**Ex. No.: 10a)**
**Date:** 09/04/2025

## BEST FIT

**Aim:**

To implement Best Fit memory allocation technique using Python.

**Algorithm:**

1. Input memory blocks and processes with sizes
2. Initialize all memory blocks as free.
3. Start by picking each process and find the minimum block size that can be assigned to current process
4. If found then assign it to the current process.
5. If not found then leave that process and keep checking the further processes.

**Program Code:**

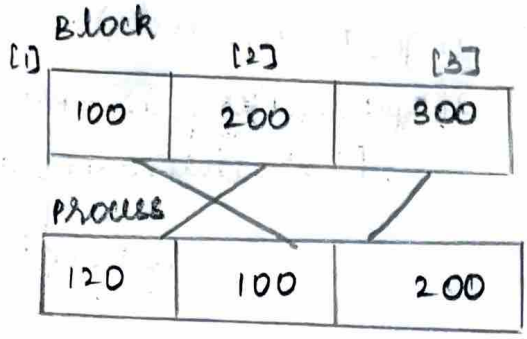
```
# no of blocks
num_blocks = int(input("Enter the no. of memory blocks: "))
# input of block sizes
block_sizes = []
print("Enter sizes of memory blocks:")
for i in range(num_blocks):
    size = int(input(f"size of block {i+1}: "))
    block_sizes.append(size)
# no of process
num_processes = int(input("\nEnter the number of Processes: "))
# input of process sizes
process_sizes = []
print("Enter sizes of processes:")
for i in range(num_processes):
    size = int(input(f"size of process {i+1}: "))
    Process_sizes.append(size)
```

```
allocation = [-1] * num_processes

for i in range(num_processes):
        best_index = -1
        for j in range(num_blocks):
                if block_sizes[j] >= process_sizes[i]:
                    if best_index == -1 or block_sizes[j] <
                                        block_sizes[best_index]
                        best_index = j

        if best_index != -1:
                allocation[i] = best_index
                block_sizes[best_index] -= process_sizes[i]

print("\nProcess No.\tProcess size\tBlock Allocated")
for i in range(num_processes):
        print(f"{i+1}\t\t{process_sizes[i]}\t\t", end="")
        if allocation[i] != -1:
                print(f"{allocation[i] + 1}")
        else:
                print("Not Allocated")
```




| Process | Process_Size | Block No | Fragment |
|---|---|---|---|
| $P_1$ | 120 | 2 | [200-120] = 80 |
| $P_2$ | 100 | 1 | [100-100] = 0 |
| $P_3$ | 200 | 3 | [300-200] = 100 |

**Sample Output:**

| Process No. | Process Size | Block no. |
|---|---|---|
| 1 | 212 | 4 |
| 2 | 417 | 2 |
| 3 | 112 | 3 |
| 4 | 426 | 5 |

Output:

Enter no. of memory blocks: 3

Enter sizes of memory blocks:

100

200

300

Enter no. of processes : 3

Enter sizes of processes:

120

100

200

| Process No. | Process Size | Block Allocated |
|---|---|---|
| 1 | 120 | 2 |
| 2 | 100 | 1 |
| 3 | 200 | 3 |

**Result:**

Thus the Python program to implement Best fit was executed successfully.