Instructions

-Run the script: Mini_A3.py, it will run the program for you.
-vi Results. It shows the matrices each time an update occurs. To view the final result, go to the bottom of the page.

Approach

As per the assignment specifications, I have between 5 to 12 processes (determined randomly at each program run) and 4 resources. This is my initialization step when the program runs. Once the initialization step is complete, I generate the maximum need of a resource that each process can have. This number is between 5 to 10 (determined randomly). I also generate a request array. For each process this is a random number that could be larger than the max need. Lastly, I generate the availability of the each resource. This is 0.6*the sum of the max need for that resources. Once the maximum needs and request are initialized, I run banker's algorithm to grant/ignore requests. If a given request will result in an unsafe state, I ignore that request and process the other ones. When a given process fulfills all its needs, it releases its resources so that the other processes can also complete. If all of the processes are able to satisfy their needs then I generate new requests and run Banker's algorithm again. The program ends when there are no requests remaining that will result in a safe state. This will most likely be due to the fact that a given process is asking for more resources than it requires.

Sample successful run (Example taken from class notes)

```
Generated new request.
Orignal Request Matrix
5
4
2

Hold Matrix
3
1
1

Need Matrix
2
3
1

Availability Matrix
2


State changed.
```

```
Orignal Request Matrix
5
4
2

Hold Matrix
5
1
1

Need Matrix
0
3
1

Availability Matrix
0


State after releasing resources.
Orignal Request Matrix
5
4
2

Hold Matrix
0
1
1

Need Matrix
0
3
1

Availability Matrix
5


State changed.
Orignal Request Matrix
5
4
2

Hold Matrix
0
4
```

```
1

Need Matrix
0
0
1

Availability Matrix
2


State after releasing resources.
Orignal Request Matrix
5
4
2

Hold Matrix
0
0
1

Need Matrix
0
0
1

Availability Matrix
6


State changed.
Orignal Request Matrix
5
4
2

Hold Matrix
0
0
2

Need Matrix
0
0
0

Availability Matrix
```

```
5


State after releasing resources.
Orignal Request Matrix
5
4
2

Hold Matrix
0
0
0

Need Matrix
0
0
0

Availability Matrix
7
```

Sample unsuccessful run (Example taken from class notes)

```
Generated new request.
Orignal Request Matrix
5
5
2

Hold Matrix
2
2
1

Need Matrix
3
3
1

Availability Matrix
1


Orignal Request Matrix
5
5
2
```

Hold Matrix
2
2
1

Need Matrix
3
3
1

Availability Matrix
1