

Kamil Kałwelis 16943/IT/2020/NS

**ARCHITEKTURA
KOMPUTERÓW
LAB3**

Kamil Kałwelis 16943/IT/2020/NS

9.4.1 Sumowanie dwóch liczb – argumenty natychmiastowe

Registers			Flag		Load me at
A	06		S	0	<pre>1 jmp start 2 start: nop 3 mvi A, 02h 4 mvi B, 0x04h 5 add B 6 hlt</pre>
BC	04	00	Z	0	
DE	00	00	AC	0	
HL	30	07	P	1	
PSW	00	00	C	0	
PC	42	0A			
SP	FF	FF			
Int-Reg	00				

* wynik działania programu znajduje się w rejestrze A

9.4.2 Sumowanie dwóch liczb – argumenty w ustalonych komórkach

Code		Start
<pre>1 jmp start 2 start: NOP 3 LXI H, 3005h 4 MOV A, M 5 INX H 6 ADD M 7 INX H 8 MOV M, A 9 HLT</pre>		3005h
Address (Hex)	Address	Data
3005	12293	3
3006	12294	7
3007	12295	10
3008	12296	0
3009	12297	0

*wynik działania programu znajduje się w komórce 3007h

9.4.3 Sumowanie dwóch liczb - nazwy zmiennych symboliczne

1	jmp start	Address	Variable	Value	Value
2	arg1: db 2	4205	suma	06h	6
3	arg2: db 4	4203	arg1	02h	2
4	suma: db 0	4204	arg2	04h	4
5	start: nop				
6	MVI C,00H				
7	LXI H,arg1				
8	MOV A,M				
9	LXI H,arg2				
10	MOV B,M				
11	ADD B				
12	LXI H,suma				
13	MOV M,A				
14	hlt				

*wynik działania programu zapisany w zmiennej "suma"

9.4.4 Sumowanie elementów tablicy

Load me at		Data	Stack	Keypad	Memory
1	jmp start	Address	Variable	Value	Value (Decimal)
2	;code	4228	suma	5Bh	91
3	start: LXI H,tabcia	4220	tabcia	0Ah	10
4	MVI C,00H	4220	tabcia + 0 0Ah		10
5	MOV B,C	4221	tabcia + 1 0Bh		11
6	nxtbyte: MOV A,M	4222	tabcia + 2 0Ch		12
7	CPI 00h	4223	tabcia + 3 0Dh		13
8	JZ OUTPUT	4224	tabcia + 4 0Eh		14
9	ADD C	4225	tabcia + 5 0Fh		15
10	JNC SAVE	4226	tabcia + 6 10h		16
11	INR B	4227	tabcia + 7 00h		0
12	SAVE: MOV C,A				
13	INX H				
14	JMP nxtbyte				
15	OUTPUT: LXI H,suma				
16	MOV M,C				
17	INX H				
18	MOV M,B				
19	HLT				
20	; DANE				
21	tabcia: db 10,11,12,13,14,15,16,0				
22	suma: db 0, 0				
Line No		Assembler Message			
0		Program assembled successfully			

jmp start

;code

start: LXI H,tabcia

MVI C,00H

MOV B,C

nxtbyte: MOV A,M

CPI 00h

JZ OUTPUT

ADD C

JNC SAVE

INR B

```

SAVE: MOV C,A

INX H

JMP nxtbyte

OUTPUT: LXI H,suma

MOV M,C

INX H

MOV M,B

HLT





; DANE

tablica: db 10,11,12,13,14,15,16,0

suma: db 0, 0

```

Zad 9.5.6 Znajdowanie największej liczby w tablicy

<div>  Data  Stack  KeyPad  Memory </div>			
Address	Variable	Value	Value (Decimal)
420D	pozycja	05h	5
▶ 4203	tab	10h	16
420C	maks	12h	18
420B	ile	08h	8

;9.5.6

```
jmp start
```

```
;data
```

```
tab: db 16,10,2,3,18,5,6,0
```

```
ile: db 8
```

maks: db 0

pozycja: db 0

licznik: db 1

;code

start: nop

lxi h, tab

lda tab

sta maks

mov b, a

lda licznik

sta pozycja

back: inx h

lda licznik

inr a

sta licznik

mov a, m

cmp b

jc back

jz skip

jnc store

store: sta maks

mov b, a

lda licznik

sta pozycja

jnz back

skip: nop

hlt