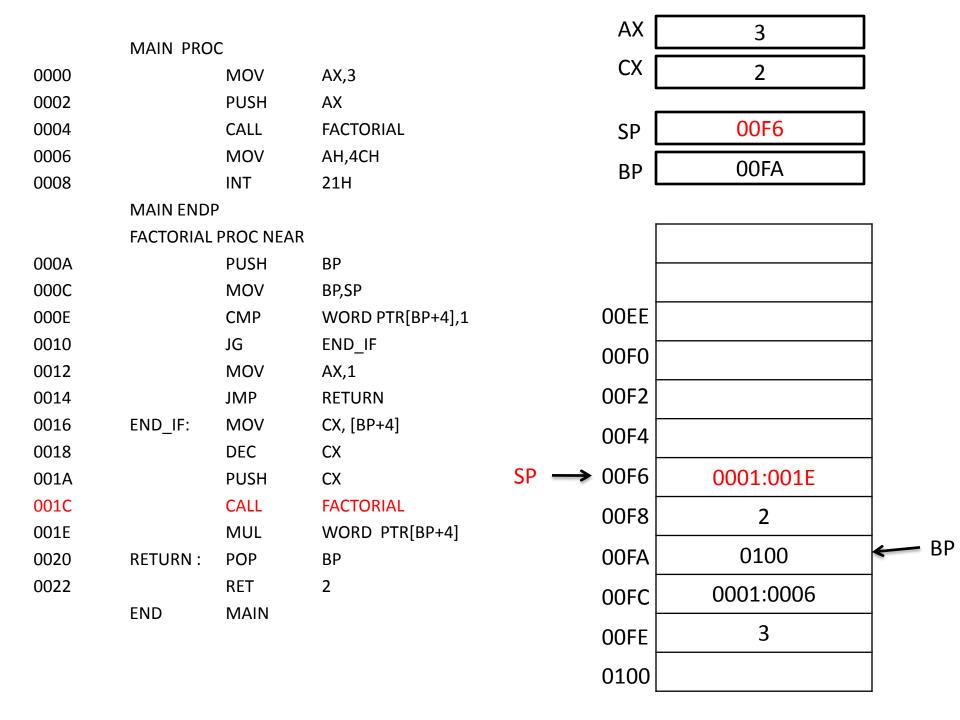




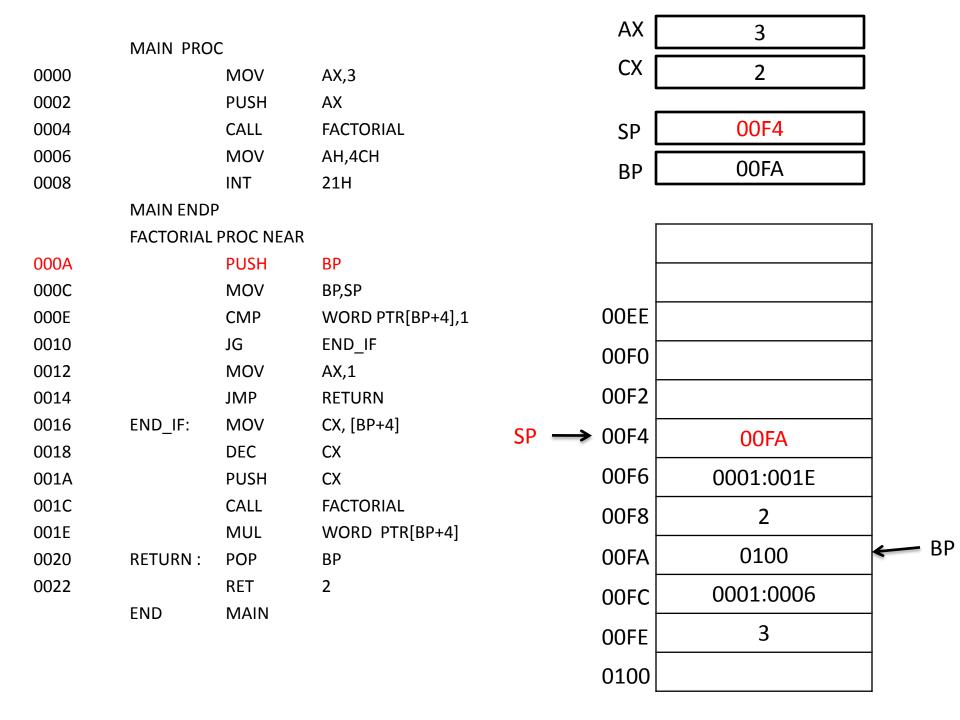


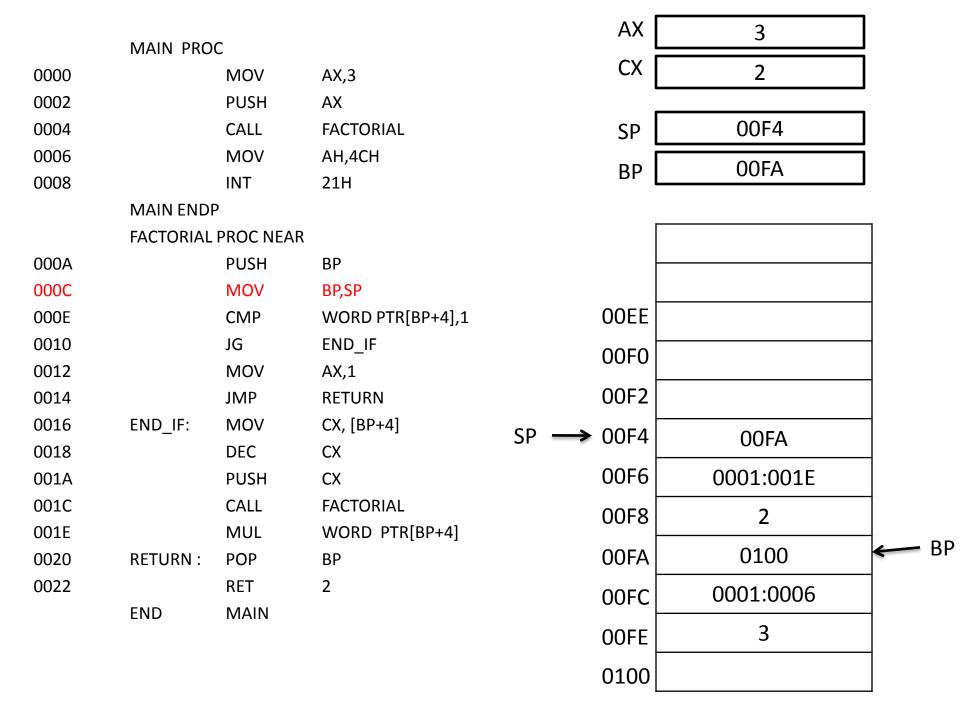






	NAAINI DDO	C		AX	3	
0000	MAIN PRO	MOV	AX,3	CX	2	
0002		PUSH	AX			
				25	0050	
0004		CALL	FACTORIAL	SP	00F6	
0006		MOV	AH,4CH	BP	00FA	
8000		INT	21H	ן אט	00171	
	MAIN END	P				
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	00F4		-
0018		DEC	CX	001 4		_
001A		PUSH	CX	$SP \longrightarrow 00F6$	0001:001E	
001C		CALL	FACTORIAL	00F8	2	1
001E		MUL	WORD PTR[BP+4]			₩ BP
0020	RETURN :	POP	BP	00FA	0100	DP
0022		RET	2	00FC	0001:0006	
	END	MAIN			2	-
				00FE	3	
				0100		
						_





		_		AX	3	
	MAIN PRO			CV	_	
0000		MOV	AX,3	CX	2	
0002		PUSH	AX	•		
0004		CALL	FACTORIAL	SP	00F4	
0006		MOV	AH,4CH	55	0054	
8000		INT	21H	BP	00F4	
	MAIN END	P				
	FACTORIAL	PROC NEAR	1			
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	← BP
0018		DEC	CX	51 2 001 4	UUFA	DI
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			_
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			2	-
				00FE	3	
				0100		

			AX	3	
MAIN PR	OC			3	
0000	MOV	AX,3	CX	2	
0002	PUSH	AX	-		
0004	CALL	FACTORIAL	SP	00F4	
0006	MOV	AH,4CH	DD.	00F4	
0008	INT	21H	BP	0074	
MAIN ENI	OP				_
FACTORIA	L PROC NEAI	R			
000A	PUSH	ВР			-
000C	MOV	BP,SP			_
000E	CMP	WORD PTR[BP+4],1	00EE		
0010	JG	END_IF	00F0		┥
0012	MOV	AX,1			-
0014	JMP	RETURN	00F2		
0016 END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	← BP
0018	DEC	CX			- 5.
001A	PUSH	CX	00F6	0001:001E	
001C	CALL	FACTORIAL	00F8	2]
001E	MUL	WORD PTR[BP+4]			-
0020 RETURN :	POP	ВР	00FA	0100	
0022	RET	2	00FC	0001:0006	
END	MAIN		00FE	3	1
					-
			0100		

			AX	2	
MAIN PRO	C			3	
0000	MOV	AX,3	CX	2	
0002	PUSH	AX			
0004	CALL	FACTORIAL	SP	00F4	
0006	MOV	AH,4CH	DD	00F4	
0008	INT	21H	BP	UUF 4	
MAIN ENDP	•				
FACTORIAL	PROC NEAR]
000A	PUSH	ВР			-
000C	MOV	BP,SP			
000E	CMP	WORD PTR[BP+4],1	00EE		
0010	JG	END_IF	00F0		1
0012	MOV	AX,1]
0014	JMP	RETURN	00F2		
0016 END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018	DEC	CX			-
001A	PUSH	CX	00F6	0001:001E	
001C	CALL	FACTORIAL	00F8	2	
001E	MUL	WORD PTR[BP+4]			-
0020 RETURN:	POP	ВР	00FA	0100	
0022	RET	2	00FC	0001:0006	
END	MAIN		00FE	3	1
					1
			0100]

	MAIN PRO			AX	3	
0000	IVIAIIV PNO	MOV	AX,3	CX	2	
0002		PUSH	AX	l		
0002		CALL	FACTORIAL	SP	00F4	
0004		MOV	AH,4CH	3r	001 7	
				ВР	00F4	
8000		INT	21H	,		
	MAIN ENDI			1		ا ا
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX		UUIA	-
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN				-
				OOFE	3	
				0100		
						_

	MAIN PRO			AX	3	
0000	MAIN PRO	MOV	AX,3	СХ	2	
0002		PUSH	AX		_	
0002		CALL	FACTORIAL	CD	00F4	
				SP	0074	
0006		MOV	AH,4CH	BP	00F4	
8000		INT	21H	וט		
	MAIN END	P				_
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			_
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		_
0012		MOV	AX,1	UUFU		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX	31 - 0014	UUIA	- 51
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			_
0020	RETURN:	POP	ВР	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN				_
				00FE	3	_
				0100		
				'		-

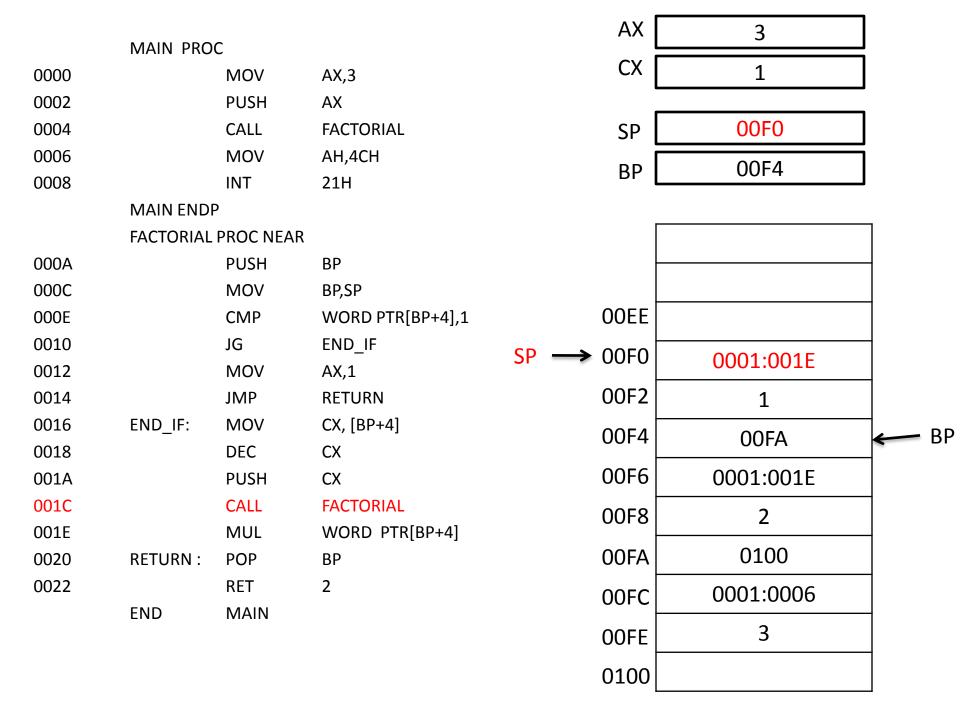
	MAIN PRO			AX	3	
0000	IVIAIIN PNO	MOV	AX,3	CX	2	
0002		PUSH	AX	l		
0002		CALL	FACTORIAL	SP	00F4	
0004		MOV	AH,4CH	3r	001 7	
				ВР	00F4	
8000	244121 5215	INT	21H	,		
	MAIN END			1		٦
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			1
000C		MOV	BP,SP]
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX		UUFA	- 01
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			_
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN				
				OOFE	3	
				0100		
				1		

	MAIN PRO			AX	3	
0000	IVIAIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX		_	
0004		CALL	FACTORIAL	SP	00F4	
0004		MOV	AH,4CH	3F		
0008		INT	21H	BP	00F4	
0000	MAIN END		2111			
		r . PROC NEAR				7
0004	FACTORIAL					
000A		PUSH	BP			
000C		MOV	BP,SP	2255		_
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		_
0012		MOV	AX,1	0010		
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX		UUIA	- 51
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			_
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			3	_
				00FE	3	
				0100		
				•		

	MAINI DDO			AX	3	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	00F4	
0004		MOV	AH,4CH	3F		
0008		INT	21H	BP	00F4	
0000	MAIN END		2111			
		r . PROC NEAR		1		1
000A	FACTORIAL	PUSH	ВР			
000C		MOV	BP,SP	0055		1
000E		CMP	WORD PTR[BP+4],1	OOEE		
0010		JG	END_IF	00F0		
0012		MOV	AX,1			-
0014		JMP	RETURN	00F2		
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	← BP
0018		DEC	CX		0017	- 5.
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			1
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN				1
				00FE	3	
				0100		
				•		_

				,		
	MAIN PRO	·C		AX	3	
0000	IVIAIIV FNO	MOV	AX,3	CX	1	
0002		PUSH	AX	•	_	
0004		CALL	FACTORIAL	SP	00F2	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN ENDI					
	FACTORIAL	PROC NEAR]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE]
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	UUFU		
0014		JMP	RETURN	SP → 00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	BP
0018		DEC	CX		UUFA	- 01
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	1
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			3	†
				00FE	J	-
				0100		
				•		_

	MAIN PRO	\C		AX	3	
0000	IVIAIIN PNO	MOV	AX,3	СХ	1	
0002		PUSH	AX	I	_	
0004		CALL	FACTORIAL	SP	00F2	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN ENDI	Р				
	FACTORIAL	PROC NEAR	\]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE		
0010		JG	END_IF	00F0		┥ ╿
0012		MOV	AX,1	0010		_
0014		JMP	RETURN	$SP \longrightarrow 00F2$	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	BP
0018		DEC	CX			
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022	END		2	00FC	0001:0006	
	END	MAIN		00FE	3]
				0100		
0022	END	RET MAIN	2	00FC 00FE	0001:0006	_



MAIN PROC MOV AX,3 CX	1
CV	1
0002 PUSH AX	
	00F0
0006 MOV AH.4CH	
0008 INT 21H BP	00F4
MAIN ENDP	
FACTORIAL PROC NEAR	
000A PUSH BP	
000C MOV BP,SP	
000E CMP WORD PTR[BP+4],1 00EE	
$\begin{array}{cccccccccccccccccccccccccccccccccccc$	01·001E
0012 MOV AX,1	01:001E
0014 JMP RETURN 00F2	1
0016 END_IF: MOV CX, [BP+4] 00F4	00FA ← BP
0018 DEC CX	
	01:001E
001C CALL FACTORIAL 00F8	2
001E MUL WORD PTR[BP+4]	0100
	0100
0022 RET 2 00FC 000	01:0006
OOFE	3
0100	

MAIN PROC 0000 MOV AX,3 AX 3 CX 1	
CV	
0002 PUSH AX	
0004 CALL FACTORIAL SP OOEE	
0006 MOV AH.4CH	
0008 INT 21H BP 00F4	
MAIN ENDP	
FACTORIAL PROC NEAR	
000A PUSH BP	
000C MOV BP,SP	
000E CMP WORD PTR[BP+4],1 SP \longrightarrow 00EE 00F4	
0010 JG END IF	
0012 MOV AX,1	
0014 JMP RETURN 00F2 1	
0016 END_IF: MOV CX, [BP+4] 00F4 00FA	← BP
0018 DEC CX	J.
001A PUSH CX 00F6 0001:001E	
001C CALL FACTORIAL 00F8 2	
001E MUL WORD PTR[BP+4]	
0020 RETURN: POP BP 00FA 0100	1
0022 RET 2 00FC 0001:0006	
END MAIN 00FE 3	
0100	

						,
	MAIN PRO			AX	3	
0000	IVIAIIN PNO	MOV	AX,3	CX	1	
0002		PUSH	AX	l	_	
0004		CALL	FACTORIAL	SP	00EE	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN END					
	FACTORIAL	PROC NEAR]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	SP → 00EE	00F4]
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	BP
0018		DEC	CX			-
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	<u> </u>
0022	5115	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	1
						-
				0100		

	MAIN PRO	C		AX	3	
0000	IVIAIIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX		_	
0004		CALL	FACTORIAL	SP	00EE	
0004		MOV	AH,4CH	3F	OOLL	
0008		INT	21H	ВР	00EE	
0008	MAIN END		2111	•		
		PROC NEAR	,			7
0004	FACTORIAL					
000A		PUSH	BP			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	$SP \longrightarrow 00EE$	00F4	← BP
0010		JG	END_IF	00F0	0001:001E	1
0012		MOV	AX,1	0010	0001:001E	_
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	1
0018		DEC	CX	0014	UUFA	_
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	1
				UUFE	<u> </u>	-
				0100		

	MAIN DDO			AX	3	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0002		PUSH	AX			
0002		CALL	FACTORIAL	SP	00EE	
				38	OOLL	
0006		MOV	AH,4CH	ВР	00EE	
8000		INT	21H			
	MAIN END			,		-
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	$SP \longrightarrow 00EE$	00F4	← BP
0010		JG	END_IF	00F0		_
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	_
0018		DEC	CX	0014	UUFA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			2	_
				00FE	3	_
				0100		

	MAIN DDO			AX	3	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0002		PUSH	AX			
0002		CALL	FACTORIAL	SP	00EE	
				38	OOLL	
0006		MOV	AH,4CH	ВР	00EE	
8000		INT	21H			
	MAIN END	Р		,		_
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	SP → 00EE	00F4	← BP
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	UUFU	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	1
0018		DEC	CX	0014	UUFA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			2	-
				00FE	3	
				0100		

	MAIN DDO			AX	3	
0000	MAIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX			
0002		CALL	FACTORIAL	cn l	00EE	
				SP	UULL	
0006		MOV	AH,4CH	BP	00EE	
8000		INT	21H	5. (
	MAIN END	Р				_
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			_
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	$SP \longrightarrow 00EE$	00F4	← BP
0010		JG	END_IF	00F0		_
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	_
0018		DEC	CX	001 4	UUFA	_
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			_
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			2	_
				00FE	3	_
				0100		

	MAIN PRO			AX	1	
0000	IVIAIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX		_	
0004		CALL	FACTORIAL	SP	00EE	
0004		MOV	AH,4CH	34	OOLL	
0008		INT	21H	BP	00EE	
0006			2111	•		
	MAIN END			1		1
	FACTORIAL	PROC NEAF				
000A		PUSH	BP			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	$SP \longrightarrow 00EE$	00F4	← BP
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	UUFU	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	1
0018		DEC	CX	0014	UUFA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	ВР	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			3	-
				00FE	3	_
				0100		

	MAIN DDO			AX	1	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	00EE	
0004		MOV	AH,4CH	Jr		
0008		INT	21H	BP	00EE	
0000	MAIN END		2111			
		r . PROC NEAR				1
000A	TACTORIAL	PUSH	BP			
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	SP → 00EE	0054	← BP
0010		JG	END_IF	Ji - OOLL	00F4	Di
0010		MOV	AX,1	00F0	0001:001E	
0012		JMP	RETURN	00F2	1	-
0014	END IF:	MOV	CX, [BP+4]	001 2	1	
0018	END_IF.	DEC	CX, [BP+4]	00F4	00FA	
0018 001A		PUSH	CX	00F6	0001:001E	1
001A 001C		CALL	FACTORIAL	0010	0001.001L	_
001C 001E		MUL		00F8	2	
0016	DETLIDAL		WORD PTR[BP+4] BP	00FA	0100	
	RETURN :	POP		001 A		_
0022	END	RET MAIN	2	00FC	0001:0006	
	EIND	IVIAIIN		00FE	3	
				0100		
						_

	NAAINI DDO	C		AX	1	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	00EE	
0004		MOV	AH,4CH	Jr		
0008		INT	21H	BP	00EE	
0000	MAIN ENDI		2111			
		PROC NEAR		I		1
000A	TACTORIAL	PUSH	ВР			
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	SP → 00EE	0054	← BP
0010		JG	END_IF	Si POOLL	00F4	Di
0010		MOV	AX,1	00F0	0001:001E	
0012		JMP	RETURN	00F2	1	1
0014	END_IF:	MOV	CX, [BP+4]	0012	1	1
0018	END_IF.	DEC	СХ, [БР+4] СХ	00F4	00FA	
0018 001A			CX	00F6	0001:001E	1
		PUSH		0010	0001.0016	1
001C		CALL	FACTORIAL	00F8	2	
001E	DETUDN.	MUL	WORD PTR[BP+4]	00FA	0100	1
0020	RETURN :	POP	BP	UUFA	0100	-
0022	END	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	
						†
				0100]

	MAIN PRO	·C		AX [1	
0000	IVIAIN FRO	MOV	AX,3	CX	1	
0002		PUSH	AX	l		'
0004		CALL	FACTORIAL	SP	00F0	
0006		MOV	AH,4CH	ſ		<u>'</u>
0008		INT	21H	BP [00F4	
	MAIN END	Р				
	FACTORIAL	. PROC NEAR	₹			
000A		PUSH	ВР	ļ		-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	SP → 00F0	0001:0015	
0012		MOV	AX,1	-	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	BP
0018		DEC	CX	-		-
001A		PUSH	CX	00F6	0001:001E	
		CALL	FACTORIAL	00F8	2	
				-	0100	-
	RETURN:			UUFA	0100	_
0022	ראום		2	00FC	0001:0006	
	END	IVIAIIN		00FE	3	7
				0100		
001C 001E 0020 0022	RETURN : END	CALL MUL POP RET MAIN	FACTORIAL WORD PTR[BP+4] BP 2	00FA 00FC 00FE	0100 0001:0006	- - -

MAIN PROC 0000 MOV AX,3 0002 PUSH AX	
0000 MOV AX,3 CX 1	Ī,
	_
υισε γυρη Αλ	,
0004 CALL FACTORIAL SP 00F0	7
0006 MOV AH.4CH	╡ '
0008 INT 21H BP 00F4	
MAIN ENDP	
FACTORIAL PROC NEAR	
000A PUSH BP	_
000C MOV BP,SP	!
000E CMP WORD PTR[BP+4],1 00EE 00F4	!
0010 JG END IF	-
0012 MOV AX,1	
0014 JMP RETURN 00F2 1	!
0016 END_IF: MOV CX, [BP+4] 00F4 00FA	BP
0018 DEC CX	
001A PUSH CX 00F6 0001:001E	
001C CALL FACTORIAL 00F8 2	
001E MUL WORD PTR[BP+4]	-
0020 RETURN: POP BP 00FA 0100	
0022 RET 2 00FC 0001:0006	!
END MAIN 00FE 3	
0100	

	MAIN PRO	·C		AX	1	
0000	IVIAIIV PNO	MOV	AX,3	CX	1	
0002		PUSH	AX	l	_	
0004		CALL	FACTORIAL	SP	00F4	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN ENDI					
	FACTORIAL	. PROC NEAR	₹			7
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	7
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010	0001:001E	<u> </u>
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX			-
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022	ENID.	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	
				0100		-
				0100		

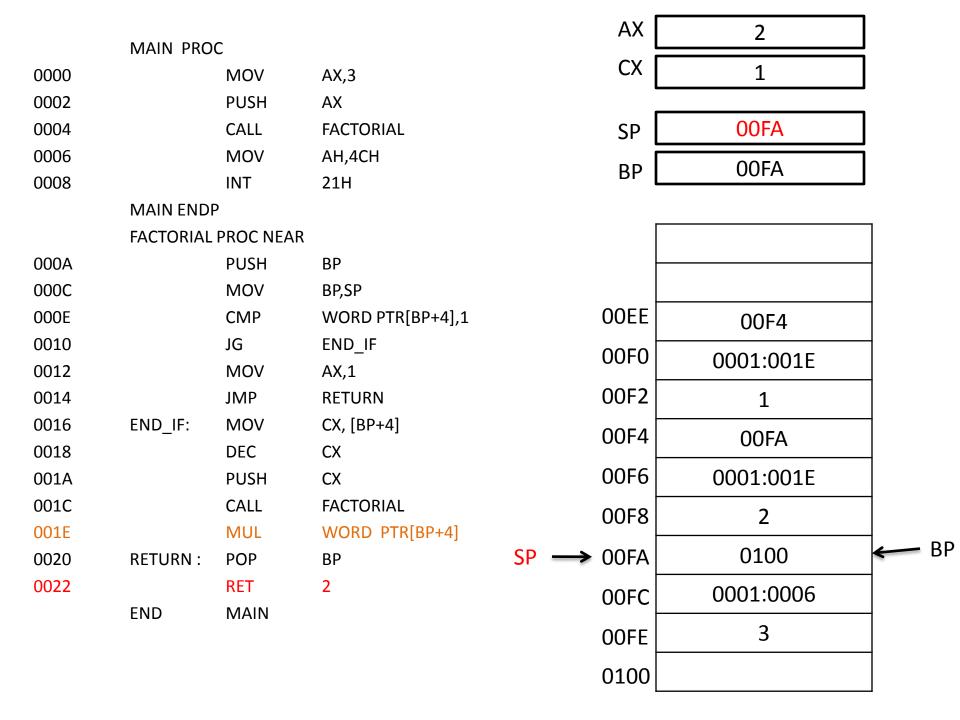
	MAIN PRO	·C		AX	1	
0000	IVIAIN PNO	MOV	AX,3	CX	1	
0002		PUSH	AX	l		
0004		CALL	FACTORIAL	SP	00F4	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN ENDI					
	FACTORIAL	PROC NEAR	k	1		7
000A		PUSH	ВР			_
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4]
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	← BP
0018		DEC	CX			- 01
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022	5445	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	
						-
				0100		_

				,		. ,
	MAIN PRO	\C		AX	2	
0000	IVIAIIV FNO	MOV	AX,3	CX	1	
0002		PUSH	AX	'		J
0004		CALL	FACTORIAL	SP	00F4	
0006		MOV	AH,4CH			
0008		INT	21H	BP	00F4	
	MAIN ENDI	Р				
	FACTORIAL	PROC NEAR	\	1		7
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0		-
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX			-
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022	END	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	1
				0100		
					3	

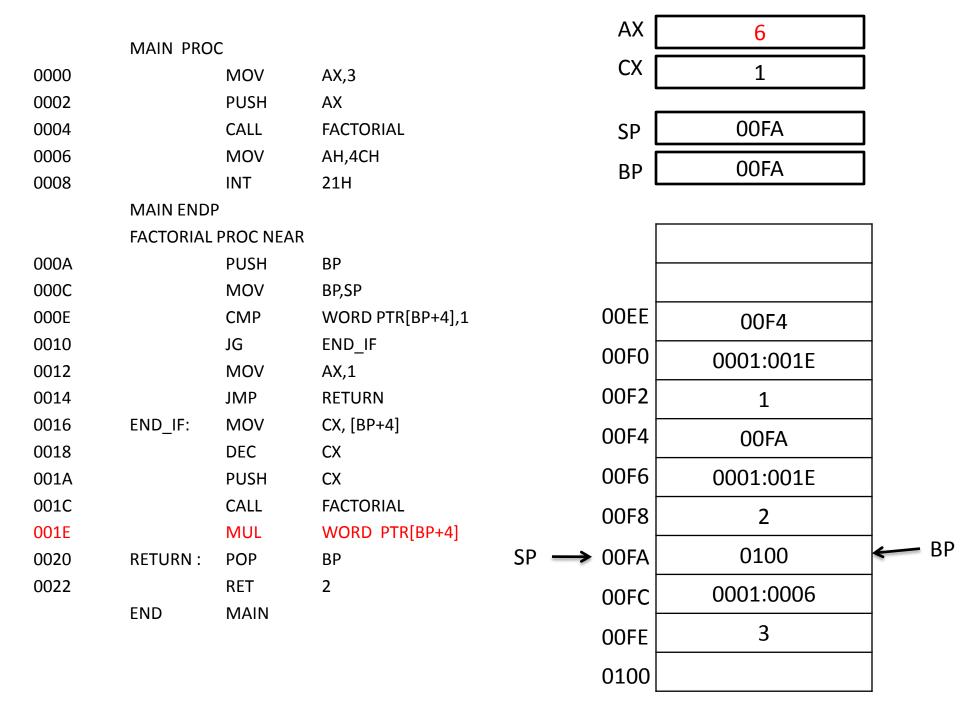
				AX	2	
	MAIN PRO	C		CV		
0000		MOV	AX,3	CX	1	
0002		PUSH	AX			ı
0004		CALL	FACTORIAL	SP	00F4	
0006		MOV	AH,4CH	חח	00F4	
8000		INT	21H	BP	0074	
	MAIN END	Р				
	FACTORIAL	PROC NEA	R]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0	0001.0015	†
0012		MOV	AX,1	0010	0001:001E	_
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	SP → 00F4	00FA	BP
0018		DEC	CX			-
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022	5115	RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	1
				0100		-
				0100		

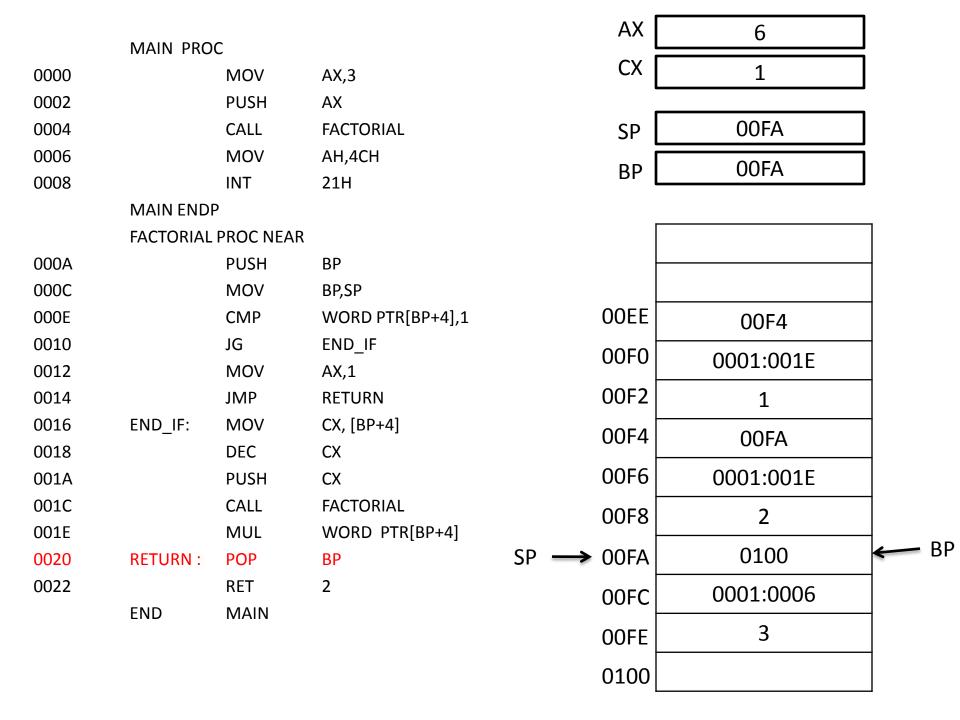
	MAINI DDC	\ C		AX [2	
0000	MAIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX,3	ı		
0002		CALL	FACTORIAL	SP	00F6	
0004		MOV		or	0010	
			AH,4CH	ВР	00FA	
8000	244445	INT	21H	•		1
	MAIN ENDI					¬
	FACTORIAL	PROC NEAR	i			
000A		PUSH	BP	}		†
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0	0001.001E	†
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	7
0018		DEC	CX	-		-
001A		PUSH	CX	SP → 00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	7
001E		MUL	WORD PTR[BP+4]	-		BP
0020	RETURN:	POP	BP	00FA	0100	וט
0022		RET	2	00FC	0001:0006	
	END	MAIN		0055	3	†
				00FE	5	4
				0100		
				L		_

	NAAINI DDO	.		AX	2	
0000	MAIN PRO	MOV	AX,3	СХ	1	
0000		PUSH	AX,3		<u> </u>	
0002		CALL	FACTORIAL	cn l	00F6	
				SP	0010	
0006		MOV	AH,4CH	BP	00FA	
8000		INT	21H			
	MAIN END			,		,
	FACTORIAL	. PROC NEAF	₹			
000A		PUSH	ВР			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0	0001.0015	-
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	
0018		DEC	CX		0017	
001A		PUSH	CX	SP → 00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]	20-1	04.00	BP
0020	RETURN :	POP	BP	00FA	0100	DI
0022		RET	2	00FC	0001:0006	
	END	MAIN		00FE	3	
				UUFE	3	-
				0100		



	MAIN PRO	NC		AX	2	
0000	IVIAIN PRO	MOV	AX,3	CX	1	
0002		PUSH	AX	·	_	
0004		CALL	FACTORIAL	SP	00FA	
0006		MOV	AH,4CH			
0008		INT	21H	BP (00FA	
	MAIN END	Р				
	FACTORIAL	PROC NEAF	R]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	7
0010		JG	END_IF	00F0		+
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	†
0018		DEC	CX			4
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			→ BP
0020	RETURN :	POP	ВР	$SP \longrightarrow 00FA$	0100	Di
0022		RET	2	00FC	0001:0006	
	END	MAIN			3	1
				00FE	J	4
				0100		





		_		AX [6
	MAIN PRO			сх [1
0000		MOV	AX,3		1
0002		PUSH	AX	Г	
0004		CALL	FACTORIAL	SP L	00FC
0006		MOV	AH,4CH	вр [0100
8000		INT	21H	DP [0100
	MAIN END	Р		_	
	FACTORIAL	PROC NEAR			
000A		PUSH	BP	-	
000C		MOV	BP,SP		
000E		CMP	WORD PTR[BP+4],1	00EE	00F4
0010		JG	END_IF	00F0	0001.001
0012		MOV	AX,1	_	0001:001E
0014		JMP	RETURN	00F2	1
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA
0018		DEC	CX	-	001A
001A		PUSH	CX	00F6	0001:001E
001C		CALL	FACTORIAL	00F8	2
001E		MUL	WORD PTR[BP+4]	-	
0020	RETURN:	POP	ВР	00FA _	0100
0022		RET	2	SP → 00FC	0001:0006
	END	MAIN		-	
				00FE	3
				0100	

BP

		_		AX	6	
	MAIN PRO			CV		
0000		MOV	AX,3	CX	1	
0002		PUSH	AX	1		
0004		CALL	FACTORIAL	SP	00FC	
0006		MOV	AH,4CH	DD	0100	
8000		INT	21H	BP	0100	
	MAIN END	Р				
	FACTORIAL	PROC NEAR				
000A		PUSH	BP			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	0050		-
0012		MOV	AX,1	00F0	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	0054	-
0018		DEC	CX	0014	00FA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	1
001E		MUL	WORD PTR[BP+4]			-
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	SP → 00FC	0001:0006	
	END	MAIN			2	-
				00FE	3	_
				0100		← BP

				AX	6	
	MAIN PRO	С		CV		
0000		MOV	AX,3	CX	1	
0002		PUSH	AX	•		
0004		CALL	FACTORIAL	SP	0100	
0006		MOV	AH,4CH	חח	0100	
8000		INT	21H	BP	0100	
	MAIN ENDI)				
	FACTORIAL	PROC NEAR]
000A		PUSH	ВР			-
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0		_
0012		MOV	AX,1	UUFU	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	0054	1
0018		DEC	CX	0014	00FA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			1
0020	RETURN :	POP	BP	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN			<u> </u>	1
				00FE	3	1
				SP → 0100		← BP

				AX	6	
	MAIN PRO	С		CV I		
0000		MOV	AX,3	CX	1	
0002		PUSH	AX	•		
0004		CALL	FACTORIAL	SP	0100	
0006		MOV	AH,4CH	חח	0100	
8000		INT	21H	BP	0100	
	MAIN ENDF)				
	FACTORIAL	PROC NEAR]
000A		PUSH	ВР			1
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0		1
0012		MOV	AX,1	0010	0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	0054	1
0018		DEC	CX	UUF4	00FA	
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2]
001E		MUL	WORD PTR[BP+4]			1
0020	RETURN:	POP	ВР	00FA	0100	
0022		RET	2	00FC	0001:0006]
	END	MAIN				1
				00FE	3	
				SP → 0100		← BP

NAVINI DDOC		6 l	
MAIN PROC	CV		
0000 MOV AX,3	CX	1	
0002 PUSH AX			
0004 CALL FACTORIAL	SP	0100	
0006 MOV AH,4CH	DD	0100	
0008 INT 21H	BP	0100	
MAIN ENDP			
FACTORIAL PROC NEAR]
000A PUSH BP			-
000C MOV BP,SP			
000E CMP WORD PTR[BP+4],1	00EE	00F4	
0010 JG END_IF	00F0		_
0012 MOV AX,1	UUFU	0001:001E	
0014 JMP RETURN	00F2	1	
0016 END_IF: MOV CX, [BP+4]	00F4	0054	1
0018 DEC CX	0014	00FA	_
001A PUSH CX	00F6	0001:001E	
001C CALL FACTORIAL	00F8	2]
001E MUL WORD PTR[BP+4]		0400	-
0020 RETURN: POP BP	00FA	0100	
0022 RET 2	00FC	0001:0006	
END MAIN	00FE	3	1
	SP → 0100		← BP

				AX	6	
	MAIN PRO	С			6	
0000		MOV	AX,3	CX	1	
0002		PUSH	AX			
0004		CALL	FACTORIAL	SP	00FE	
0006		MOV	AH,4CH	חח	0100	
8000		INT	21H	BP	0100	
	MAIN END					_
	FACTORIAL	PROC NEAR				
000A		PUSH	ВР			
000C		MOV	BP,SP			
000E		CMP	WORD PTR[BP+4],1	00EE	00F4	
0010		JG	END_IF	00F0	0001.0015	
0012		MOV	AX,1		0001:001E	
0014		JMP	RETURN	00F2	1	
0016	END_IF:	MOV	CX, [BP+4]	00F4	00FA	
0018		DEC	CX			
001A		PUSH	CX	00F6	0001:001E	
001C		CALL	FACTORIAL	00F8	2	
001E		MUL	WORD PTR[BP+4]			
0020	RETURN:	POP	ВР	00FA	0100	
0022		RET	2	00FC	0001:0006	
	END	MAIN				
				$SP \longrightarrow 00FE$	0001:000A	
				0100		-