Name:Mohamed yasser Ahmed mahmoud omara

ID:20191476257

Double server code Assignment

In [1]:
```
import pandas as pd
```

In [2]:
```
pd.set_option('display.max_columns', None)
```

In [3]:
```
randoms = [[1, 0, 32], [2, 94, 96], [3, 73, 89], [4, 70, 32], [5, 82, 67],
[6, 25, 48], [7, 35, 63], [8, 61, 99], [10, 42, 98], [11, 48, 66], [12, 26,
85], [13, 88, 58], [14, 31, 6], [15, 90, 39], [
    16, 55, 15], [17, 95, 2], [18, 58, 48], [19, 70, 63], [20, 15, 85], [2
1, 73, 61], [22, 65, 40], [23, 74, 16], [24, 75, 18], [25, 98, 52], [26, 7
2, 71], [27, 59, 16], [28, 85, 34], [29, 98, 96], [30, 21, 90]]
```

In [4]:
```
baker = {"service time": [3, 4, 5, 6],
         "probability": [0.35, 0.25, 0.2, 0.2],
         "cumulative probability": [0.35, 0.6, 0.8, 1],
         "random numbers start": [0, 35, 60, 80],
         "random numbers end": [34, 59, 79, 99]
         }
```

In [5]:
```
able = {"service time": [2, 3, 4, 5],
        "probability": [0.3, 0.28, 0.25, 0.17],
        "cumulative probability": [0.3, 0.58, 0.83, 1],
        "random numbers start": [0, 30, 58, 83],
        "random numbers end": [29, 57, 82, 99]
        }
```

In [6]:
```
iat = {
    "inter arrival time": [1, 2, 3, 4],
    "probability": [0.25, 0.4, 0.2, 0.15],
    "cumulative probability": [0.25, 0.65, 0.85, 1],
    "random numbers start": [0, 25, 65, 85],
    "random numbers end": [24, 64, 84, 99]
}
```

In [7]:
```
randomsdf = pd.DataFrame(randoms, columns=["customerId", "IAT", "ST"])
```

In [8]:
```
abledf = pd.DataFrame(able)
```

In [9]:
```
bakerdf = pd.DataFrame(baker)
```

In [10]:

```
iatdf = pd.DataFrame(iat)
```

In [11]:

```
result = pd.DataFrame({}, columns=["Call ID", "RN - IAT", "IAT", "Clock", "RN - ST", "Able ST Begins", "Able ST", "Able ST Ends",
                      "Baker ST Begins", "Baker ST", "Baker ST Ends", "Queuing Time", "Time Spent in System", "Able - Idle Time", "Baker - Idle Time"])
```

In [12]:

```
def getservicetime(rst, df):
    return df[(rst >= df["random numbers start"])
              & (rst <= df["random numbers end"])]["service time"].to_string(index=False)
```

In [13]:

```
def getiat(riat):
    return iatdf[(riat >= iatdf["random numbers start"])
                 & (riat <= iatdf["random numbers end"])]["inter arrival time"].to_string(index=False)
```

In [14]:

```
ebaker = 0
eable = 0
```

In [15]:

```
def usebaker(riat, iat, clock, rst):
    global ebaker
    begin = max(clock, ebaker)
    service = int(getservicetime(randomsdf["ST"][ind], bakerdf))
    end = begin + service
    time_in_sys = end - clock
    idle = begin - ebaker
    waiting = begin - clock
    row = {"Call ID": ind + 1, "RN - IAT": riat, "IAT": iat, "Clock": clock, "RN - ST": rst,
           "Baker ST Begins": begin, "Baker ST": service, "Baker ST Ends": end, "Queuing Time": waiting, "Time Spent in System": time_in_sys,  "Baker - Idle Time": idle}
    ebaker = end
    return row
```

In [16]:

```
def useable(riat, iat, clock, rst):
    global eable
    begin = max(clock, eable)
    service = int(getservicetime(randomsdf["ST"][ind], abledf))
    end = begin + service
    time_in_sys = end - clock
    idle = begin - eable
    waiting = begin - clock
    row = {"Call ID": ind + 1, "RN - IAT": riat, "IAT": iat, "Clock": clock, "RN - ST": rst,
           "Able ST Begins": begin, "Able ST": service, "Able ST Ends": end, "Queuing Time": waiting, "Time Spent in System": time_in_sys,  "Able - Idle Time": idle}
    eable = end
    return row
```

In [17]:

```
# baker starts if both are idle
for ind in randomsdf.index:
    if ind == 0:
        riat = int(randomsdf["IAT"][ind])
        iat = 0
        clock = 0
        rst = randomsdf["ST"][ind]
        row = usebaker(riat, iat, clock, rst)
        result = result.append(row,  ignore_index=True)
    else:
        riat = int(randomsdf["IAT"][ind])
        iat = int(getiat(riat))
        clock = iat + int(result["Clock"][ind - 1])
        rst = int(randomsdf["ST"][ind])
        if clock >= ebaker:
            row = usebaker(riat, iat, clock, rst)
            result = result.append(row,  ignore_index=True)
        elif clock >= eable:
            row = useable(riat, iat, clock, rst)
            result = result.append(row,  ignore_index=True)
        elif ebaker <= eable:
            row = usebaker(riat, iat, clock, rst)
            result = result.append(row,  ignore_index=True)
        elif ebaker > eable:
            row = useable(riat, iat, clock, rst)
            result = result.append(row,  ignore_index=True)
```

In [18]:

```
result.to_csv('Double server simulation code .csv')
```

In [19]:

```
Doubleservertable=pd.read_csv('Double server simulation code .csv')
```

```
In [20]: Doubleservertable
```

```
Out[20]:
```

	Unnamed: 0	Call ID	RN - IAT	IAT	Clock	RN - ST	Able ST Begins	Able ST	Able ST Ends	Baker ST Begins	Baker ST	Baker ST Ends	Queu T
0	0	1.0	0.0	0.0	0.0	32.0	NaN	NaN	NaN	0.0	3.0	3.0	
1	1	2.0	94.0	4.0	4.0	96.0	NaN	NaN	NaN	4.0	6.0	10.0	
2	2	3.0	73.0	3.0	7.0	89.0	7.0	5.0	12.0	NaN	NaN	NaN	
3	3	4.0	70.0	3.0	10.0	32.0	NaN	NaN	NaN	10.0	3.0	13.0	
4	4	5.0	82.0	3.0	13.0	67.0	NaN	NaN	NaN	13.0	5.0	18.0	
5	5	6.0	25.0	2.0	15.0	48.0	15.0	3.0	18.0	NaN	NaN	NaN	
6	6	7.0	35.0	2.0	17.0	63.0	NaN	NaN	NaN	18.0	5.0	23.0	
7	7	8.0	61.0	2.0	19.0	99.0	19.0	5.0	24.0	NaN	NaN	NaN	
8	8	9.0	42.0	2.0	21.0	98.0	NaN	NaN	NaN	23.0	6.0	29.0	
9	9	10.0	48.0	2.0	23.0	66.0	24.0	4.0	28.0	NaN	NaN	NaN	
10	10	11.0	26.0	2.0	25.0	85.0	28.0	5.0	33.0	NaN	NaN	NaN	
11	11	12.0	88.0	4.0	29.0	58.0	NaN	NaN	NaN	29.0	4.0	33.0	
12	12	13.0	31.0	2.0	31.0	6.0	NaN	NaN	NaN	33.0	3.0	36.0	
13	13	14.0	90.0	4.0	35.0	39.0	35.0	3.0	38.0	NaN	NaN	NaN	
14	14	15.0	55.0	2.0	37.0	15.0	NaN	NaN	NaN	37.0	3.0	40.0	
15	15	16.0	95.0	4.0	41.0	2.0	NaN	NaN	NaN	41.0	3.0	44.0	
16	16	17.0	58.0	2.0	43.0	48.0	43.0	3.0	46.0	NaN	NaN	NaN	
17	17	18.0	70.0	3.0	46.0	63.0	NaN	NaN	NaN	46.0	5.0	51.0	
18	18	19.0	15.0	1.0	47.0	85.0	47.0	5.0	52.0	NaN	NaN	NaN	
19	19	20.0	73.0	3.0	50.0	61.0	NaN	NaN	NaN	51.0	5.0	56.0	
20	20	21.0	65.0	3.0	53.0	40.0	53.0	3.0	56.0	NaN	NaN	NaN	
21	21	22.0	74.0	3.0	56.0	16.0	NaN	NaN	NaN	56.0	3.0	59.0	
22	22	23.0	75.0	3.0	59.0	18.0	NaN	NaN	NaN	59.0	3.0	62.0	
23	23	24.0	98.0	4.0	63.0	52.0	NaN	NaN	NaN	63.0	4.0	67.0	
24	24	25.0	72.0	3.0	66.0	71.0	66.0	4.0	70.0	NaN	NaN	NaN	
25	25	26.0	59.0	2.0	68.0	16.0	NaN	NaN	NaN	68.0	3.0	71.0	
26	26	27.0	85.0	4.0	72.0	34.0	NaN	NaN	NaN	72.0	3.0	75.0	
27	27	28.0	98.0	4.0	76.0	96.0	NaN	NaN	NaN	76.0	6.0	82.0	
28	28	29.0	21.0	1.0	77.0	90.0	77.0	5.0	82.0	NaN	NaN	NaN	