```
import java.util.Scanner;

public class Main
{
    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 no.");
        for (int i = 0; i < n; i++)
        {
            System.out.print(" " + (i+1) + "\t\t" +
                             processSize[i] + "\t\t");
```

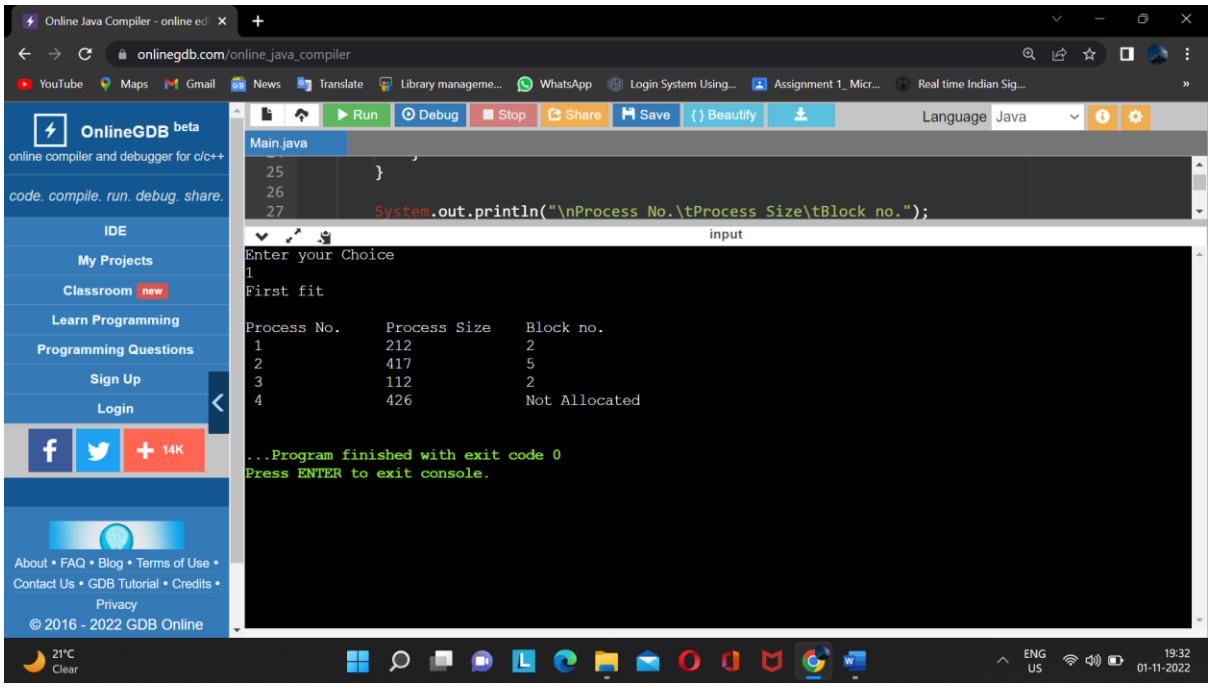
```
        if (allocation[i] != -1)
            System.out.print(allocation[i] + 1);
        else
            System.out.print("Not Allocated");
        System.out.println();
    }
}

 static void bestFit(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++)
    {
        int bestIdx = -1;
        for (int j=0; j<m; j++)
        {
            if (blockSize[j] >= processSize[i])
            {
                if (bestIdx == -1)
                    bestIdx = j;
                else if (blockSize[bestIdx] > blockSize[j])
                    bestIdx = j;
            }
        }

        if (bestIdx != -1)
        {
```

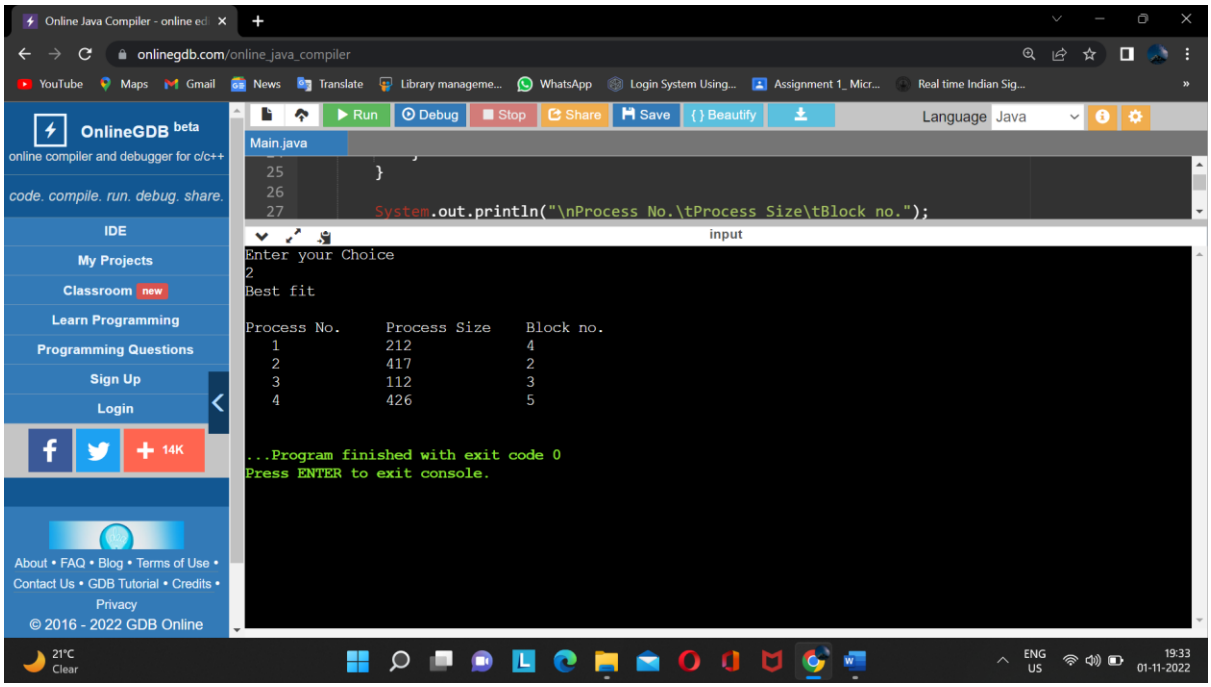
```
            allocation[i] = bestIdx;
            blockSize[bestIdx] -= processSize[i];
        }
    }

    System.out.println("\nProcess No.\tProcess Size\tBlock no.");
    for (int i = 0; i < n; i++)
    {
        System.out.print("   " + (i+1) + "\t\t" + processSize[i] + "\t\t");
        if (allocation[i] != -1)
            System.out.print(allocation[i] + 1);
        else
            System.out.print("Not Allocated");
        System.out.println();
    }
}

 static void worstFit(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++)
    {
        int wstIdx = -1;
        for (int j=0; j<m; j++)
        {
            if (blockSize[j] >= processSize[i])
```

```
            {
                if (wstIdx == -1)
                    wstIdx = j;
                else if (blockSize[wstIdx] < blockSize[j])
                    wstIdx = j;
            }
        }

        if (wstIdx != -1)
        {

            allocation[i] = wstIdx;
            blockSize[wstIdx] -= processSize[i];
        }
    }

    System.out.println("\nProcess No.\tProcess Size\tBlock no.");
    for (int i = 0; i < n; i++)
    {
        System.out.print("   " + (i+1) + "\t\t" + processSize[i] + "\t\t");
        if (allocation[i] != -1)
            System.out.print(allocation[i] + 1);
        else
            System.out.print("Not Allocated");
        System.out.println();
    }
}

 public static void main(String[] args)
{
    int blockSize[] = {100, 500, 200, 300, 600};
```

```
        int processSize[] = {212, 417, 112, 426};
        int m = blockSize.length;
        int n = processSize.length;
        Scanner obj = new Scanner(System.in);
        System.out.println("Enter your Choice");
        int choice = obj.nextInt();

        switch(choice){
           case 1:
              System.out.println("First fit");
              firstFit(blockSize, m, processSize, n);
              break;
          case 2:
             System.out.println("Best fit");
             bestFit(blockSize, m, processSize, n);
             break;
          case 3:
             System.out.println("Worst fit");
             worstFit(blockSize, m, processSize, n);
             break;
          default:
          System.out.println("Invalid choice");

        }



    }



}
```

## Output :


Online Java Compiler - online ed
onlinegdb.com/online_java_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
14K
About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy
© 2016 - 2022 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Language Java
Main.java
25 }
26
27 System.out.println("\nProcess No.\tProcess Size\tBlock no.");
input
Enter your Choice
1
First fit
Process No. Process Size Block no.
1 212 2
2 417 5
3 112 2
4 426 Not Allocated
...Program finished with exit code 0
Press ENTER to exit console.


Online Java Compiler - online ed
onlinegdb.com/online_java_compiler
OnlineGDB beta
online compiler and debugger for c/c++
code. compile. run. debug. share.
IDE
My Projects
Classroom new
Learn Programming
Programming Questions
Sign Up
Login
14K
About • FAQ • Blog • Terms of Use •
Contact Us • GDB Tutorial • Credits •
Privacy
© 2016 - 2022 GDB Online
Run
Debug
Stop
Share
Save
{ } Beautify
Language Java
Main.java
25 }
26
27 System.out.println("\nProcess No.\tProcess Size\tBlock no.");
input
Enter your Choice
2
Best fit
Process No. Process Size Block no.
1 212 4
2 417 2
3 112 3
4 426 5
...Program finished with exit code 0
Press ENTER to exit console.

Online Java Compiler - online ed

onlinegdb.com/online_java_compiler

YouTube Maps Gmail News Translate Library manageme... WhatsApp Login System Using... Assignment 1_ Micr... Real time Indian Sig...

OnlineGDB beta

online compiler and debugger for c/c++

code. compile. run. debug. share.

IDE

My Projects

Classroom new

Learn Programming

Programming Questions

Sign Up

Login

14K

About • FAQ • Blog • Terms of Use • Contact Us • GDB Tutorial • Credits • Privacy

© 2016 - 2022 GDB Online

Run Debug Stop Share Save { } Beautify Language Java

Main.java

```
        }

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

input

```
Enter your Choice
3
Worst fit

Process No.     Process Size    Block no.
   1            212             5
   2            417             2
   3            112             5
   4            426             Not Allocated


...Program finished with exit code 0
Press ENTER to exit console.
```

21°C Clear

ENG US

19:33 01-11-2022