

**CSE4001 - Parallel and Distributed Computing, Fall 2019**  
**Vellore Institute of Technology**  
**Instructor: Prof Deebak B D - SCOPE**

**Lab report**

**Title of Lab: Linear search using section clause in OpenMP**

**Assessment #: 7**

**Date: 09|09|2019**

**Author's name: Gagan Deep Singh**

**Registration ID: 17BCI0140**

**Lab section: Friday L59 + L60**

**AIM:** Write a simple OpenMP program to demonstrate Linear Search using Section Clause.

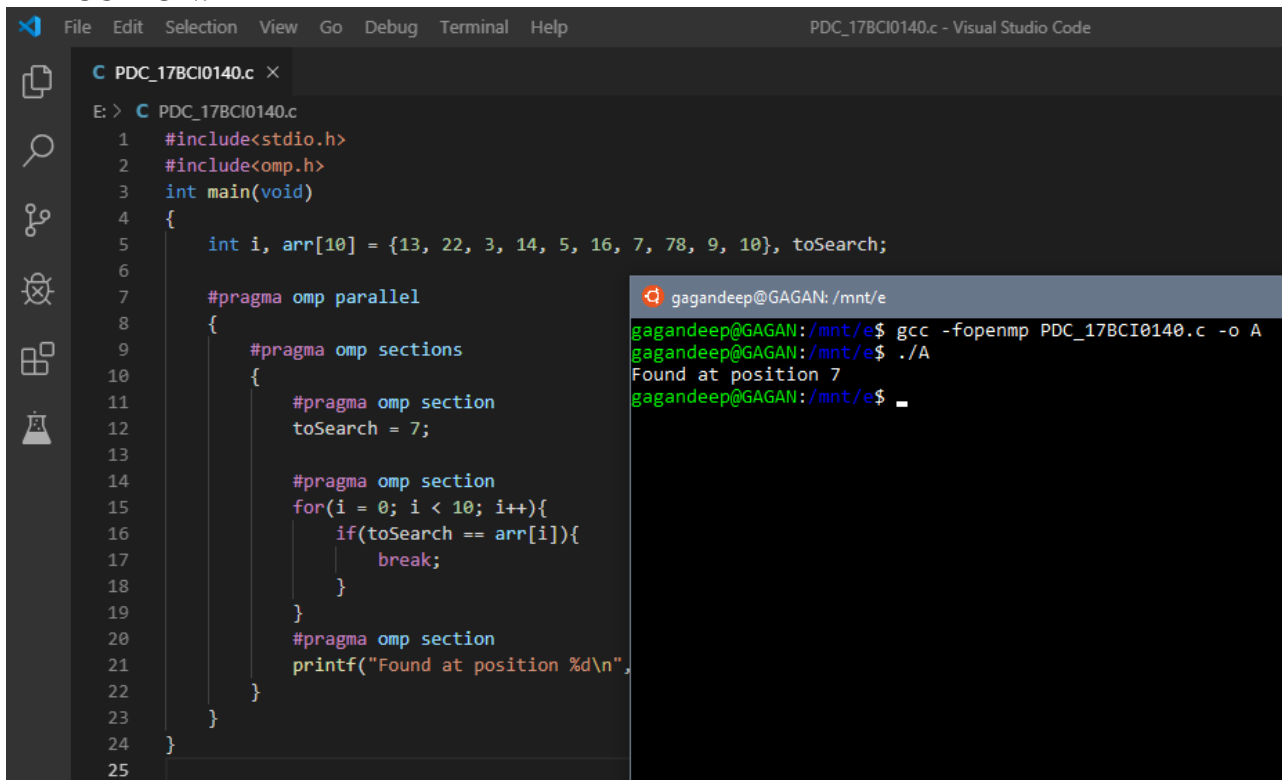
**SOURCE CODE:**

```
#include<stdio.h>
#include<omp.h>
int main(void)
{
    int i, arr[10] = {13, 22, 3, 14, 5, 16, 7, 78, 9, 10}, toSearch;

    #pragma omp parallel
    {
        #pragma omp sections
        {
            #pragma omp section
            toSearch = 7;

            #pragma omp section
            for(i = 0; i < 10; i++){
                if(toSearch == arr[i]){
                    break;
                }
            }
            #pragma omp section
            printf("Found at position %d\n", i+1);
        }
    }
}
```

## EXECUTION:



```
File Edit Selection View Go Debug Terminal Help PDC_17BCI0140.c - Visual Studio Code

C PDC_17BCI0140.c x
E: > C PDC_17BCI0140.c
1  #include<stdio.h>
2  #include<omp.h>
3  int main(void)
4  {
5      int i, arr[10] = {13, 22, 3, 14, 5, 16, 7, 78, 9, 10}, toSearch;
6
7      #pragma omp parallel
8      {
9          #pragma omp sections
10         {
11             #pragma omp section
12             toSearch = 7;
13
14             #pragma omp section
15             for(i = 0; i < 10; i++){
16                 if(toSearch == arr[i]){
17                     break;
18                 }
19             }
20             #pragma omp section
21             printf("Found at position %d\n",
22
23         }
24     }
25 }
```

```
gagandeep@GAGAN: /mnt/e
gagandeep@GAGAN:/mnt/e$ gcc -fopenmp PDC_17BCI0140.c -o A
gagandeep@GAGAN:/mnt/e$ ./A
Found at position 7
gagandeep@GAGAN:/mnt/e$ _
```

**RESULTS:** The omp sections directive distributes work among threads bound to a defined parallel region.