# Exercise 4 Problems

Do NOT use a calculator on this assignment. You will not be allowed to use one on the test or quizzes.

1. Convert the following assembly code fragments into machine code using the most efficient addressing mode available (i.e. if possible, use direct instead of extended).
   1. MOVB #45,$1500
   2. MOVB $2000,10,X
   3. MOVW 5,X,-10,Y
   4. TFR A,Y
   5. TFR D,SP
   6. TFR X,A
   7. EXG A,B

* 1. EXG X,B
  2. EXG D,SP
  3. EXG Y,A

1. Convert the program below to a listing file. Include a column for addresses, a contents column, and a symbol table at the end.

|  |  |  |
| --- | --- | --- |
| lowbits | equ | %00000111 |
|  |  |  |
|  | org | 3000h |
| data | dc.b | 05h,1Fh,02h,26h |
|  |  |  |
|  | org | 3100h |
| digit | dc.b | 7h |
| length | dc.b | 3h |
| count | ds.b | 1 |
| array | dc.w | 3000h |
|  |  |  |
|  | org | C000h |
|  | ldx | array |
|  | clr | count |
|  | ldaa | length |
| loop | beq | finish |
|  | ldab | 0,x |
|  | andb | #lowbits |
|  | cmpb | digit |
|  | bne | skip |
|  | inc | count |
| skip | inx |  |
|  | deca |  |
|  | bra | loop |
| finish | swi |  |

1. Convert the program below to a listing file. Include a column for addresses, a contents column, and a symbol table at the end.

|  |  |  |
| --- | --- | --- |
| limit | equ | 70 |
|  |  |  |
|  | org | 2000 |
| length | ds.b | 2 |
| grades | dc.w | $3200 |
| count | ds.b | 1 |
|  |  |  |
|  | org | C000h |
|  | ldx | grades |
|  | clr | count |
|  | ldab | #limit |
|  | ldy | length |
| again | beq | done |
|  | ldaa | 0,x |
|  | cba |  |
|  | bhs | skip |
|  | inc | count |
| skip | inx |  |
|  | dey |  |
|  | bra | again |
| done | swi |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. Convert the program below to a listing file. Include a column for addresses, a contents column, and a symbol table at the end.

org 1000h

string dc.w 3000h

array ds.w 1

arrlen dc.w 0200

endarr ds.w 1

found ds.w 1

org C000h

ldd array

addd arrlen

subd #2

std endarr

ldx string

ldy array

loop cpy endarr

beq done

ldaa 0,y

cmpa 0,x

bne next

ldaa 1,y

cmpa 1,x

bne next

ldaa 2,y

cmpa 2,x

bne next

sty found

ldy endarr

bra loop

next iny

bra loop

done swi