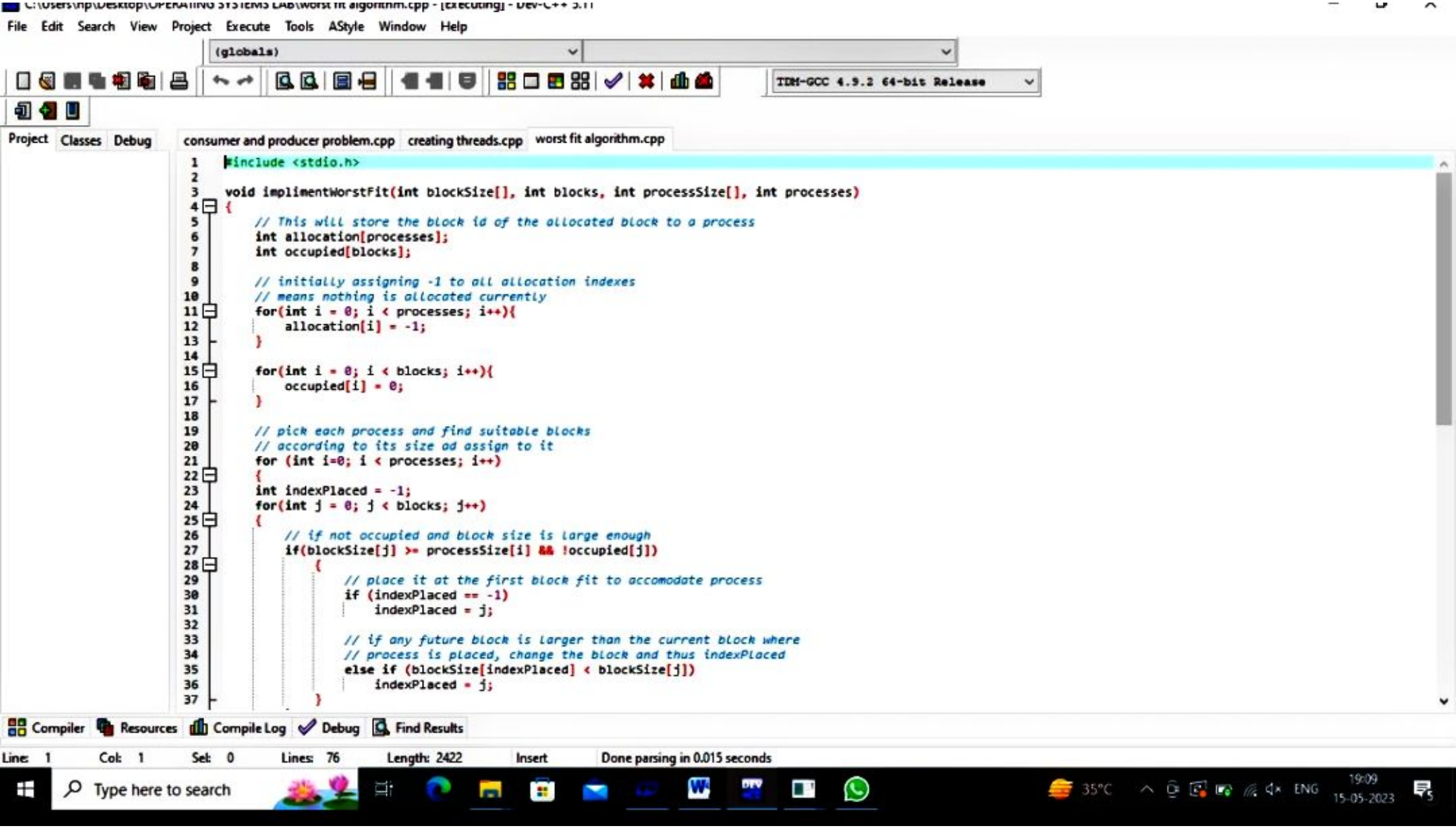
20. Write a C program to implement the worst-fit algorithm and allocate the memory block to each process.

Test Case:

Memory partitions: 300 KB, 600 KB, 350 KB, 200 KB, 750 KB, and 125 KB (in order),

Show the outcome for the test case with the worst-fit algorithms to place processes of size 115 KB, 500 KB, 358 KB, 200 KB, and 375 KB (in order)


C:\Users\hp\Desktop\OPERATING SYSTEMS LAB\worst fit algorithm.cpp - [Executing] - Dev-C++ 5.11
File Edit Search View Project Execute Tools AStyle Window Help
(globals)
TDM-GCC 4.9.2 64-bit Release
Project Classes Debug
consumer and producer problem.cpp
creating threads.cpp
worst fit algorithm.cpp
#include <stdio.h>
void implimentWorstFit(int blockSize[], int blocks, int processSize[], int processes)
{
// This will store the block id of the allocated block to a process
int allocation[processes];
int occupied[blocks];
// initially assigning -1 to all allocation indexes
// means nothing is allocated currently
for(int i = 0; i < processes; i++){
allocation[i] = -1;
}
for(int i = 0; i < blocks; i++){
occupied[i] = 0;
}
// pick each process and find suitable blocks
// according to its size ad assign to it
for (int i=0; i < processes; i++)
{
int indexPlaced = -1;
for(int j = 0; j < blocks; j++)
{
// if not occupied and block size is large enough
if(blockSize[j] >= processSize[i] && !occupied[j])
{
// place it at the first block fit to accomodate process
if (indexPlaced == -1)
indexPlaced = j;
// if any future block is larger than the current block where
// process is placed, change the block and thus indexPlaced
else if (blockSize[indexPlaced] < blockSize[j])
indexPlaced = j;
}
Compiler Resources Compile Log Debug Find Results
Line: 1 Col: 1 Sel: 0 Lines: 76 Length: 2422 Insert Done parsing in 0.015 seconds
Type here to search
35°C
ENG
19:09
15-05-2023

C:\Users\hp\Desktop\OPERATING SYSTEMS LAB\worst fit algorithm.cpp - [Executing] - Dev-C++ 5.11

e Edit Search View Project Execute Tools AStyle Window Help

(globals)

TDM-GCC 4.9.2 64-bit Release

oject Classes Debug

consumer and producer problem.cpp | creating threads.cpp | worst fit algorithm.cpp

```
40          // If we were successfully able to find block for the process
41          if (indexPlaced != -1)
42          {
43              // allocate this block j to process p[i]
44              allocation[i] = indexPlaced;
45
46              // make the status of the block as occupied
47              occupied[indexPlaced] = 1;
48
49              // Reduce available memory for the block
50              blockSize[indexPlaced] -= processSize[i];
51          }
52      }
53
54      printf("\nProcess No.\tProcess Size\tBlock no.\n");
55      for (int i = 0; i < processes; i++)
56      {
57          printf("%d \t\t\t %d \t\t\t", i+1, processSize[i]);
58          if (allocation[i] != -1)
59              printf("%d\n",allocation[i] + 1);
60          else
61              printf("Not Allocated\n");
62      }
63  }
64
65  // Driver code
66  int main()
67  {
68      int blockSize[] = {300, 600, 350, 200, 750,125};
69      int processSize[] = {115, 300, 350, 200,375};
70      int blocks = sizeof(blockSize)/sizeof(blockSize[0]);
71      int processes = sizeof(processSize)/sizeof(processSize[0]);
72
73      implimentWorstFit(blockSize, blocks, processSize, processes);
74
75      return 0;
76  }
```

Compiler | Resources | Compile Log | Debug | Find Results

e: 1 Col: 1 Sel: 0 Lines: 76 Length: 2422 Insert Done parsing in 0.015 seconds

Type here to search

35°C ENG 19:09 15-05-2023

```
Process No.     Process Size    Block no.
1                       115                     5
2                       300                     2
3                       350                     3
4                       200                     1
5                       375                     Not Allocated

--------------------------------
Process exited after 0.169 seconds with return value 0
Press any key to continue . . .
```