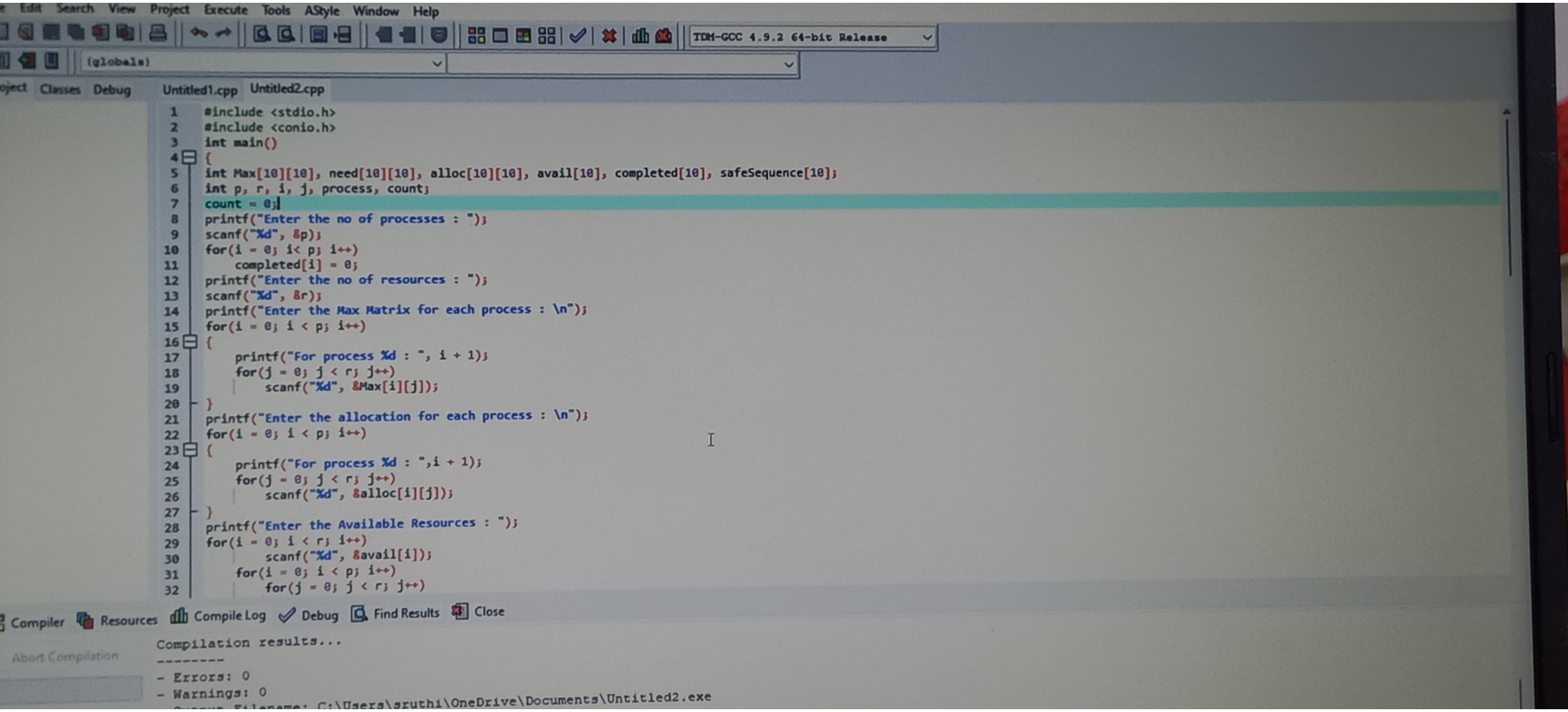

Edit Search View Project Execute Tools AStyle Window Help
TDM-GCC 4.9.2 64-bit Release
(globals)
Classes Debug
Untitled1.cpp
Untitled2.cpp
#include <stdio.h>
#include <conio.h>
int main()
{
int Max[10][10], need[10][10], alloc[10][10], avail[10], completed[10], safeSequence[10];
int p, r, i, j, process, count;
count = 0;
printf("Enter the no of processes : ");
scanf("%d", &p);
for(i = 0; i< p; i++)
completed[i] = 0;
printf("Enter the no of resources : ");
scanf("%d", &r);
printf("Enter the Max Matrix for each process : \n");
for(i = 0; i < p; i++)
{
printf("For process %d : ", i + 1);
for(j = 0; j < r; j++)
scanf("%d", &Max[i][j]);
}
printf("Enter the allocation for each process : \n");
for(i = 0; i < p; i++)
{
printf("For process %d : ",i + 1);
for(j = 0; j < r; j++)
scanf("%d", &alloc[i][j]);
}
printf("Enter the Available Resources : ");
for(i = 0; i < r; i++)
scanf("%d", &avail[i]);
for(i = 0; i < p; i++)
for(j = 0; j < r; j++)
Compiler Resources Compile Log Debug Find Results Close
Abort Compilation
Compilation results...

- Errors: 0
- Warnings: 0
C:\Users\gruthi\OneDrive\Documents\Untitled2.exe

Project | Classes | Debug

Untitled1.cpp | Untitled2.cpp

```
31      for(i = 0; i < p; i++)
32          for(j = 0; j < r; j++)
33              need[i][j] = Max[i][j] - alloc[i][j];
34  do
35  {
36      printf("\nMax matrix:\tAllocation matrix:\n");
37      for(i = 0; i < p; i++)
38      {
39          for( j = 0; j < r; j++)
40              printf("%d  ", Max[i][j]);
41          printf("\t\t");
42          for( j = 0; j < r; j++)
43              printf("%d  ", alloc[i][j]);
44          printf("\n");
45      }
46      process = -1;
47      for(i = 0; i < p; i++)
48      {
49          if(completed[i] == 0)
50          {
51              process = i ;
52              for(j = 0; j < r; j++)
53              {
54                  if(avail[j] < need[i][j])
55                  {
56                      process = -1;
57                      break;
58                  }
59              }
60          }
61          if(process != -1)
62              break;
```

Compiler | Resources | Compile Log | Debug | Find Results | Close

Abort Compilation

Shorten compiler paths

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\sruthi\OneDrive\Documents\Untitled2.exe
- Output Size: 131.1123046875 KiB
- Compilation Time: 0.44s
```

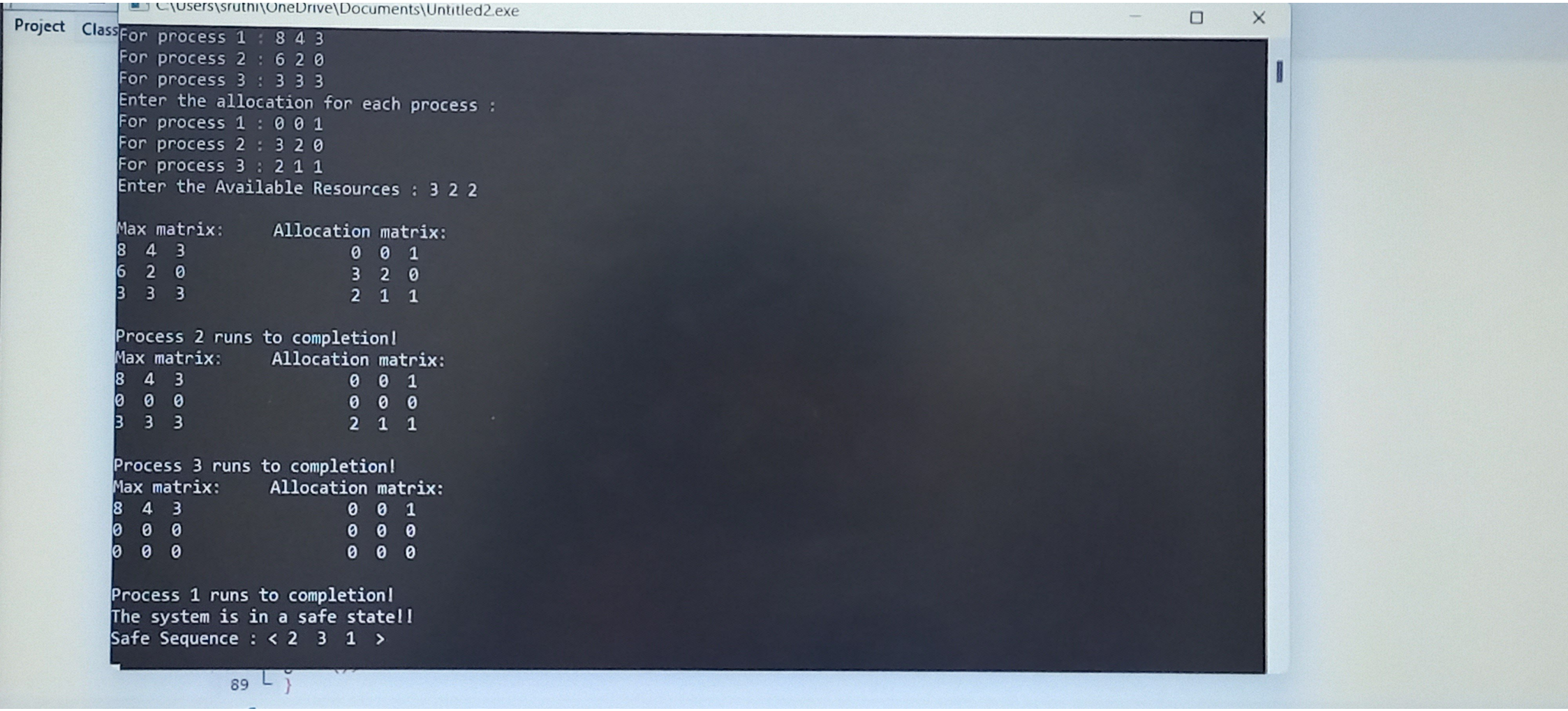
```
            }
        }
    }
    if(process != -1)
        break;
}
if(process != -1)
{
    printf("\nProcess %d runs to completion!", process + 1);
    safeSequence[count] = process + 1;
    count++;
    for(j = 0; j < r; j++)
    {
        avail[j] += alloc[process][j];
        alloc[process][j] = 0;
        Max[process][j] = 0;
        completed[process] = 1;
    }
}
}while(count != p && process != -1);
if(count == p)
{
    printf("\nThe system is in a safe state!!\n");
    printf("Safe Sequence : < ");
    for( i = 0; i < p; i++)
        printf("%d  ", safeSequence[i]);
    printf(">\n");
}
else
    printf("\nThe system is in an unsafe state!!");
getch();
}
```

Compiler | Resources | Compile Log | Debug | Find Results | Close

Abort Compilation

Shorten compiler paths

```
Compilation results...
--------
- Errors: 0
- Warnings: 0
- Output Filename: C:\Users\sruthi\OneDrive\Documents\Untitled2.exe
- Output Size: 131.1123046875 KiB
- Compilation Time: 0.44s
```


Project
Class
C:\Users\sruthi\OneDrive\Documents\Untitled2.exe
For process 1 : 8 4 3
For process 2 : 6 2 0
For process 3 : 3 3 3
Enter the allocation for each process :
For process 1 : 0 0 1
For process 2 : 3 2 0
For process 3 : 2 1 1
Enter the Available Resources : 3 2 2
Max matrix: Allocation matrix:
8 4 3 0 0 1
6 2 0 3 2 0
3 3 3 2 1 1
Process 2 runs to completion!
Max matrix: Allocation matrix:
8 4 3 0 0 1
0 0 0 0 0 0
3 3 3 2 1 1
Process 3 runs to completion!
Max matrix: Allocation matrix:
8 4 3 0 0 1
0 0 0 0 0 0
0 0 0 0 0 0
Process 1 runs to completion!
The system is in a safe state!!
Safe Sequence : < 2 3 1 >
89 }