Read Me for COP5615 DOS Project1

Group member:
Zhai keke UFID: 9869 - 7900
Xiaomeng Xie UFID: 9150 - 1938

Bits Coin Mining Usage:
Start Server: >mix escript.build
>./project1 4
Start Worker/Client: >mix escript.build
>./project1 IP(server's ip)

1. **Size of the work unit: the number of cores available in the system.**

   In Elxir/Erlang, strings are a subset of integers. In this project, we convert the string partition problem into Integer range partition problem. For instance, process1(Here I mean Erlang's process) goes through from 0 to 0xffffffff; Process 2(Erlang) goes through from 0x100000000 to 0x1fffffff, and so on, where each process can take care of 0xffffffff numbers. To reduce the runtime of each thread, a smaller range could apply, such as 0xffffff.

2. **The result of running program for ./project1 4**

```
$ ./project1 4
zhaikeke;4bp6GB2PE7mcFUnIKuUTTUIYdQmkoiRSaCLFYMInlznFdK2sM2B
     0000D2D5A364543E834B6398EEAA6AFD8B14AE99A0286ADBA38264F5D50621AE
zhaikeke;Tj0lC1oKf52orttntCc5XQAKrBeocNN2AemHosYetHmFjFtmHgJ1pP25uOdAC5aba9
     000066C54A6C178BC22E692D52481FCE6F4215D7A4FEC13328FFF5525A0F2FC0
zhaikeke;0P2itodc7   000047E5CBDED743695332BDF25927D8A0D4B8FE999EFE41E1C87B2C828DA275
zhaikeke;PWBpyq6GDerQx7GjTAKmbRB3BlbY1J8F0Q4TgDmYvnE4PKS4ew7qA2FrXFOn98EpRsu2cpuhbG
Rpfl0000626A4A55FAB96ED41D697D0D4831E0495E383516ED64B40E7AFF77045CB7
```

3. **The running time & ratio of CPU**

   The running time on 2-core machine:

```
$ /usr/bin/time ./project1 4
^CCommand terminated abnormally.
     3.20 real        8.32 user        0.52 sys
```

   CPU time 8.32 to real time 3.2 ~= 2.6 which is larger than 1. That means our project is parallelism.

Ratio of CPU on 2-core machine of command: ./project1 4:
%CPU = 363%

PID	COMMAND	%CPU	TIME	#TH	#WQ	#PORTS	MEM	PURG	CMPRS	PGRP
76984	top	3.5	00:00.81	1/1	0	20	2448K	0B	0B	76984
76982	erl_child_se	0.0	00:00.00	1	0	10	320K	0B	0B	76982
76978	beam.smp	363.8	00:39.91	32/4	0	46	22M	0B	0B	76978
76975	ReportCrash	0.0	00:00.67	4	1	95-	9632K-	0B	0B	76975
76968	QuickLookSat	0.0	00:00.09	2	1	45	3040K	0B	0B	76968
76966	com.apple.ap	0.0	00:01.56	4/1	2/1	245	18M	48K	2880K	76966
76934	helpd	0.0	00:00.03	2	1	36	40K	0B	956K	76934
76905	quicklookd	0.0	00:00.15	4	1	88	1984K	0B	1688K	76905
76902	mdworker	0.0	00:00.08	3	1	49	1760K	0B	1144K	76902
76901	mdworker	0.0	00:00.10	3	1	56	2232K	0B	768K	76901
76804	crashpad_han	0.0	00:00.03	4	1	28	316K	0B	612K	76803
76797	Code Helper	0.0	00:00.99	20	1	132	9276K	0B	23M	76792
76794	Code Helper	0.0	00:03.64	6	2	99	7900K	0B	16M	76792
76792	Electron	0.0	00:09.00	38	1	407-	21M-	0B	13M	76792
76696	mdworker	0.0	00:00.09	3	1	49	880K	0B	2200K	76696

Ratio of CPU on 4-core machine:
%CPU = 398%

ER	PR	NI	VIRT	RES	SHR	S	%CPU	%MEM	TIME+	COM
veti	20	0	2803012	31968	6652	S	398.0	0.1	1:29.10	bea
ot	20	0	224104	12156	8372	S	0.0	0.0	0:02.24	sys
ot	20	0	0	0	0	S	0.0	0.0	0:00.01	kth
ot	0	-20	0	0	0	S	0.0	0.0	0:00.00	kwo
ot	0	-20	0	0	0	S	0.0	0.0	0:00.00	mm_
ot	20	0	0	0	0	S	0.0	0.0	0:00.01	kso
ot	20	0	0	0	0	S	0.0	0.0	0:00.38	rcu
ot	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu
ot	20	0	0	0	0	S	0.0	0.0	0:00.34	rcu
ot	20	0	0	0	0	S	0.0	0.0	0:00.00	rcu
ot	rt	0	0	0	0	S	0.0	0.0	0:00.00	mig
ot	rt	0	0	0	0	S	0.0	0.0	0:00.00	wat
ot	20	0	0	0	0	S	0.0	0.0	0:00.00	cpu
ot	20	0	0	0	0	S	0.0	0.0	0:00.00	cpu
ot	rt	0	0	0	0	S	0.0	0.0	0:00.00	wat
ot	rt	0	0	0	0	S	0.0	0.0	0:00.00	mig
ot	20	0	0	0	0	S	0.0	0.0	0:00.00	kso

4. The coin with the most 0s you managed to find

The most 0's we managed to find is **5:**

\Users\zhaikeke\Documents\LeetCode\Elixir\project1>escript project1 5
zhaikeke;A0t3wMmBvsxSymF7YQoAIeA26WyOljE43ZqrnpFib4j8VDvmPa1sO4xHKqxId5jNsU
DfkDm3ThIdPZLfu
000003C51C97F0171B267BDEB29A754BC33EEC4D2FC08DCFA07AAAF127C2CB1E
zhaikeke;QeM38kCS73YLHy
00000EFBC088DC720351F26260588B20E2E41B31071D3EB3B809A3FF69C7F377
zhaikeke;plQWCTxD5ajSK6PDMVkjJ5GLwuqdNum4NLN0EvtCemZkHZmuNigH3JO1X7zqNBx
QTJRoQiQiE9aMsHvf0GFJ5AWUi5uA1v
000007BFC9E344DBF9A7C2BFA45F3FCB9C9B429784ADDDB7C2E8607A1939F407
zhaikeke;jXJE5Xgoe6yBXlse3tinm5bU0
00000CF3C016D52EEA7D273FA210B0BBBD30061E6EA39BC0E6D1798A0A2B81C4

5. **The largest number of working machines I were able to run my code with.**
   We used 2VM and 2 physical machines to run our codes.
   We implemented at most 8 workers.