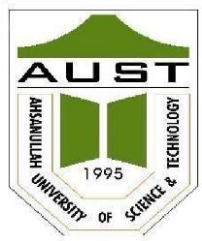
AHSANULLAH UNIVERSITY OF SCIENCE AND TECHNOLOGY

DHAKA-1208, BANGLADESH.



Department of Computer Science and Engineering

Spring 2019

Program: Bachelor of Science in Computer Science and Engineering

Course No: IPE 4111

Course Title: Industrial Management

Assignment

Date of Submission: 19/11/2019

Submitted to

Kazi Wahadul Hasan

Lecturer

Department of Mechanical and Production Engineering

Submitted By

Name : Robiul Hasan Nowshad

Student ID : 16.01.04.061

Lab Group : B1

Task: Implementation of Johnson's Rule

Implementation:

Here, I implemented Johnson's Rule for 2 machine job sequence with **Python.**

Data set: JR.csv

Job	M1	M2
A	3	5
B	4	6
C	2	7
D	6	9
E	1	2

Source Code: JR_Problem.py

```
1. # -*- coding: utf-8 -*-
2. """
3. Created on Mon Nov 18 03:12:40 2019
4. @author: nowshad
5. """
6. import pandas as pd
7. import numpy as np
8.
9. def Find_seq(DataSet):
10.          l=DataSet.shape[0]
11.          seq=[]
12.          for i in range(l):
13.              seq.insert(i,0)
14.          i,j=0,l-1
15.          while DataSet.empty==False:
16.              M1_min=DataSet['M1'].min()
17.              M2_min=DataSet['M2'].min()
18.              if M1_min>M2_min:
19.                  index=DataSet['M2'].idxmin()
20.                  job=DataSet['Job'][index]
21.                  seq[j]=job
22.                  j-=1
23.                  DataSet = DataSet.drop([index], axis=0)
24.              else:
25.                  index=DataSet['M1'].idxmin()
26.                  job=DataSet['Job'][index]
27.                  seq[i]=job
28.                  i+=1
29.                  DataSet = DataSet.drop([index], axis=0)
30.          return seq
```

```
31.
32.     def FindInOut_Table(DataSet,seq):
33.         InOut={'JobSeq':['M1_in','M1_out','M2_in','M2_out']}

34.         for i in range(DataSet.shape[0]):
35.             if i==0:
36.                 idx = DataSet[DataSet['Job']==seq[i]].index.v
    alues.astype(int)
37.                 InOut[seq[i]]=[0,DataSet['M1'][idx[0]],DataSe
    t['M1'][idx[0]],DataSet['M1'][idx[0]]+DataSet['M2'][idx[0]]]
38.             else:
39.                 idx = DataSet[DataSet['Job']==seq[i]].index.v
    alues.astype(int)
40.                 M1outTemp=InOut[seq[i-
    1]][1]+DataSet['M1'][idx[0]];
41.                 if M1outTemp>InOut[seq[i-1]][3]:
42.                     M2inTemp=M1outTemp
43.                 else:
44.                     M2inTemp=InOut[seq[i-1]][3]
45.                 InOut[seq[i]]=[InOut[seq[i-
    1]][1],M1outTemp,M2inTemp,M2inTemp+DataSet['M2'][idx[0]]]
46.
47.         InOutTable=pd.DataFrame.from_dict(InOut, orient='inde
    x')
48.         return InOutTable
49.
50.     def Calculate_FlowAndIdleTime(InOutTable):
51.         print("\nTotal Flow Time: ",InOutTable[3][InOutTable.
    shape[0]-1])
52.         M1_IdleTime=InOutTable[3][InOutTable.shape[0]-1]-
    InOutTable[1][InOutTable.shape[0]-1]
53.         print("M1 Idle Time: ", M1_IdleTime)
54.         M2_IdleTime=0
55.         for i in range(InOutTable.shape[0]-1):
56.             if i==0:
57.                 M2_IdleTime=InOutTable[2][1]-
    InOutTable[0][1]
58.
59.             else:
60.                 M2_IdleTime=M2_IdleTime+(InOutTable[2][i+1]-
    InOutTable[3][i])
61.         print("M2 Idle Time: ", M2_IdleTime)
62.
63.     #Main
64.     dataset = pd.read_csv('G:\AUST4.1\IPE\ASSgnmt\JR.csv')
65.     print("\nSize:- ", dataset.shape)
66.     print("\n", dataset)
67.     seq=Find_seq(dataset)
68.     print("\nSequence: ",seq,"\n")
69.     InOutTable=FindInOut_Table(dataset,seq)
```

```
70.      print(InOutTable)
71.      Calculate_FlowAndIdleTime(InOutTable)
```

Output: