PARCS-python

Distributed computations platform

Node types

- Master - dispatches work
- Worker - performs work

SDK setup

- Install gcloud sdk, login with `gcloud auth login`
- `gcloud projects create parcs-python`
- `gcloud config set project parcs-python`
- `# enable billing and compute APIs for project`
- `gcloud config set compute/zone europe-north1-a`
- `gcloud config set compute/region europe-north1`

Firewall setup

-
  ```
  gcloud compute
    firewall-rules create allow-all
    --direction=INGRESS --priority=1000
    --network=default --action=ALLOW
    --rules=all --source-ranges=0.0.0.0/0
  ```
- 🎉🎉🎉

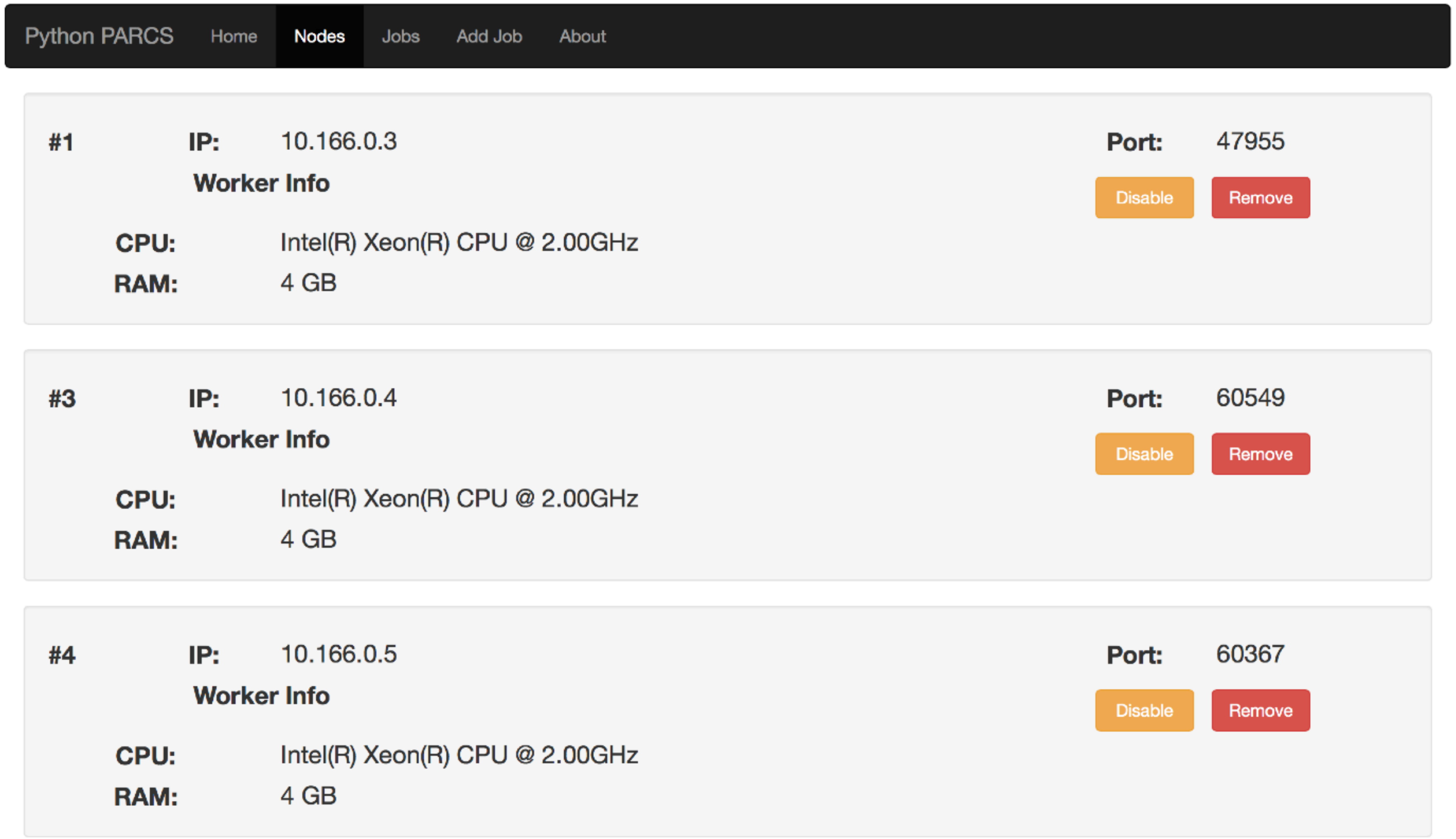
Instances

-
  ```
  gcloud compute instances
  create-with-container master
  --container-image=registry.hub.docker.com/
  hummer12007/parcs-node
  --container-env PARCS_ARGS="master"
  ```
- **Record master’s IP, e.g. 10.166.0.2**

-
  ```
  gcloud compute instances
  create-with-container worker1 worker2 worker3…
  --container-image=registry.hub.docker.com/
  hummer12007/parcs-node
  --container-env PARCS_ARGS="worker 10.166.0.2"
  ```

🎉🎉🎉🎉🎉🎉🎉🎉🎉

- Control panel accessible at http://**$MASTER_IP**:8080



#1 **IP:** 10.166.0.3 **Port:** 47955

Worker Info

CPU: Intel(R) Xeon(R) CPU @ 2.00GHz
RAM: 4 GB

#3 **IP:** 10.166.0.4 **Port:** 60549

Worker Info

CPU: Intel(R) Xeon(R) CPU @ 2.00GHz
RAM: 4 GB

#4 **IP:** 10.166.0.5 **Port:** 60367

Worker Info

CPU: Intel(R) Xeon(R) CPU @ 2.00GHz
RAM: 4 GB

Example problem

https://git.sr.ht/~hummer12007/parcs-python/tree/master/examples/scripts

```
from Pyro4 import expose

class Solver:
    def __init__(self, workers=None, input_file_name=None, output_file_name=None):
        self.input_file_name = input_file_name
        self.output_file_name = output_file_name
        self.workers = workers
        print("Inited")

    def solve(self):
        print("Job Started")
        print("Workers %d" % len(self.workers))
        n = self.read_input()
        step = n / len(self.workers)

        # map
        mapped = []
        for i in xrange(0, len(self.workers)):
            mapped.append(self.workers[i].mymap(i * step, i * step + step))

        print('Map finished: ', mapped)

        # reduce
        reduced = self.myreduce(mapped)
        print("Reduce finished: " + str(reduced))

        # output
        self.write_output(reduced)

        print("Job Finished")
```

```
   @staticmethod
    @expose
    def mymap(a, b):
        print (a, b)
        res = 0
        for i in xrange(a, b):
            res += i
        return res

    @staticmethod
    @expose
    def myreduce(mapped):
        output = 0
        for x in mapped:
            output += x.value
        return output

    def read_input(self):
        f = open(self.input_file_name, 'r')
        line = f.readline()
        f.close()
        return int(line)

    def write_output(self, output):
        f = open(self.output_file_name, 'w')
        f.write(str(output))
        f.write('\n')
        f.close()
```

Example problem

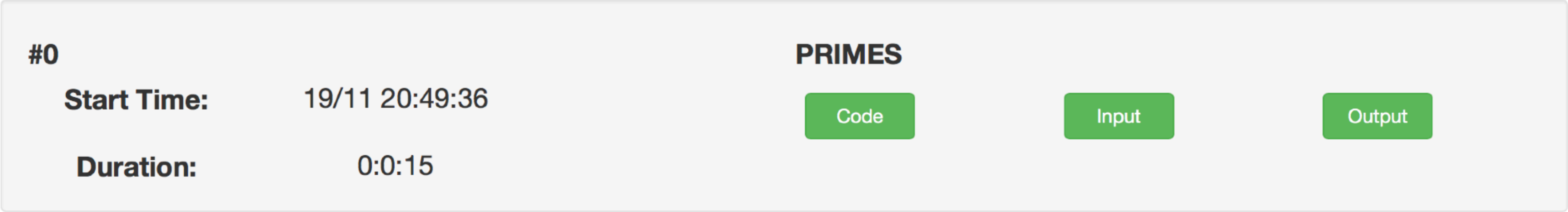
https://git.sr.ht/~hummer12007/parcs-python/tree/master/examples/scripts

```
mapped = []
for i in xrange(0, len(self.workers)):
    mapped.append(
        self.workers[i].mymap(i * step, i * step + step))
print('Map finished: ', mapped)
# reduce
reduced = self.myreduce(mapped)
print("Reduce finished: " + str(reduced))
```

```
@staticmethod
@expose
def mymap(a, b):
    print (a, b)
    res = 0
    for i in xrange(a, b):
        res += i
        return res
```

```
@staticmethod
@expose
def myreduce(mapped):
    output = 0
    for x in mapped:
        output += x.value
    return output
```

Done!



#0

PRIMES

Start Time: 19/11 20:49:36

Duration: 0:0:15

https://git.sr.ht/~hummer12007/parcs-python