**Algorithm 7 :-** *Write a program to demonstrate the use of First Fit memory allocation algorithms.*

import java.util.Scanner;

import java.util.Random;

class FirstFit {

static void firstFit(int blockSize[], int m, int processSize[], int n) {

int allocation[] = new int[n];

for (int i = 0; i < allocation.length; i++)

allocation[i] = -1;

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

for (int j = 0; j < m; j++) {

if (blockSize[j] >= processSize[i]) {

allocation[i] = j;

blockSize[j] -= processSize[i];

break;

}

}

}

System.out.println("\nProcess No.\tProcess Size\tBlock Size\tBlock no.");

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

System.out.print(" " + (i+1) + "\t\t" + processSize[i] + "\t\t"+blockSize[i]+ "\t\t");

if (allocation[i] != -1)

System.out.print(allocation[i] + 1);

else

System.out.print("Not Allocated");

System.out.println();

}

}

// Driver Code

public static void main(String[] args) {

Random random = new Random();

Scanner input = new Scanner(System.in);

System.out.print("Enter no. of Processes : ");

int in = input.nextInt();

System.out.print("Enter no. of Blocks : ");

int im = input.nextInt();

int processSize[] = new int[in];

int blockSize[] = new int[im];

for (int j = 0; j < in; j++) {

processSize[j] = random.nextInt(500);

}

for (int i = 0; i < im; i++) {

blockSize[i] = random.nextInt(500);

}

int m = blockSize.length;

int n = processSize.length;

firstFit(blockSize, m, processSize, n);

}

}

**OUTPUT :-**

****