

Q1)

0-ADDRESS

PUSH A; PUSH A ONTO THE STACK

PUSH B; PUSH B ONTO THE STACK

PUSH C; PUSH C ONTO THE STACK

MUL; MULTIPLY B AND C

ADD; ADD A TO CURRENT RESULT

PUSH D; PUSH D ONTO THE STACK

PUSH E; PUSH E ONTO THE STACK

PUSH F; PUSH F ONTO THE STACK

MUL; MULTIPLY E AND F

SUB; SUBTRACT E AND F FROM D

DIV; DIVIDE THE RESULTS

POP X; RETURN X

1-ADDRESS

LOAD E; LOAD THE VALUE OF E

MULT F; MULTIPLY E AND F

STORE Y; STORE THE MULTIPLICATIVE IN Y

LOAD D; LOAD VALUE D

SUB Y; SUBTRACT COMPUTED Y BY D

STORE Y; STORE  $(D - E * F)$

LOAD B; LOAD VALUE B

MUL C; MULTIPLY B\*C

ADD A; ADD TO B\*C

DIV Y; DIVIDE A+B\*C/Y

STORE X; STORE COMPUTED VALUES IN X

2- ADDRESS

MOVE R1, E; MOVE THE VALUE OF E INTO FIRST REGISTER

MULT R1, F; MULTIPLY THE VALUE IN THE REGISTER BY F

MOVE R2, D; MOVE VALUE D IN ANOTHER REGISTER 2

SUB R2, R1; SUBTRACT R1 FROM R2 AND STORE IN R2

MOVE R1, B; MOVE B INTO REG 1

MUL R1, C; MULTIPLY VALUE B\*C AND STORE IN R1

ADD R1, A; ADD A TO R1

DIV R1, R2; DIVIDE R1 BY R2 AND STORE VALUE IN R1

MOVE X, R1; MOVE VALUES OF R1 TO X

3- ADDRESS

MULT R1, E, F; MULTIPLY E\*F AND STORE IN R1

SUB R1, D, R1; SUBTRACT R1 FROM D AND STORE IN R1

MULT R2, B, C; MULTIPLY B\*C AND STORE IN R2

ADD R2, A, R2; ADD A TO R2 AND STORE IN R2

DIV X, R1, R2; DIVIDE R1 AND R2 AND STORE IN X

Q2)

20 – 40

30 – 50

40 – 60

50 – 70

A) 20

B) 40

C) 60

D) 30

E) 50

F) 70

Q3)

A)

AS11M 01.05 Thu Apr 01, 2021 23:36 reversingArray.lst

```
0001 0000          ORG    $0000; start data at 0000
0002 0000 05      N      FCB    5;   array size
0003 0001 01 02 03 04 05  ARR    FCB    1,2,3,4,5;   array values
0004 0006 00      NEW    FCB;
0005 e000          ORG    $E000; continue with program at E000
0006 e000 ce 00 01      LDX    #ARR; load the address of the arrayS END
0007 e003 18 ce 00 06      LDY    #NEW;
0008 e007 96 00      LDAA   N;   load the size of the array
0009
0010 e009 bd e0 0d      JSR    LOAD; call the subroutine
0011 e00c 01          NOP;      this is where your program ends
0012                ;          (you may need to set a
0013                ;          breakpoint here)
```

```

0014          ;
0015 e00d e6 00      LOAD          LDAB  0,X;
0016 e00f 18 e7 00      STAB  0,Y;
0017 e012 18 08          INY;
0018 e014 08          INX;
0019 e015 4a          DECA;
0020 e016 26 f5          BNE  LOAD;
0021 e018 18 ce 00 0a      LDY  #NEW+4;
0022 e01c ce 00 01      LDX  #ARR;
0023 e01f 96 00          LDAA  N;
0024 e021 bd e0 24          JSR  REVERSE1;
0025
0026 e024 18 e6 00      REVERSE1 LDAB  0,Y;
0027 e027 e7 00          STAB  0,X;
0028 e029 08          INX;
0029 e02a 18 09          DEY;
0030 e02c 4a          DECA;
0031 e02d 26 f5          BNE  REVERSE1;
0032 e02f 39          RTS;
0033

```

Number of errors 0

Number of warnings 0

B)

Me8HC11 CPU

IRQ

Port A

XIRQ

Port B

Browse Mem.

Reset

Port C

Memory Watch

Registers

Port D

View Code

Display

Port E

Break Point

Keypad

Pin Scope

MCU

Name	Value
ACCA	\$0
ACCB	\$5
ACCD	\$5
IX	\$1
IY	\$A
SP	\$FD
PC	\$E01F
CCR	01010000
	SXHNZVC

Modify

Update

Close

Memory

address	00	01	02	03	04	05	06	07	address	08	09	0A	0B	0C	0D	0E	0F	address
0000	05	01	02	03	04	05	01	02	0000	03	04	05	5C	66	63	73	5C	0000
0010	73	6F	66	74	77	61	72	65	0010	5C	4A	61	76	61	5C	6A	64	0010
0020	68	00	4C	4F	43	41	4C	41	0020	50	50	44	41	54	41	3D	43	0020
0030	3A	5C	55	73	65	72	73	5C	0030	77	69	6E	31	30	2D	6C	61	0030
0040	62	5C	41	70	70	44	61	74	0040	61	5C	4C	6F	63	61	6C	00	0040
0050	4C	4F	47	4F	4E	53	45	52	0050	56	45	52	3D	5C	5C	57	31	0050
0060	30	56	49	52	54	4C	41	42	0060	00	4D	49	4E	49	54	41	42	0060
0070	5F	4C	49	43	45	4E	53	45	0070	5F	46	49	4C	45	3D	43	3A	0070
0080	5C	50	72	6F	67	72	61	6D	0080	44	61	74	61	5C	4D	69	6E	0080
0090	69	74	61	62	5C	4D	69	6E	0090	69	74	61	62	20	31	39	00	0090
00A0	4E	55	4D	42	45	52	5F	4F	00A0	46	5F	50	52	4F	43	45	53	00A0
00B0	53	4F	52	53	3D	32	00	4F	00B0	53	3D	57	69	6E	64	6F	77	00B0
00C0	73	5F	4E	54	00	50	61	74	00C0	68	3D	43	3A	5C	6F	70	74	00C0
00D0	5C	66	63	73	5C	73	6F	66	00D0	74	77	61	72	65	5C	50	79	00D0
00E0	74	68	6F	6E	33	37	5C	53	00E0	63	72	69	70	74	73	5C	3B	00E0
00F0	43	3A	5C	6F	70	74	5C	66	00F0	63	73	5C	73	6F	66	E0	0C	00F0
0100	61	72	65	5C	50	79	74	68	0100	6F	6E	33	37	5C	3B	43	3A	0100
0110	5C	50	72	6F	67	72	61	6D	0110	20	46	69	6C	65	73	20	28	0110
0120	78	38	36	29	5C	43	6F	6D	0120	6D	6F	6E	20	46	69	6C	65	0120
0130	73	5C	4F	72	61	63	6C	65	0130	5C	4A	61	76	61	5C	6A	61	0130
0140	76	61	70	61	74	68	38	43	0140	3A	5C	57	69	6E	64	6F	77	0140
0150	73	5C	73	79	73	74	65	6D	0150	33	32	38	43	3A	5C	57	69	0150
0160	6E	64	6F	77	73	38	43	3A	0160	5C	57	69	6E	64	6F	77	73	0160
0170	5C	53	79	73	74	65	6D	33	0170	32	5C	57	62	65	6D	3B	43	0170
0180	3A	5C	57	69	6E	64	6F	77	0180	73	5C	53	79	73	74	65	6D	0180
0190	33	32	5C	57	69	6E	64	6F	0190	77	73	50	6F	77	65	72	53	0190
01A0	68	65	6C	6C	5C	76	31	2E	01A0	30	5C	3B	43	3A	5C	57	69	01A0
01B0	6E	64	6F	77	73	5C	53	73	01B0	73	74	65	6D	33	32	5C	4F	01B0

Update

Close

C)

00FE – E0

00FF – 0C

D)

Memory

address	00	01	02	03	04	05	06	07	address	08	09	0A	0B	0C	0D	0E	0F	address
0000	05	05	04	03	02	01	01	02	0000	03	04	05	5C	66	63	73	5C	0000
0010	73	6F	66	74	77	61	72	65	0010	5C	4A	61	76	61	5C	6A	64	0010
0020	68	00	4C	4F	43	41	4C	41	0020	50	50	44	41	54	41	3D	43	0020
0030	3A	5C	55	73	65	72	73	5C	0030	77	69	6E	31	30	2D	6C	61	0030
0040	62	5C	41	70	70	44	61	74	0040	61	5C	4C	6F	63	61	6C	00	0040
0050	4C	4F	47	4F	4E	53	45	52	0050	56	45	52	3D	5C	5C	57	31	0050
0060	30	56	49	52	54	4C	41	42	0060	00	4D	49	4E	49	54	41	42	0060
0070	5F	4C	49	43	45	4E	53	45	0070	5F	46	49	4C	45	3D	43	3A	0070
0080	5C	50	72	6F	67	72	61	6D	0080	44	61	74	61	5C	4D	69	6E	0080
0090	69	74	61	62	5C	4D	69	6E	0090	69	74	61	62	20	31	39	00	0090
00A0	4E	55	4D	42	45	52	5F	4F	00A0	46	5F	50	52	4F	43	45	53	00A0
00B0	53	4F	52	53	3D	32	00	4F	00B0	53	3D	57	69	6E	64	6F	77	00B0
00C0	73	5F	4E	54	00	50	61	74	00C0	68	3D	43	3A	5C	6F	70	74	00C0
00D0	5C	66	63	73	5C	73	6F	66	00D0	74	77	61	72	65	5C	50	79	00D0
00E0	74	68	6F	6E	33	37	5C	53	00E0	63	72	69	70	74	73	5C	3B	00E0
00F0	43	3A	5C	6F	70	74	5C	66	00F0	63	73	5C	73	E0	24	E0	0C	00F0
0100	61	72	65	5C	50	79	74	68	0100	6F	6E	33	37	5C	3B	43	3A	0100
0110	5C	50	72	6F	67	72	61	6D	0110	20	46	69	6C	65	73	20	28	0110
0120	78	38	36	29	5C	43	6F	6D	0120	6D	6F	6E	20	46	69	6C	65	0120
0130	73	5C	4F	72	61	63	6C	65	0130	5C	4A	61	76	61	5C	6A	61	0130
0140	76	61	70	61	74	68	38	43	0140	3A	5C	57	69	6E	64	6F	77	0140
0150	73	5C	73	79	73	74	65	6D	0150	33	32	38	43	3A	5C	57	69	0150
0160	6E	64	6F	77	73	38	43	3A	0160	5C	57	69	6E	64	6F	77	73	0160
0170	5C	53	79	73	74	65	6D	33	0170	32	5C	57	62	65	6D	3B	43	0170
0180	3A	5C	57	69	6E	64	6F	77	0180	73	5C	53	79	73	74	65	6D	0180

Update

Close

E)

ARR	NEW	ARR	NEW	RETURN
-----	-----	-----	-----	--------

5	5	1	5	
4	4	2	4	
3	3	3	3	
2	2	4	2	
1	1	5	1	
*	*	*	*	*