```
main:
.data #static data section
array:      .word 7 0 0 0 0 0 6
oddarr:         .word 0 0 0 0 0 0 0
evenarr:        .word 0 0 0 0 0 0 0
zeros:          .word 0 0 0 0 0 0 0
size:           .word 7
numodd:         .asciiz "positive odd numbers: "
numeven:        .asciiz "positive even numbers: "
zeromsg:        .asciiz "Zeros: "
.text

#load initial values
la x17, array
la x18, oddarr
la x19, evenarr
la x21, zeros
lw x14, size

addi x26, x26, 2
addi x28, x28, 1

#store in the odd, even and zero arrays
checkandsort:
bge x16, x14, EXIT
addi x16, x16, 1
lw x27, 0(x17)
rem x25, x27, x26
addi x17, x17, 4
bge x25, x28, sortOdd
div x29, x27, x26
blt x29, x28, sortZero
sw x27, 0(x19)
addi x19, x19, 4
addi x5, x5, 1
jal x0, checkandsort

#store in the zero array
sortZero:
sw x27, 0(x21)
addi x21, x21, 4
addi x6, x6, 1
jal x0, checkandsort

#store in the odd array
sortOdd:
sw x27, 0(x18)
addi x18, x18, 4
addi x8, x8, 1
jal x0, checkandsort

EXIT:
addi x27, x0, 0
addi x28, x0, 0
addi x10, x0, 1
la x18, oddarr
la x19, evenarr
la x21, zeros

#print odd text message
printOdd:
addi x10 x0 1
addi x10 x0 4
la x11 numodd
ecall

#loop thru odd array to print
printOddstart:
bge x24, x8, loop1

addi x10, x0, 1
lw x11, 0(x18)
addi x18, x18, 4
ecall
addi x11, x0, 32
addi x10, x0, 11
ecall
addi x24, x24, 1
jal x0, printOddstart

loop1:
addi x11, x0, 10
addi x10, x0, 11
ecall
addi x27, x0, 0

#print odd text message
printEven:
addi x10 x0 4
la x11 numeven
ecall

#loop thru even array to print
printEvenstart:
bge x27, x5, loop2

addi x10, x0, 1
lw x11, 0(x19)
addi x19, x19, 4
ecall
addi x11, x0, 32
addi x10, x0, 11
ecall
addi x27, x27, 1
jal x0, printEvenstart

loop2:
addi x11, x0, 10
addi x10, x0, 11
ecall

#print zero text message
printZeros:
addi x10 x0 4
la x11 zeromsg
ecall

#loop thru zero array to print
printZerosstart:
bge x28, x6, DONE

addi x10, x0, 1
lw x11, 0(x21)
addi x21, x21, 4
ecall
addi x11, x0, 32
addi x10, x0, 11
ecall
addi x28, x28, 1
jal x0, printZerosstart

DONE:
```