

Johnny – Vector Program 3

Wiktor Zmiendak

1. Task:

Create a program in Johnny to handle vectors. It will be sorting values in vector.

2. Flowchart:

A – first swapping element

B – second swapping element

Vector_end – duplicated vector length

Vector_length – number of elements in vector



3. Solution:

000					
Adr	Hi	Lo	Asm	Opnd	^
000:	01	064	TAKE	064	
001:	03	060	SUB	060	
002:	04	060	SAVE	060	
003:	06	060	TST	060	
004:	05	006	JMP	006	
005:	10	000	HLT	000	
006:	01	064	TAKE	064	
007:	04	061	SAVE	061	
01 064					
000:	01	064	TAKE	064	
001:	03	060	SUB	060	
002:	04	060	SAVE	060	
003:	06	060	TST	060	
004:	05	006	JMP	006	

000					
Adr	Hi	Lo	Asm	Opnd	^
008:	08	061	DEC	061	
009:	06	061	TST	061	
010:	05	014	JMP	014	
011:	06	063	TST	063	
012:	05	035	JMP	035	
013:	10	000	HLT	000	
014:	01	065	TAKE	065	
015:	03	066	SUB	066	
01 064					
000:	01	064	TAKE	064	
001:	03	060	SUB	060	
002:	04	060	SAVE	060	
003:	06	060	TST	060	
004:	05	006	JMP	006	

000					
Adr	Hi	Lo	Asm	Opnd	^
016:	04	060	SAVE	060	
017:	06	060	TST	060	
018:	05	027	JMP	027	
019:	08	061	DEC	061	
020:	07	014	INC	014	
021:	07	015	INC	015	
022:	07	027	INC	027	
023:	07	029	INC	029	
01 064					
000:	01	064	TAKE	064	
001:	03	060	SUB	060	
002:	04	060	SAVE	060	
003:	06	060	TST	060	
004:	05	006	JMP	006	

000					
Adr	Hi	Lo	Asm	Opnd	^
024:	07	030	INC	030	
025:	07	032	INC	032	
026:	05	009	JMP	009	
027:	01	065	TAKE	065	
028:	04	060	SAVE	060	
029:	01	066	TAKE	066	
030:	04	065	SAVE	065	
031:	01	060	TAKE	060	
01 064					
000:	01	064	TAKE	064	
001:	03	060	SUB	060	
002:	04	060	SAVE	060	
003:	06	060	TST	060	
004:	05	006	JMP	006	

000				
Adr	Hi	Lo	Asm	Opnd
032:	04	066	SAVE	066
033:	07	063	INC	063
034:	05	019	JMP	019
035:	07	062	INC	062
036:	01	064	TAKE	064
037:	03	062	SUB	062
038:	04	061	SAVE	061
039:	09	063	NULL	063
000:	01	064	TAKE	064
001:	03	060	SUB	060
002:	04	060	SAVE	060
003:	06	060	TST	060
004:	05	006	JMP	006
01 064				

000				
Adr	Hi	Lo	Asm	Opnd
040:	01	014	TAKE	014
041:	03	061	SUB	061
042:	04	014	SAVE	014
043:	01	015	TAKE	015
044:	03	061	SUB	061
045:	04	015	SAVE	015
046:	01	027	TAKE	027
047:	03	061	SUB	061
000:	01	064	TAKE	064
001:	03	060	SUB	060
002:	04	060	SAVE	060
003:	06	060	TST	060
004:	05	006	JMP	006
01 064				

000				
Adr	Hi	Lo	Asm	Opnd
048:	04	027	SAVE	027
049:	01	029	TAKE	029
050:	03	061	SUB	061
051:	04	029	SAVE	029
052:	01	030	TAKE	030
053:	03	061	SUB	061
054:	04	030	SAVE	030
055:	01	032	TAKE	032
000:	01	064	TAKE	064
001:	03	060	SUB	060
002:	04	060	SAVE	060
003:	06	060	TST	060
004:	05	006	JMP	006
01 064				

000				
Adr	Hi	Lo	Asm	Opnd
054:	04	030	SAVE	030
055:	01	032	TAKE	032
056:	03	061	SUB	061
057:	04	032	SAVE	032
058:	08	061	DEC	061
059:	05	009	JMP	009
060:	00	001		
061:	00	000		
000:	01	064	TAKE	064
001:	03	060	SUB	060
002:	04	060	SAVE	060
003:	06	060	TST	060
004:	05	006	JMP	006
01 064				

The sorting works by continuously iterating through the vector and comparing each pair of adjacent elements A and B. When the algorithm encounters a situation where the first element is greater than the second it swaps their positions using a temporary variable. This cycle of comparisons and swaps is repeated until the entire vector is ordered correctly and no further swaps are required indicating that the vector is sorted.

