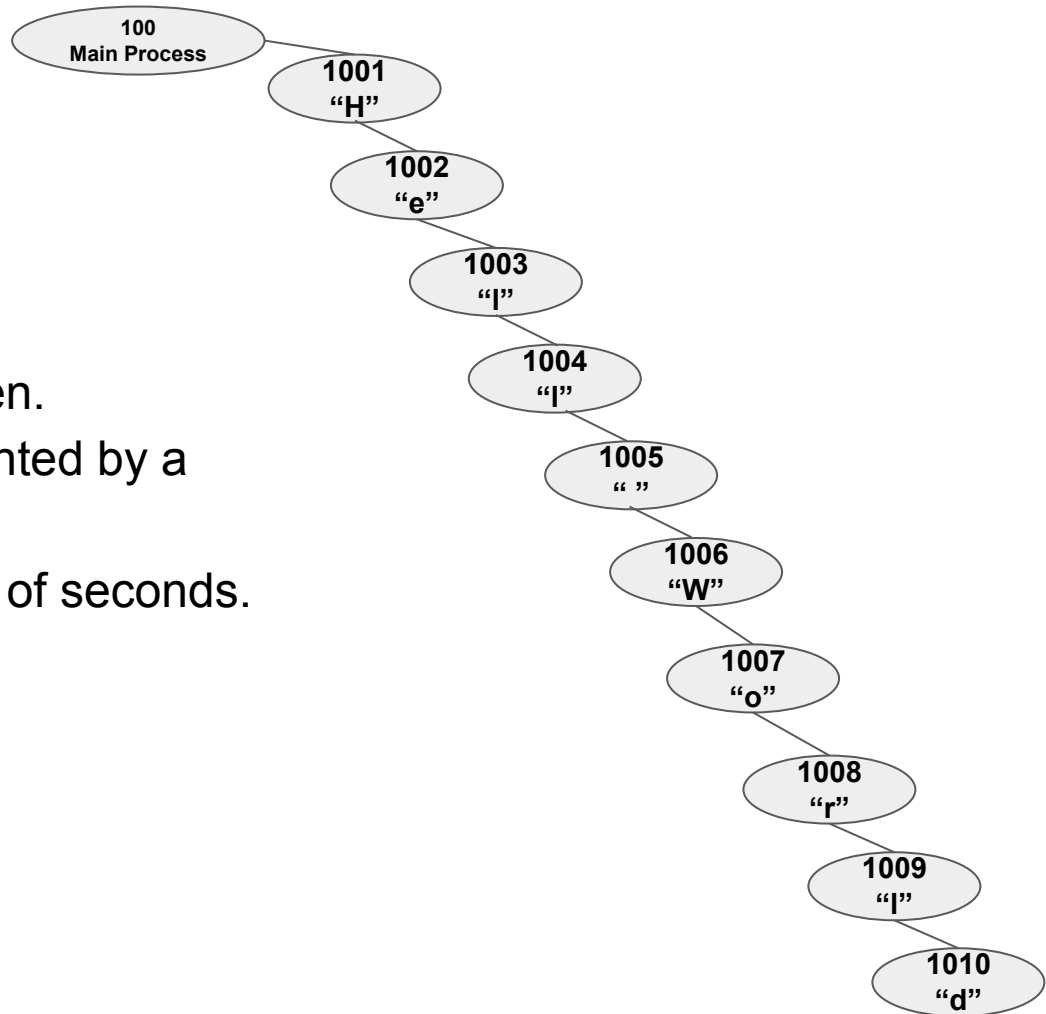


Lab-2

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`

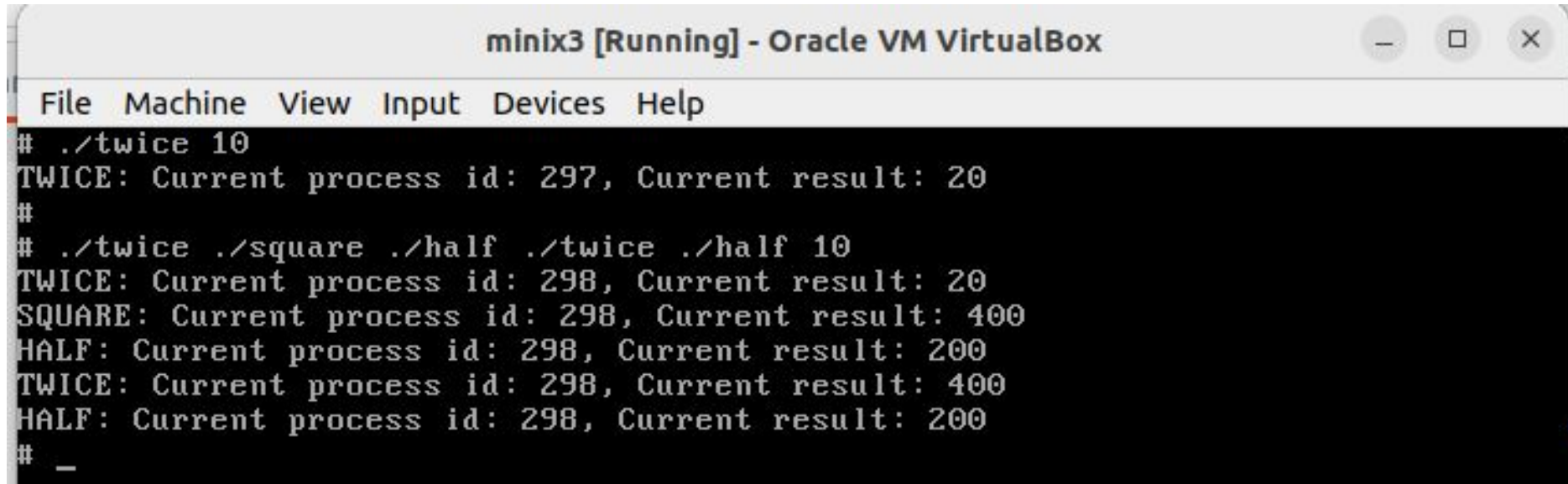
should calculate $half(twice(half(square(twice(10))))))$ and print **200** as result. It should also print the process ids of each program as it executes. **(process-id printed by each of these programs should be the same, in this case)**

- *`pl p2 p3 ... pn arg_value`*

Hint:

- Use `execvp` family of system calls.

Sample output

A screenshot of a virtual machine window titled "minix3 [Running] - Oracle VM VirtualBox". The window has a menu bar with "File", "Machine", "View", "Input", "Devices", and "Help". The main area is a black terminal with white text. The text shows a series of commands and their outputs. The first command is `./twice 10`, which outputs `TWICE: Current process id: 297, Current result: 20`. The second command is `./twice ./square ./half ./twice ./half 10`, which outputs four lines: `TWICE: Current process id: 298, Current result: 20`, `SQUARE: Current process id: 298, Current result: 400`, `HALF: Current process id: 298, Current result: 200`, and `TWICE: Current process id: 298, Current result: 400`. The third command is `./half 10`, which outputs `HALF: Current process id: 298, Current result: 200`. The prompt `#` is visible at the end of each line.

```
minix3 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
# ./twice 10
TWICE: Current process id: 297, Current result: 20
#
# ./twice ./square ./half ./twice ./half 10
TWICE: Current process id: 298, Current result: 20
SQUARE: Current process id: 298, Current result: 400
HALF: Current process id: 298, Current result: 200
TWICE: Current process id: 298, Current result: 400
HALF: Current process id: 298, Current result: 200
# _
```

Minix3 Important Folders

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

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?