Untitled

February 7, 2022

Interval Request Scheduling – Greedy Algorithm

```
[35]: import random

      # random pair (s, f) where s and f are integers between
      # 0 and 99, inclusive
      def random_request():
          return tuple(sorted(random.sample(range(0, 100), 2)))

      # make n random requests and return them all
      def make_requests(n):
          #requests = []
          #for i in range(n):
          #    requests.append(random_request())
          #return requests
          return [random_request() for i in range(n)]
```

```
[36]: make_requests(5)
```

```
[36]: [(32, 98), (3, 27), (36, 57), (41, 62), (74, 92)]
```

random_request()

```
[29]: random.sample([1,2,3],4)
```

```
---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
Input In [29], in <module>
----> 1 random.sample([1,2,3],4)

File /usr/local/Cellar/python@3.9/3.9.10/Frameworks/Python.framework/Versions/3
 ↪9/lib/python3.9/random.py:449, in Random.sample(self, population, k, counts)
    447 randbelow = self._randbelow
    448 if not 0 <= k <= n:
--> 449     raise ValueError("Sample larger than population or is negative")
    450 result = [None] * k
    451 setsize = 21        # size of a small set minus size of an empty list
```

```
ValueError: Sample larger than population or is negative
```

```
[ ]:
```