**Introduction.**

The SIMMAC is a simulated virtual machine that will run programs in its own programming dialect. This design will use Java to create and execute the SIMMAC. The memory of this virtual machine will have a 512 piece expression, registers, an arithmetic logical unit, operands, and opcodes. The operand will perform a function, while the opcode will hold values or addresses. The purpose of this exercise is to handle multiple processes. Processes will be feed to the SIMMAC via its programming dialect in a text file. These processes validate and execute in a round robin fashion. IF everything checks out the contents of all the memory and registers are printed to screen.

**Design and Usage.**

For the memory distribution of the SIMMAC, guidelines our put in a cluster. Memory is made for every hub that is the exact length of the hub. The Process Control Block executes processes in round robin fashion by storing the processes in an array list data structure.