Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

A.Y. 2022-2023

Subject: Process Organization and Architecture **SAP ID: 60004220253 – Devansh Mehta**

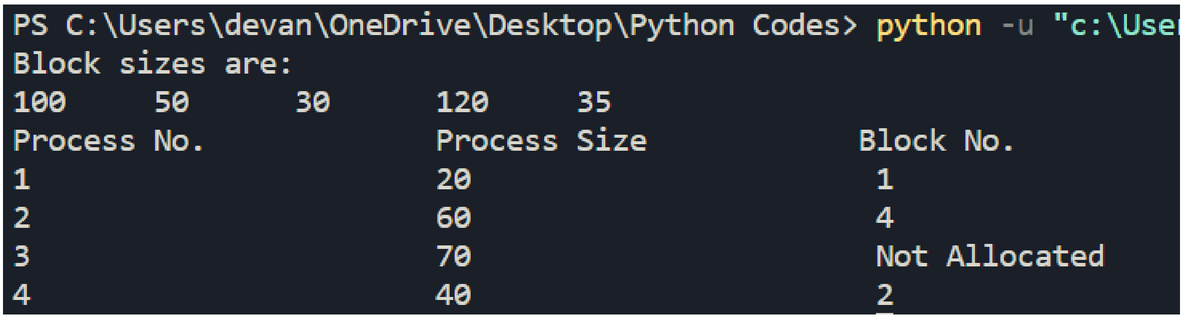
Experiment No: 03

Aim: To implement Best Fit and First Fit Memory Allocation policy.

Code:

First Fit

```
def FirstFit(block_Size, blocks, process_Size, proccesses):
    allocate = [-1] * proccesses
    occupied = [False] * blocks
    for i in range(proccesses):
        for j in range(blocks):
            if not occupied[j] and (block_Size[j] >= process_Size[i]):
                allocate[i] = j
                occupied[j] = True
                break
    print("Block sizes are:")
    for i in range(len(block_Size)):
        print(block_Size[i],end='\t')
    print("\nProcess No. \t\tProcess Size \t\tBlock No.")
    for i in range(proccesses):
        print(str(i + 1) + "\t\t\t" + str(process_Size[i]) + "\t\t\t", end=" ")
        if allocate[i] != -1:
            print(allocate[i] + 1)
        else:
            print("Not Allocated")
block_Size = [100, 50, 30, 120, 35]
process_Size = [20, 60, 70, 40]
m = len(block_Size)
n = len(process_Size)
FirstFit(block_Size, m, process_Size, n)
```

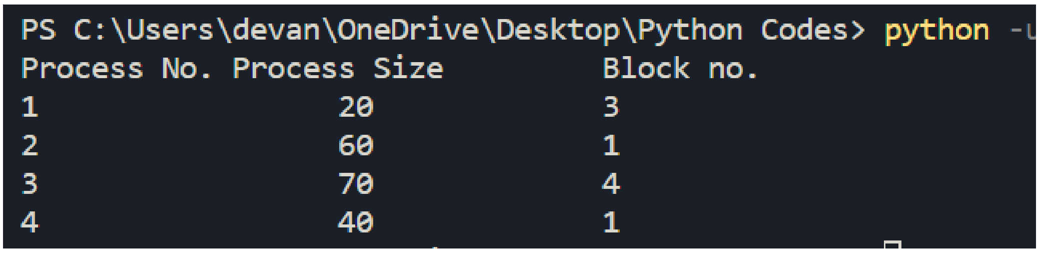


```
PS C:\Users\devan\OneDrive\Desktop\Python Codes> python -u "c:\Use
Block sizes are:
100     50      30      120     35
Process No.             Process Size            Block No.
1                       20                       1
2                       60                       4
3                       70                       Not Allocated
4                       40                       2
```

Subject: Process Organization and Architecture **SAP ID: 60004220253 – Devansh Mehta**

Best Fit

```
def bestFit(blockSize, m, processSize, n):
    allocation = [-1] * n
    for i in range(n):
        bestIdx = -1
        for j in range(m):
            if blockSize[j] >= processSize[i]:
                if bestIdx == -1:
                    bestIdx = j
                elif blockSize[bestIdx] > blockSize[j]:
                    bestIdx = j
        if bestIdx != -1:
            allocation[i] = bestIdx
            blockSize[bestIdx] -= processSize[i]
    print("Process No. Process Size  Block no.")
    for i in range(n):
        print(i + 1, "       ", processSize[i],
                          end = "       ")
        if allocation[i] != -1:
            print(allocation[i] + 1)
        else:
            print("Not Allocated")
if __name__ == '__main__':
    blockSize = [100, 50, 30, 120, 35]
    processSize = [20,60,70,40]
    m = len(blockSize)
    n = len(processSize)
    bestFit(blockSize, m, processSize, n)
```



```
PS C:\Users\devan\OneDrive\Desktop\Python Codes> python -
Process No. Process Size         Block no.
1                 20             3
2                 60             1
3                 70             4
4                 40             1
```