Assignment 2 (Use of family of exec system call)

Write a collection of programs p1, p2, p3 such that they execute sequentially with the same process-id, and each program should also print its PID. *(process id)* The user should be able to invoke any combination of these programs, to achieve the required functionality.

For example consider three programs *twice , half, square* which accept only one integer as argument and does some specific operation.

$twice 10 prints **20** and some int which is its process-id as output
$square 10 prints **100** and some int which is its process-id as output
$half 10 prints **5** and some int which is its process-id as output

Now the user should be able to combine these programs in any combination to achieve the required result.

For example:
$twice square half twice half 10
should calculate *half(twice(half(square(twice(10)))))* and print **200** as result. It should also print the process ids of each program as it executes. **Note that the process-id printed by each of these programs should be the same, in this case.**

$square twice 2
should calculate *twice(square(2))* and print **8** as result, and the process id of square and twice, which should be the same.
The evaluation order is from left to right

Note that the last argument is the integer, and the remaining arguments are the programs to be invoked.

This should be generally applicable to any *n* number of processes, all of which are written by you. *Minimum of three should be written*

pl p2 p3 ... pn arg_value
Instructions

- Remeber that any number of programs should executed within one pid. The final result may be printed by the last process to execute.