Ethan Crawford

CSS 422

February 10, 2009

Homework Problem Set #4

From <https://faculty.washington.edu/aberger/CSS422W09/Homework/Homework%204/CSS422W09HW4.htm>

Problem #1

Source file:

\*----------------------------------------------------------------------------

\* Program : Homework 4, problem 1

\* Written by : Ethan Crawford

\* Course : CSS 422

\* Date : February 10, 2009

\* Description: write a program that fills all of memory between $0000A000 to

\* $0000A2FF, inclusive, with the longword pattern $5555AAAA.

\*----------------------------------------------------------------------------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\* EQUates section

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

startAddr EQU $0000A000

endAddr EQU $0000A2FF

pattern EQU $5555AAAA

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\* Beginning of code segment.

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

OPT CRE

START ORG $400

\* Set up registers

MOVE.L #pattern,D0 \* Move fill value to D0

MOVEA.L #startAddr,A0 \* Move starting addr to A0

MOVEA.L #endAddr,A1 \* Move ending addr to A1

LOOP MOVE.L D0,(A0)+ \* Write fill value and increment startAddr

CMPA.L A1,A0 \* Subtract endAddr from startAddr

BLE LOOP \* Do it again while startAddr <= endAddr

MOVE.B #9,D0

TRAP #15 Halt Simulator

END START

List file:

00000400 Starting Address

Assembler used: EASy68K Editor/Assembler v4.1.1

Created On: 2/6/2009 3:23:51 PM

00000000 1 \*------------------------------------------------------------------------------

00000000 2 \* Program : Homework 4, problem 1

00000000 3 \* Written by : Ethan Crawford

00000000 4 \* Course : CSS 422

00000000 5 \* Date : February 10, 2009

00000000 6 \* Description: write a program that fills all of memory between $0000A000 to

00000000 7 \* $0000A2FF, inclusive, with the longword pattern $5555AAAA.

00000000 8 \*------------------------------------------------------------------------------

00000000 9

00000000 10 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 11 \*

00000000 12 \* EQUates section

00000000 13 \*

00000000 14 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 =0000A000 15 startAddr EQU $0000A000

00000000 =0000A2FF 16 endAddr EQU $0000A2FF

00000000 =5555AAAA 17 pattern EQU $5555AAAA

00000000 18

00000000 19 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 20 \*

00000000 21 \* Beginning of code segment.

00000000 22 \*

00000000 23 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 24 OPT CRE

00000400 25 START ORG $400

00000400 26

00000400 27 \* Set up registers

00000400 203C 5555AAAA 28 MOVE.L #pattern,D0 \* Move fill value to D0

00000406 207C 0000A000 29 MOVEA.L #startAddr,A0 \* Move starting addr to A0

0000040C 227C 0000A2FF 30 MOVEA.L #endAddr,A1 \* Move ending addr to A1

00000412 31

00000412 20C0 32 LOOP MOVE.L D0,(A0)+ \* Write fill value and increment startAddr

00000414 B1C9 33 CMPA.L A1,A0 \* Subtract endAddr from startAddr

00000416 6FFA 34 BLE LOOP \* Do it again while startAddr <= endAddr

00000418 35

00000418 103C 0009 36 MOVE.B #9,D0

0000041C 4E4F 37 TRAP #15 Halt Simulator

0000041E 38

0000041E 39 END START

No errors detected

No warnings generated

SYMBOL TABLE INFORMATION

Symbol-name Value

-------------------------

ENDADDR A2FF

LOOP 412

PATTERN 5555AAAA

START 400

STARTADDR A000



Problem #2

Source file:

\*----------------------------------------------------------------------------

\* Program : Homework 4, problem 2

\* Written by : Ethan Crawford

\* Course : CSS 422

\* Date : February 10, 2009

\* Description: Reads the memory between the addresses $000A0000 and $000C0000,

\* searching for the 16-bit word $ABC5. If found, it stores the

\* address in a memory location located just after the program code and

\* terminates. If it fails to find the pattern with the specified memory range,

\* it writes $FFFF to the memory location and terminates.

\*----------------------------------------------------------------------------

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\* EQUates section

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

startAddr EQU $000A0000

endAddr EQU $000C0000

pattern EQU $ABC5

wFail EQU $FFFF

lRet DS.L 1 \* Allocate one long for return address

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*

\* Beginning of code segment.

\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

OPT CRE

START ORG $400

\* Set up registers and reserve memory for found word

MOVE.L #pattern,D0 \* Move search value to D0

MOVEA.L #startAddr,A0 \* Move starting addr to A0

MOVEA.L #endAddr,A1 \* Move ending addr to A1

CLR.L lRet \* Clear return value

LOOP MOVE.W (A0)+,D1 \* load the word and increment startAddr

CMP.W D0,D1 \* compare it to pattern

BNE ADDR \* If pattern does not match, check addresses

MOVE.L A0,lRet \* else, store the word's address in a local

BRA EXIT \* and exit

ADDR CMPA.L A1,A0 \* If startAddr <= endAddr,

BLE LOOP \* loop to the beginning.

MOVE.L #wFail,lRet \* else, store 0xFFFF in a local and exit

EXIT MOVE.B #9,D0

TRAP #15 Halt Simulator

END START

List file:

00000400 Starting Address

Assembler used: EASy68K Editor/Assembler v4.1.1

Created On: 2/6/2009 3:16:36 PM

00000000 1 \*------------------------------------------------------------------------------

00000000 2 \* Program : Homework 4, problem 2

00000000 3 \* Written by : Ethan Crawford

00000000 4 \* Course : CSS 422

00000000 5 \* Date : February 10, 2009

00000000 6 \* Description: Reads the memory between the addresses $000A0000 and $000C0000,

00000000 7 \* searching for the 16-bit word $ABC5. If found, it stores the

00000000 8 \* address in a memory location located just after the program code and

00000000 9 \* terminates. If it fails to find the pattern with the specified memory range,

00000000 10 \* it writes $FFFF to the memory location and terminates.

00000000 11 \*------------------------------------------------------------------------------

00000000 12

00000000 13 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 14 \*

00000000 15 \* EQUates section

00000000 16 \*

00000000 17 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000000 =000A0000 18 startAddr EQU $000A0000

00000000 =000C0000 19 endAddr EQU $000C0000

00000000 =0000ABC5 20 pattern EQU $ABC5

00000000 =0000FFFF 21 wFail EQU $FFFF

00000000 22 lRet DS.L 1 \* Allocate one long for return address

00000004 23

00000004 24 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000004 25 \*

00000004 26 \* Beginning of code segment.

00000004 27 \*

00000004 28 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

00000004 29 OPT CRE

00000004 30

00000400 31 START ORG $400

00000400 32

00000400 33 \* Set up registers and reserve memory for found word

00000400 203C 0000ABC5 34 MOVE.L #pattern,D0 \* Move search value to D0

00000406 207C 000A0000 35 MOVEA.L #startAddr,A0 \* Move starting addr to A0

0000040C 227C 000C0000 36 MOVEA.L #endAddr,A1 \* Move ending addr to A1

00000412 42B8 0000 37 CLR.L lRet \* Clear return value

00000416 38

00000416 3218 39 LOOP MOVE.W (A0)+,D1 \* load the word at startAddr and post-increment startAddr

00000418 B240 40 CMP.W D0,D1 \* compare it to pattern

0000041A 6600 000A 41 BNE ADDR \* If pattern does not match, check addresses

0000041E 21C8 0000 42 MOVE.L A0,lRet \* else, store the word's address in a local

00000422 6000 000E 43 BRA EXIT \* and exit

00000426 44

00000426 B1C9 45 ADDR CMPA.L A1,A0 \* If startAddr <= endAddr,

00000428 6FEC 46 BLE LOOP \* loop to the beginning.

0000042A 21FC 0000FFFF 0000 47 MOVE.L #wFail,lRet \* else, store 0xFFFF in a local and exit

00000432 48

00000432 103C 0009 49 EXIT MOVE.B #9,D0

00000436 4E4F 50 TRAP #15 Halt Simulator

00000438 51 END START

No errors detected

No warnings generated

SYMBOL TABLE INFORMATION

Symbol-name Value

-------------------------

ADDR 426

ENDADDR C0000

EXIT 432

LOOP 416

LRET 0

PATTERN ABC5

START 400

STARTADDR A0000

WFAIL FFFF

Flow chart:

