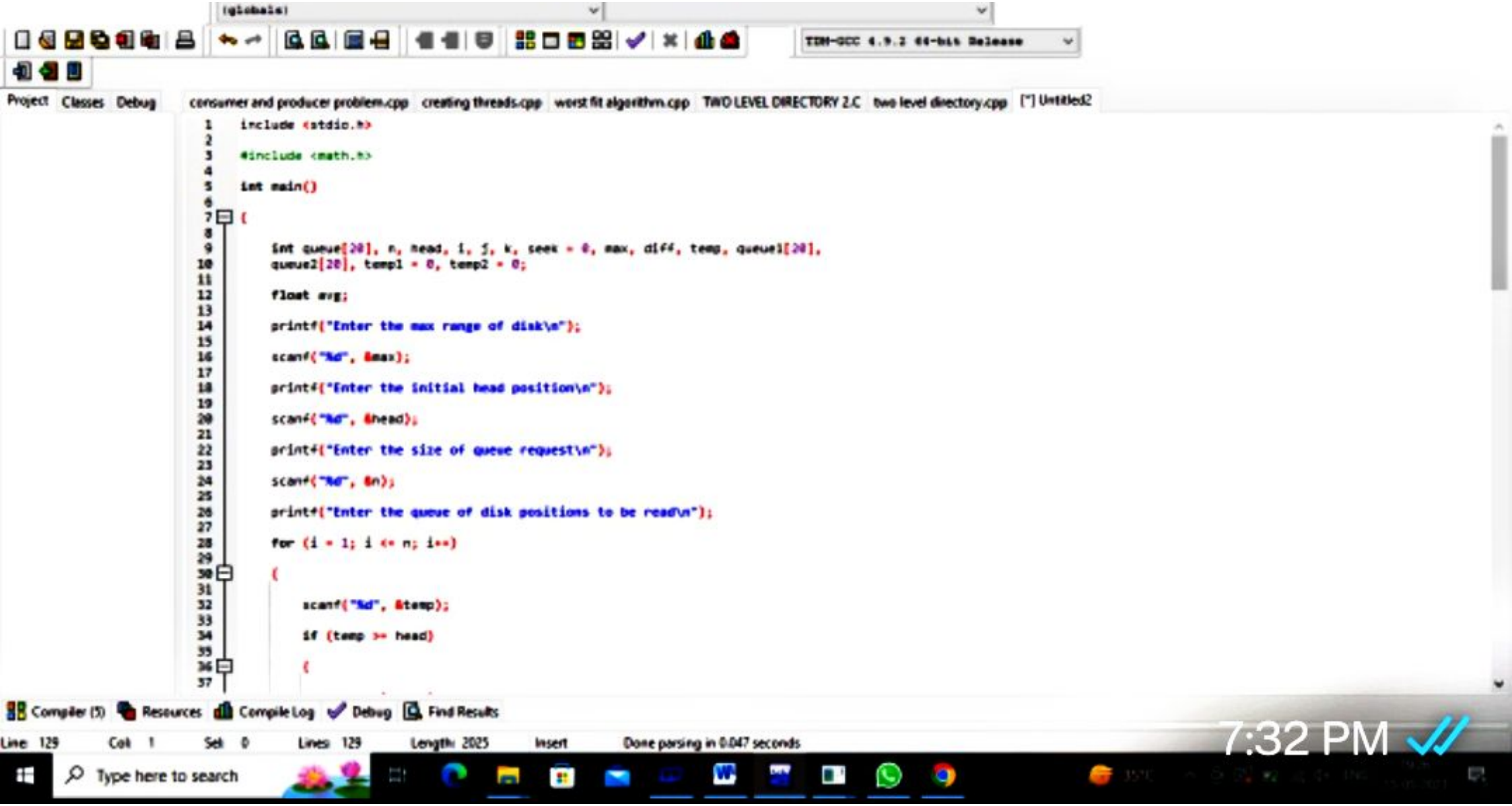
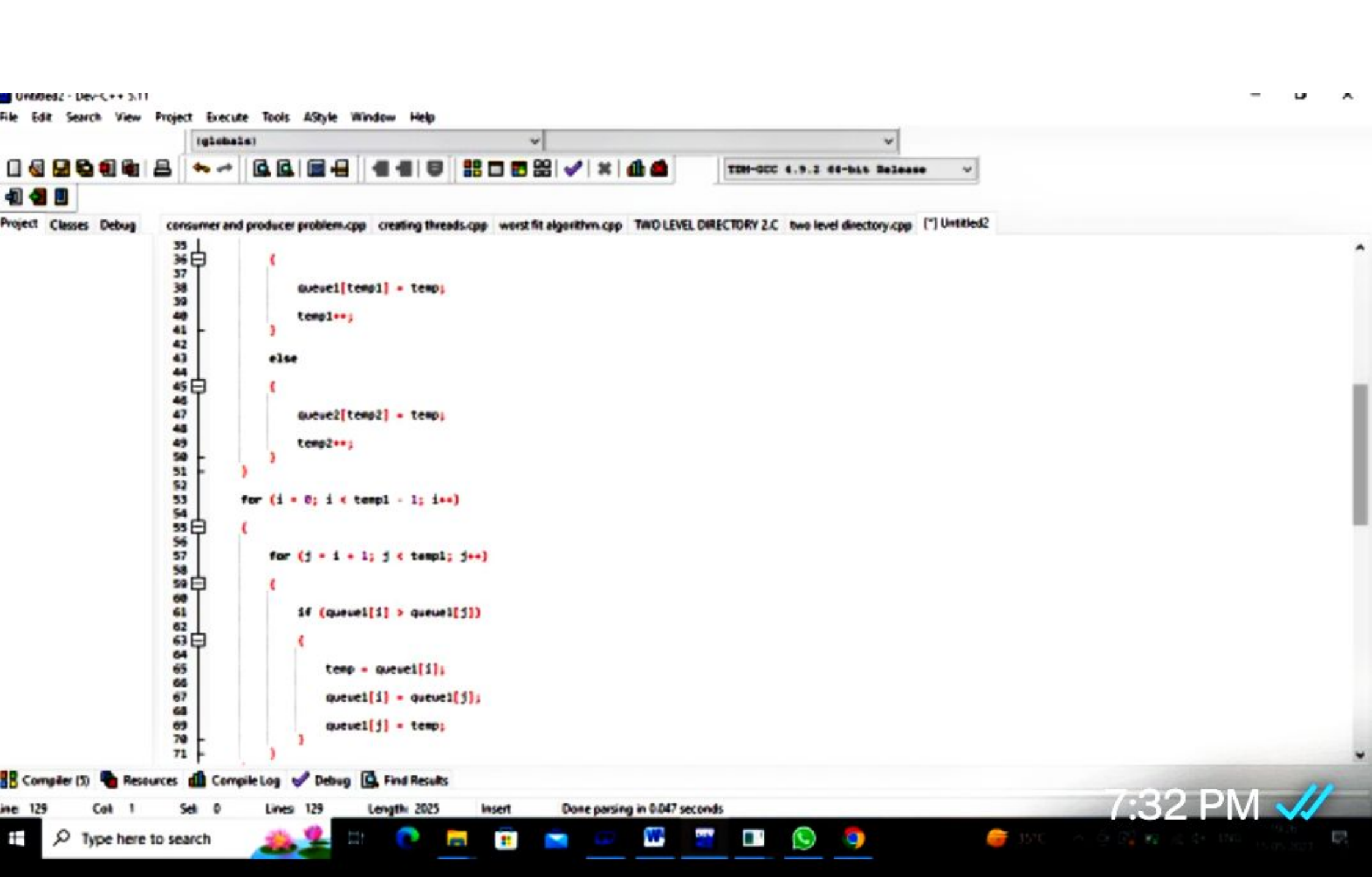
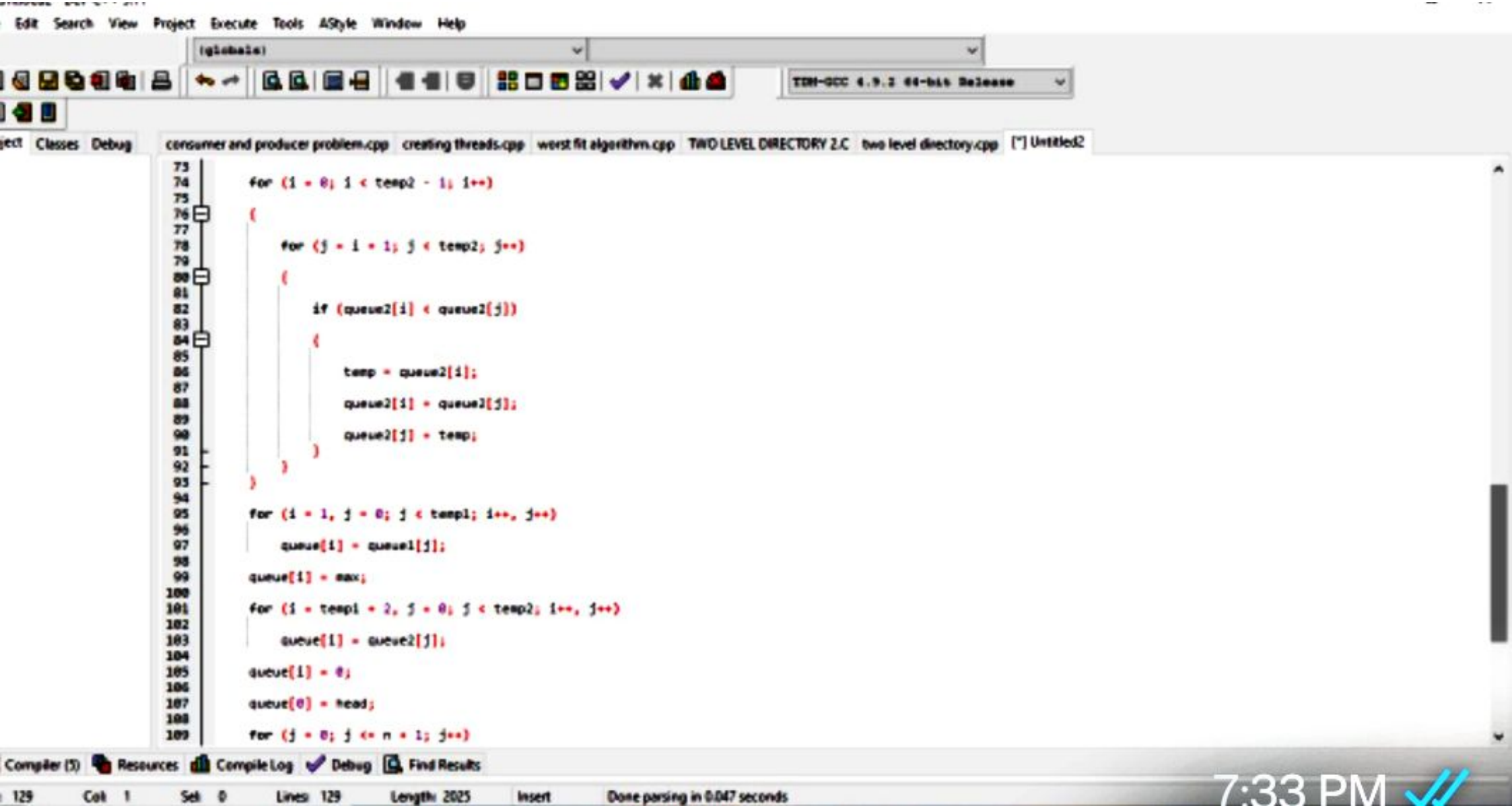
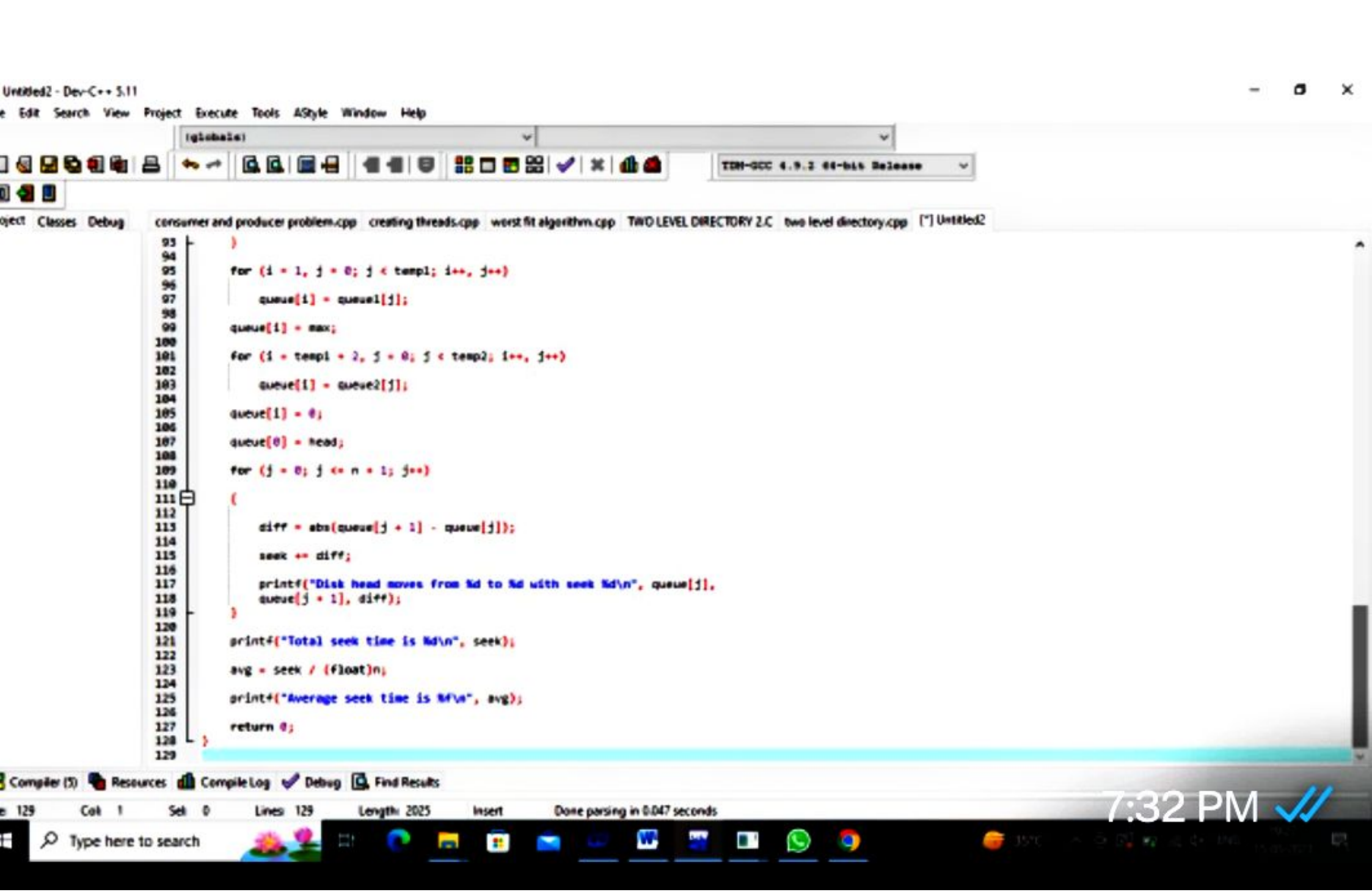
23. Write a C program to simulate SCAN disk scheduling algorithms. and execute your program and find out and print the average head movement for the following test case.

No of tracks:5; Track position:55 58 60 70 18


(globals)
TDM-GCC 4.9.2 64-bit Release
Project
Classes
Debug
consumer and producer problem.cpp
creating threads.cpp
worst fit algorithm.cpp
TWO LEVEL DIRECTORY 2.C
two level directory.cpp
[*] Untitled2
#include <stdio.h>
#include <math.h>
int main()
{
int queue[20], n, head, i, j, k, seek = 0, max, diff, temp, queue1[20],
queue2[20], temp1 = 0, temp2 = 0;
float avg;
printf("Enter the max range of disk\n");
scanf("%d", &max);
printf("Enter the initial head position\n");
scanf("%d", &head);
printf("Enter the size of queue request\n");
scanf("%d", &n);
printf("Enter the queue of disk positions to be read\n");
for (i = 1; i <= n; i++)
{
scanf("%d", &temp);
if (temp >= head)
{
Compiler (5)
Resources
Compile Log
Debug
Find Results
Line: 129
Col: 1
Sel: 0
Lines: 129
Length: 2025
Insert
Done parsing in 0.047 seconds
7:32 PM
Type here to search


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
TWO LEVEL DIRECTORY 2.C
two level directory.cpp
[*] Untitled2
{
queue1[temp1] = temp;
temp1++;
}
else
{
queue2[temp2] = temp;
temp2++;
}
}
for (i = 0; i < temp1 - 1; i++)
{
for (j = i + 1; j < temp1; j++)
{
if (queue1[i] > queue1[j])
{
temp = queue1[i];
queue1[i] = queue1[j];
queue1[j] = temp;
}
}
Compiler (5)
Resources
Compile Log
Debug
Find Results
Line: 129 Col: 1 Sel: 0 Lines: 129 Length: 2025 Insert Done parsing in 0.047 seconds
Type here to search
7:32 PM


Edit Search View Project Execute Tools AStyle Window Help
(globals)
TDM-GCC 4.9.2 64-bit Release
Classes Debug
consumer and producer problem.cpp
creating threads.cpp
worst fit algorithm.cpp
TWO LEVEL DIRECTORY 2.C
two level directory.cpp
[*] Untitled2
for (i = 0; i < temp2 - 1; i++)
{
for (j = i + 1; j < temp2; j++)
{
if (queue2[i] < queue2[j])
{
temp = queue2[i];
queue2[i] = queue2[j];
queue2[j] = temp;
}
}
}
for (i = 1, j = 0; j < temp1; i++, j++)
queue[i] = queue1[j];
queue[i] = max;
for (i = temp1 + 2, j = 0; j < temp2; i++, j++)
queue[i] = queue2[j];
queue[1] = 0;
queue[0] = head;
for (j = 0; j <= n + 1; j++)
Compiler (5)
Resources
Compile Log
Debug
Find Results
129 Col: 1 Sel: 0 Lines: 129 Length: 2025 Insert Done parsing in 0.047 seconds
7:33 PM


Untitled2 - 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
TWO LEVEL DIRECTORY 2.C
two level directory.cpp
[*] Untitled2
}
for (i = 1, j = 0; j < temp1; i++, j++)
queue[i] = queue1[j];
queue[i] = max;
for (i = temp1 + 2, j = 0; j < temp2; i++, j++)
queue[i] = queue2[j];
queue[i] = 0;
queue[0] = head;
for (j = 0; j <= n + 1; j++)
{
diff = abs(queue[j + 1] - queue[j]);
seek += diff;
printf("Disk head moves from %d to %d with seek %d\n", queue[j],
queue[j + 1], diff);
}
printf("Total seek time is %d\n", seek);
avg = seek / (float)n;
printf("Average seek time is %f\n", avg);
return 0;
}
Compiler (5)
Resources
Compile Log
Debug
Find Results
Ln 129 Col 1 Sel 0 Lines 129 Length 2025 Insert
Done parsing in 0.047 seconds
Type here to search
7:32 PM

```
Enter the max range of disk
200
Enter the initial head position
100
Enter the size of queue request
5
Enter the queue of disk positions to be read
55
58
60
70
18
Disk head moves from 100 to 10 with seek 90
Disk head moves from 10 to 70 with seek 60
Disk head moves from 70 to 60 with seek 10
Disk head moves from 60 to 58 with seek 2
Disk head moves from 58 to 55 with seek 3
Disk head moves from 55 to 18 with seek 37
Disk head moves from 18 to 0 with seek 18
Total seek time is 220
Average seek time is 44.000000

--------------------------------
Process exited after 19.2 seconds with return value 0
Press any key to continue . . .
```