**Ex No.: 10a BEST FIT**

Date : 09.04.2025

**Aim :**

To implement Best Fit memory allocation technique using Python.

**Code:**

def best\_fit():

blocks\_input = input("Enter memory block sizes: ")

blocks = [int(size) for size in blocks\_input.split()]

pros\_input = input("Enter process sizes: ")

pros = [int(size) for size in pros\_input.split()]

alloc = [-1] \* len(pros)

rem = blocks.copy()

for i in range(len(pros)):

best = None

for j in range(len(rem)):

if rem[j] >= pros[i]:

if best is None or rem[j] < rem[best]:

best = j

if best is not None:

alloc[i] = best

rem[best] -= pros[i]

print("\nProcess No.\tProcess Size\tBlock No.")

for i in range(len(pros)):

print(f"{i+1}\t\t{pros[i]}\t\t", end="")

if alloc[i] != -1:

print(alloc[i] + 1)

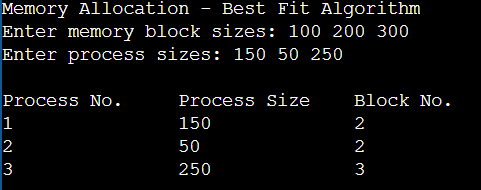
else:

print("Not Allocated")

print("Memory Allocation - Best Fit Algorithm")

best\_fit()

**Output:**

****

**Result:**

Thus the python program to implement Best Fit memory allocation technique has been executed successfully.