Problem 7.1:

The radius of a circle (in cm) is stored as an 8-bit unsigned integer in a location labeled “radius”. Write a program starting at location $4000 to calculate the area of the circle (in cm^2) and store the result as a 16-bit unsigned integer in a location labeled “area”. Round the result to the nearest integer value. You can use 3.142 as an approximation to π.

Problem 7.2:

Write a program starting at $3800 to copy the contents of memory location $1000 to location $1001 in bit reversed order. In other words, bit0 of $1000 is copied to bit7 of $1001, bit1 of $1000 is copied to bit6 of $1002, etc. At the end of the operation, memory location $1000 should be unchanged from its initial value.

Solution 7.1:

Square radius and multiply by 3142/1000. Round by adding 500 to dividend before dividing by 1000.

```
ORG     $4000
ldaa    radius          ; load A with radius
tfr     A, B            ; copy to B
mul                     ; D = radius^2
ldy     #3142
emul                    ; Y:D = 3142 * (radius^2)
addd    #500            ; allow for rounding
exg     Y, D
adcb    #0              ; propagate carry into MSbytes
adca    #0
exg     Y, D
ldx     #1000
ediv                    ; Y = (3142*(radius^2))/1000
sty     area
```

Solution 7.2:

Load source data in A. Then use a loop and the rotate instruction to move bits from A into destination memory location in reverse order.

```
        ORG     $3800
        ldy     #8              ; loop counter
        ldaa    $1000           ; source data in acc. A
loop:   lsra                    ; carry = LSB of A (with right shift of A)
        rol     $1002           ; move carry into LSB of $1002 (with left shift of mem loc)
        dbnz    Y, loop         ; done yet?
        bgnd
```