

Distributed Operating Systems

Project 1

Work Unit

Each machine connected was spawned with **100** processes. Each process was given an initial workload of **one million** coins. Upon completion of this the process requests the server for additional work. This way the process is never idle.

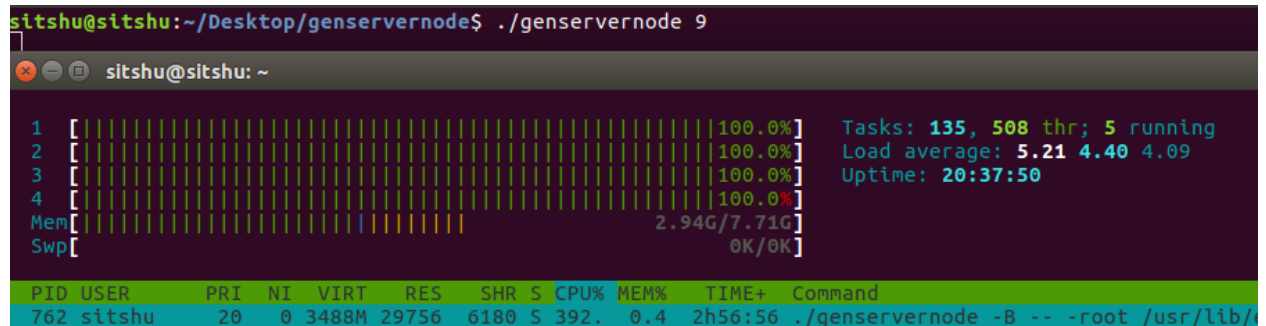
This was implemented to avoid collisions of mining the same coin by different processes. As the server has a record of range of all coins mined at all time.

Running `./project1 4`

```
➤ ./bitcoin 4
Server started as : master@10.136.193.159
misrani;iVu 0000e21208c4476bb69748cee27e9c42de877ca12539c82bc04e051da4a4f36c
misrani;4clA 000050b4f388f61047d6be88ff847be1e9f54b31da555c8e0b2e6912197d4070
misrani;7THn 00008f6eb55f659a5b34d9fefc79812f8baa7eb3846fa64f5bf805e5b2e3029b
misrani;8IjZ 00009cce482f25e3cc8d956c7b6e5b3481cb762c2b37b1a8323b4bf687e2bd39
misrani;sTrw 0000b2a7373e1bb6f48d098f2d64fc80af94ed5d08587aada8c32bb4d42813f6
misrani;nlH2 00002a29964c0f747f7cb840c1c04f2cc52ab805af3b1ef20ac76bed65a1be2b
misrani;1EWb 0000832f5f7653e127ed55005f827db380cbbbed13d4220afc71b848286fdb47e
misrani;fNPi 00008271bd50fc512849f59c2748dc83c7af89e4b2e3ffb5805a54680c5caec1
misrani;2DeX 00008c2e20e63e8ab6d26dc771de11700444aad4f37087b5c4fa57c4e061b527
```

Running time

```
sitshu@sitshu:~/Desktop/genservernode$ ./genservernode 9
```



The screenshot shows a terminal window with a dark background. At the top, the command `./genservernode 9` has been executed. Below the command, there is a window title bar with the text `sitshu@sitshu: ~`. The main content of the terminal displays system statistics: four lines of CPU usage for processors 1, 2, 3, and 4, each showing 100.0% usage with a bar chart; a line for memory usage showing 2.94G/7.71G; and a line for swap usage showing 0K/0K. To the right of these statistics, the following text is displayed: `Tasks: 135, 508 thr; 5 running`, `Load average: 5.21 4.40 4.09`, and `Uptime: 20:37:50`. At the bottom of the terminal, a table of running processes is shown, with the first line being the header: `PID USER PRI NI VIRT RES SHR S CPU% MEM% TIME+ Command`. The first data row shows `762 sitshu 20 0 3488M 29756 6180 S 392.0 0.4 2h56:56 ./genservernode -B -- -root /usr/lib/4`.

PID	USER	PRI	NI	VIRT	RES	SHR	S	CPU%	MEM%	TIME+	Command
762	sitshu	20	0	3488M	29756	6180	S	392.0	0.4	2h56:56	./genservernode -B -- -root /usr/lib/4

The ratio of CPU time to REAL TIME to obtain effectiveness of cores used in the computation is
(24 min 21 sec / 6 min 15 sec) = **3.896**

Since the process is infinite, the process is halted by `ctrl + c`, to obtain the running time

Maximum zeros

: runtime of **15 minutes** contained **8 zeros**:

```
sitshu702700718 000000014DD9ED03A7C499651E9E3A557BE8A0B9ED8FBD8C9FE47A3C853FB702 7 0s  
sitshu330824868 00000000DD7D86A446161B6FFAE12200FBA3ED5A1103608014AD401BD4DD75D0 8 0s
```

Number of working machines

We could run our program with 1 server and 4 clients. Permissible number of clients is restricted by number of strings with client name that can be generated. Which we restricted to 62^5 .