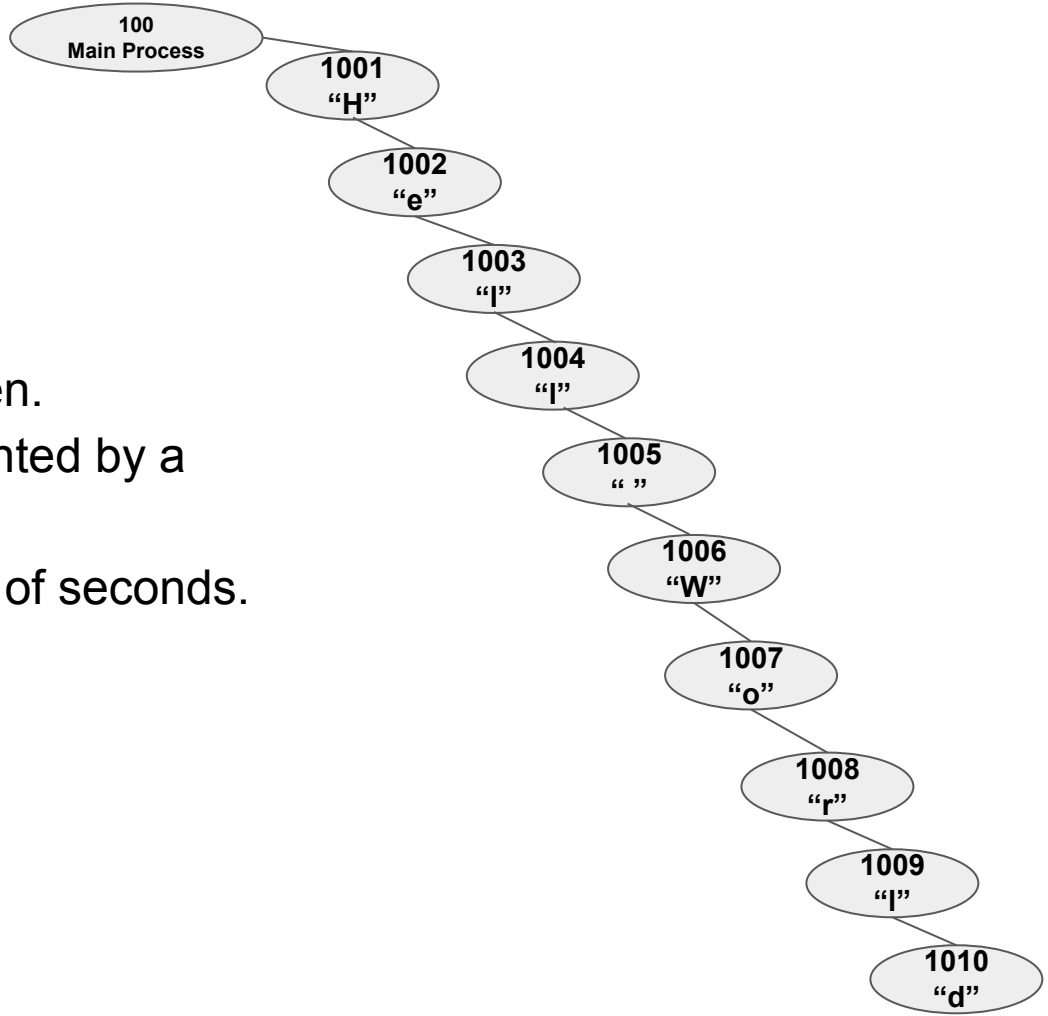
Lab-2

Part-I

- First program on Minix.
- clang compiler
- Print “Hello World” on screen.
- Each character must be printed by a different process.
- sleep for a random number of seconds.



Part-II

- Write a collection of processes twice, square and half which execute with same PID

  $./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.

  $./twice ./square ./half ./twice ./half 10

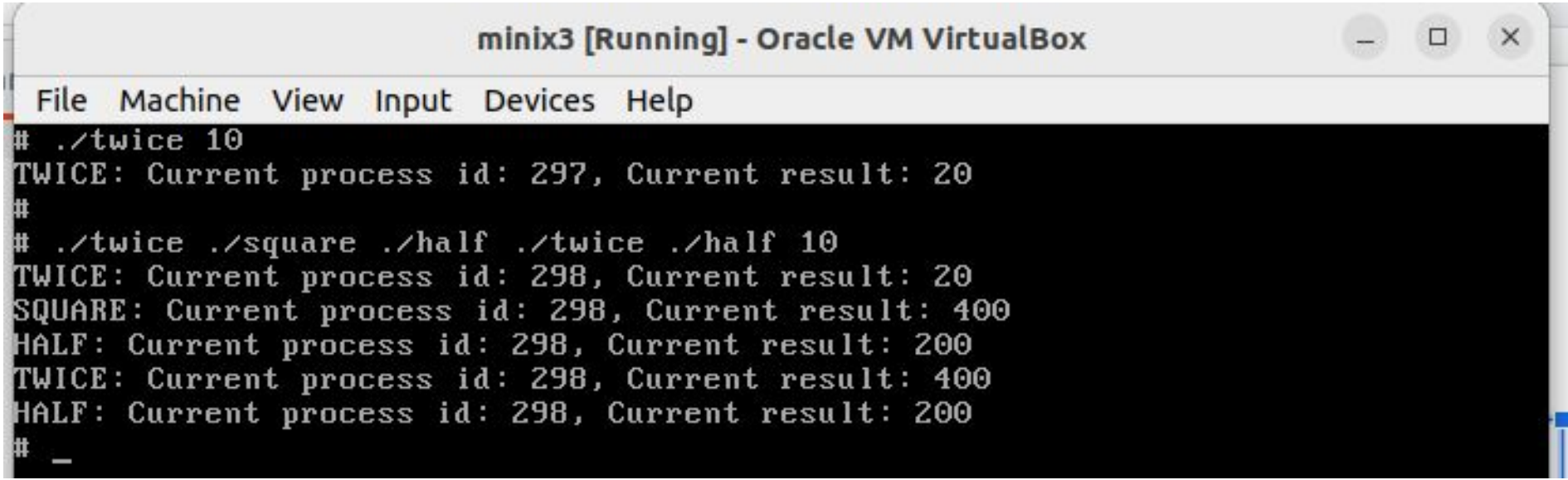
  should calculate *half(twice(half(square(twice(10)))))* and print **200** as result. It should also print the process ids of each program as it executes.(**process-id printed by each of these programs should be the same, in this case)**

- *pl p2 p3 ... pn arg_value*

Hint:

- Use execvp family of system calls.

Sample output



Minix3 Important Folders

Libraries	Source
Kernel	*kernel/*
Process Manager server	*servers/pm*
Virtual File System server	*servers/vfs*
Minix File System service	*servers/mfs*
Virtual Memory manager	*servers/vm*

Part-III

Modify minix source code:

- Whenever a process created-> Print “Minix: PID <pid> created”
- Whenever a process ends-> Print “Minix: PID <pid> exited”

Hint: look at minix/servers/pm

Questions?