```cpp
#include <bits/stdc++.h>
using namespace std;

struct Process
{
    int id;
    int size;
    int allocation;
    bool isGiven = false;
};

struct Memory
{
    int size;
    int free;
    int allocated;
    bool isTaken = false;
    int extfrag;
    int givenProcessId = -1;
};

int m;
int n;
int external_fragmentation = 0;
int internal_fragmentation = 0;



void firstFit(Process p[], int n, Memory mem[], int m)
{
    int j = 0;
    for (int i = 0; i < n; i++)
    {
        int bestIdx = -1;
        for (int j = 0; j < m; j++)
        {
            if (mem[j].size >= p[i].size)
            {
                if (bestIdx == -1)
                    bestIdx = j;
                else if (mem[bestIdx].size > mem[j].size)
                    bestIdx = j;
            }
        }

        if (bestIdx != -1)
        {
            mem[bestIdx].isTaken = true;
            p[i].isGiven = true;
            mem[bestIdx].givenProcessId = p[i].id;
            mem[bestIdx].free -= p[i].size;
            mem[bestIdx].size -= p[i].size;
            p[i].allocation = bestIdx + 1;
            mem[bestIdx].allocated = p[i].id;

        }
    }


}

```

```
62 void calcfrag(Process p[], int n, Memory mem[], int m)
63 {
64     int flag = 0;
65     for (int i = 0; i < m; i++)
66     {
67         if (mem[i].givenProcessId == -1)
68         {
69             flag = 1;
70             break;
71         }
72     }
73     if (flag == 0)
74     {
75         external_fragmentation = 0;
76     }
77     else
78     {
79         for (int i = 0; i < m; i++)
80         {
81             if (mem[i].isTaken != true || mem[i].givenProcessId == -1)
82             {
83                 external_fragmentation += mem[i].size;
84             }
85         }
86     }
87
88     for (int i = 0; i < m; i++)
89     {
90         if (mem[i].isTaken != false)
91         {
92             internal_fragmentation += mem[i].free;
93         }
94     }
95 }
96
97 void printTable(Process P[], int n, Memory mem[], int m, int memorySize[])
98 {
99
100     for (int i = 0; i < m; i++)
101     {
102         if (mem[i].free == memorySize[i])
103         {
104             mem[i].free = 0;
105         }
106     }
107
108     cout << "\nTable-->(-1 Denotes Unallocated process)\n";
109     int i;
110
111     puts("+-----+------------+--------------+-----------------+");
112     puts("| BNO | Block Size | Process All. | Internal Fragm. |");
113     puts("+-----+------------+--------------+-----------------+");
114
115     for (i = 0; i < m; i++)
116     {
117         printf("| %2d  |     %2d     |      %2d      |        %3d      |\n", i + 1,
memorySize[i], mem[i].givenProcessId, mem[i].free);
118         puts("+-----+------------+--------------+-----------------+");
119     }
120
121     cout << "External Fragmentation: " << external_fragmentation << endl;
```

```
122     cout << "Internal Fragmentation: " << internal_fragmentation << endl;
123 }
124 
125 int main()
126 {
127     cout << "\nEnter the number of memory blocks: ";
128     cin >> m;
129     Memory mem[m];
130     int memorySize[m];
131     for (int i = 0; i < m; i++)
132     {
133         cout << "\n";
134         cout << "Enter the size of the memory block " << i + 1 << ": ";
135         cin >> mem[i].size;
136         mem[i].free = mem[i].size;
137         mem[i].allocated = -1;
138         mem[i].extfrag = 0;
139         memorySize[i] = mem[i].size;
140     }
141 
142     cout << "\nEnter the number of processes: ";
143     cin >> n;
144     Process p[n];
145     for (int i = 0; i < n; i++)
146     {
147         p[i].id = i + 1;
148         cout << "\n";
149         cout << "\nEnter the size of the process" << p[i].id << ": ";
150         cin >> p[i].size;
151     }
152 
153     firstFit(p, n, mem, m);
154 
155     calcfrag(p, n, mem, m);
156 
157     printTable(p, n, mem, m, memorySize);
158 }
```

Visual Studio Code Thu Mar 31 2:19 PM 83.2kb/s 28.7kb/s 12%

Code - os lab - Visual Studio Code

File Edit Selection View Go Run Terminal Help

bestfit.cpp Code

```
anzal  anzal  ../4th sem/os lab/p11  cd "/home/anzal/Desktop/4th sem/os lab/p11/" && g++ bestfit.cpp -o bestfit && "/home/anzal/Desktop/4th sem/os lab/p11/"bestfit
Enter the number of memory blocks: 5
Enter the size of the memory block 1: 100
Enter the size of the memory block 2: 500
Enter the size of the memory block 3: 200
Enter the size of the memory block 4: 300
Enter the size of the memory block 5: 600
Enter the number of processes: 4

Enter the size of the process1: 212

Enter the size of the process2: 417

Enter the size of the process3: 112

Enter the size of the process4: 426
Table-->(-1 Denotes Unallocated process)
+-----+------------+--------------+-----------------+
| BNO | Block Size | Process All. | Internal Fragm. |
+-----+------------+--------------+-----------------+
|  1  |     100    |      -1      |          0      |
+-----+------------+--------------+-----------------+
|  2  |     500    |       2      |         83      |
+-----+------------+--------------+-----------------+
|  3  |     200    |       3      |         88      |
+-----+------------+--------------+-----------------+
|  4  |     300    |       1      |         88      |
+-----+------------+--------------+-----------------+
|  5  |     600    |       4      |        174      |
+-----+------------+--------------+-----------------+
External Fragmentation: 100
Internal Fragmentation: 433

anzal  anzal  ../4th sem/os lab/p11
```

Run Testcases 0 0 Go Live