COP 5612 – Distributed Operating Systems

Project – 1 (README)

Group members:
Suprith Reddy Gurudu (UF ID: 9961-2134)
Hima Tejaswi Gummadi (UF ID: 2455-9492)

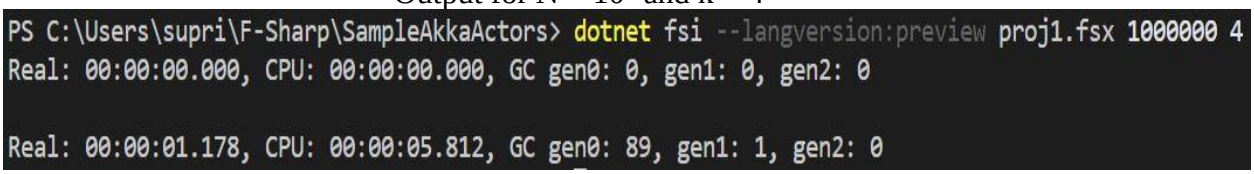
Implementation:

1. Size of the work unit that determined results in best performance = N / 100 as we use 100 worker actors controlled by a boss actor, i.e., each worker actor gets N / 100 subproblems on a single request from the boss actor. Increasing number of actors more than 100 did not actually improve the performance much. So, we implemented using N/100 worker actors.

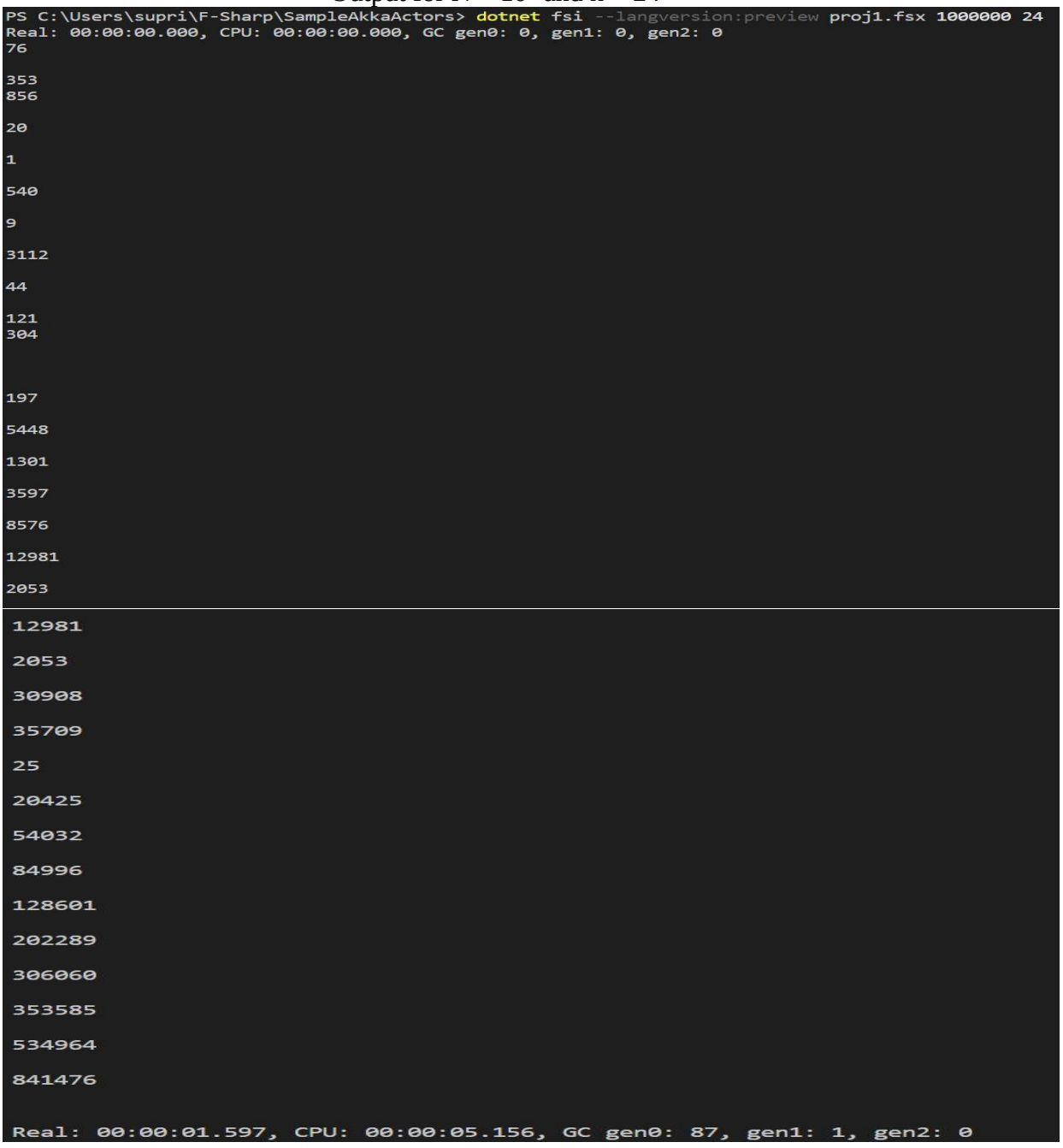
2. Result for running **dotnet fsi proj1.fsx 1000000 4**

   Note: *there is an error popping up with the VS Code on language version, so I have used a workaround by adding [--langversion:preview] when executing the program via command line and dependency in the settings.json as displayed below in the output image*

   Output for N = 10^6 and k = 4



Output for N = 10^6 and k = 24



3. Input with N = 10^6 and k = 4 did not produce any output as shown in the below snapshot. We got the CPU to real time ratio as 5812 / 1178 = 4.93. Along with this, we also used N = 10^6 and k = 24 and below is the result that we got with CPU time 05.156s and Real time 01.597s and ratio of CPU to real time is 5156 / 1597 = 3.23.

Running time for N = 10^6 and k = 4

- ⇨ CPU time: 00:00:05:812
- ⇨ Real time: 00:00:01:178
- ⇨ CPU / Real time ratio: 4.93

Running time for N = 10^6 and k = 24

- ⇨ CPU time: 00:00:05:156
- ⇨ Real time: 00:00:01:597
- ⇨ CPU / Real time ratio: 3.23

4. The largest input we managed to get the result for N = 10^9 and k = 20

Running time for N = 10^9 and k = 20

- ⇨ CPU time: 02:04:15.031
- ⇨ Real time: 00:18:51.267
- ⇨ CPU / Real time ratio: 11.02

Output for N = 10^9 and k = 20

```
PS C:\Users\supri\F-Sharp\SampleAkkaActors> dotnet fsi --langversion:preview proj1.fsx 1000000000 20
Real: 00:00:00.000, CPU: 00:00:00.000, GC gen0: 0, gen1: 0, gen2: 0
62780852

88700958

105393599

122086240

125818828

146244057

149976645

166669286

170401874

174134462

177867050

190827103

194559691

198292279

202024867

222450096

226182684
```

```
229915272
241113036
242875325
246607913
250340501
254073089
257805677
261538265
265270853
270765730
274498318
278230906
281963494
285696082
289428670
293161258
306121311
309853899
313586487
317319075
321051663
324784251
330279128
334011716
337744304
341476892
345209480
348942068
352674656
356407244
358169533
369367297
373099885
376832473
380565061
384297649
```

```
391762825
397257702
400990290
404722878
408455466
412188054
425148107
428880695
432613283
436345871
440078459
443811047
447543635
451276223
453038512
456771100
458741399
460503688
```

```
464236276
467968864
471701452
475434040
479166628
480928917
482899216
484661505
486631804
488394093
490364392
492126681
495859269
497621558
497829568
499591857
501562156
503324445
```

505086734

507057033

508819322

510789621

512551910

514522209

516284498

518254797

520017086

521987385

523749674

527482262

529244551

531214850

534947438

538680026

542412614

544174903

546145202

547907491

549877790

551640079

553610378

555372667

557342966

559105255

562837843

564808142

566570431

568540730

570303019

572065308

574035607

575797896

577768195

579530484

586995660

588965959

590728248

592698547

594460836

596223125

596431135

598193424

601926012

603896311

605658600

609391188

611153477

613123776

614886065

616856364

618618653

620588952

622351241

626083829

628054128

629816417

633549005

637281593

639043882

641014181

644746769

646509058

648479357

650241646

652211945

653974234

655944533

657706822

659677121

661439410

```
650241646

652211945

653974234

655944533

657706822

659677121

661439410

663409709

665171998

666934287

668904586

672637174

674399463

676369762

678132051

989446771

Real: 00:18:51.267, CPU: 02:04:15.031, GC gen0: 86992, gen1: 205, gen2: 11
```