Program: First Fit Memory Management

```
// C++ implementation of First - Fit algorithm
#include<bits/stdc++.h>
using namespace std;
// Function to allocate memory to
// blocks as per First fit algorithm
void firstFit(int blockSize[], int m,
int processSize[], int n)
{
// Stores block id of the
// block allocated to a process
int allocation[n];
// Initially no block is assigned to any process
memset(allocation, -1, sizeof(allocation));
// pick each process and find suitable blocks
// according to its size ad assign to it
for (int i = 0; i < n; i++)
{
for (int j = 0; j < m; j++)
{
if (blockSize[j] >= processSize[i])
{
// allocate block j to p[i] process
allocation[i] = j;
// Reduce available memory in this block.
blockSize[j] -= processSize[i];
break;
```

```
}

}

}

cout << "\nProcess No.\tProcess Size\tBlock no.\n";

for (int i = 0; i < n; i++)

{

cout << " " << i+1 << "\t\t"

<< processSize[i] << "\t\t";

if (allocation[i] != -1)

cout << allocation[i] + 1;

else

cout << "Not Allocated";

cout << endl;

}

}

// Driver code

int main()

{

int blockSize[] = {100, 500, 200, 300, 600};

int processSize[] = {212, 417, 112, 426};

int m = sizeof(blockSize) / sizeof(blockSize[0]);

int n = sizeof(processSize) / sizeof(processSize[0]);

firstFit(blockSize, m, processSize, n);

return 0 ;

}
```

OUTPUT:

Process No. Process Size Block no.

1 212 2

2 417 5

3 112 2

4 426 Not Allocated