

Homework 4 – due Wedn. 25 March

(1) Write a Pep8 assembly program that takes as input n, m (two integers in the range $1 \leq \dots < 2^{15}$), and that computes and outputs $n//m$ (quotient) and $n\%m$ (remainder).

Explain how your program works (use a flow-chart).

Test your program using the Pep8 simulator; save your program in a *file* while you debug. (The Pep8 simulator lets you open a Pep8 program file.)

(2) Write a Pep8 assembly program that takes as input n, m (two integers in the range $-2^{15} \leq \dots < 2^{15}$), and that computes and outputs $n \times m$ (product) or ‘‘overflow’’ (if $n \times m$ is not in the range $-2^{15} \leq \dots < 2^{15}$). Negative numbers are allowed.

Explain how your program works (use a flow-chart).

Test your program using the Pep8 simulator; save your program in a file while you debug.

(3) Write a Pep8 assembly program that outputs the *maximum* element in an array of k integers. Each integer is in the range $-2^{15} \leq \dots < 2^{15}$. The array starts at address `arr`. The input to the problem consists of k and `arr`. We assume that k and `arr` are non-negative, and that the array is already in memory.

1. How large can `arr + k` be so that the array fits into user memory? (Remember that an integer uses *two* bytes, whereas memory is byte-addressed.)

2. Write the program, assuming the array is already stored in memory. Use *indexed addressing mode*.

Explain how your program works (use a flow-chart).

Test your program using the Pep8 simulator; save your program in a file while you debug.