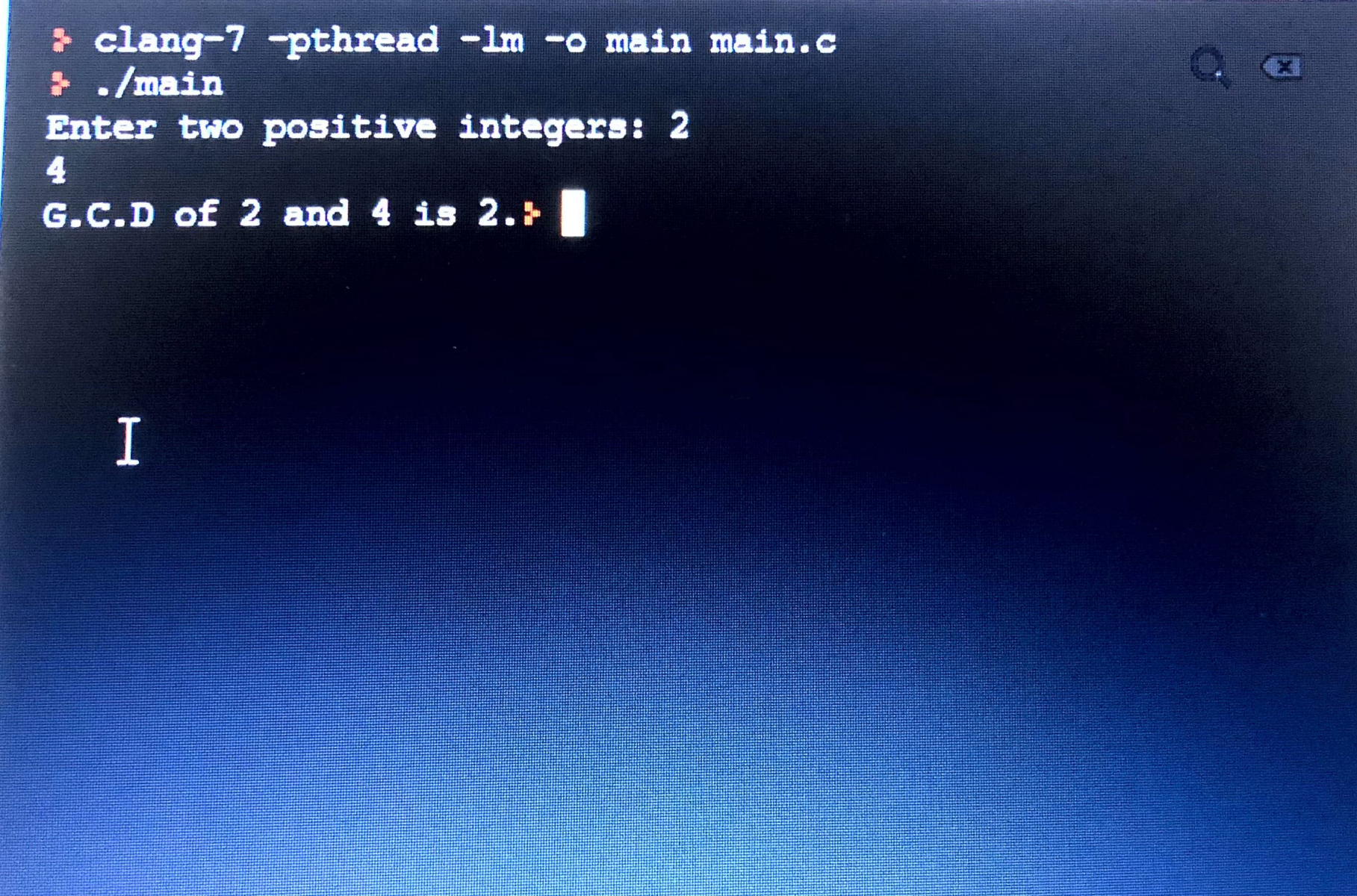
main.c

```c
#include <stdio.h>
int fact (int n)
{
  if(n==0)
  return 1;
  return(n*fact(n-1));
}
int main()
{
  int n;
  printf("enter the value of n\n");
  scanf("%d",&n);
  printf("the factorial of %d=%d\n",n,fact(n));
  return 0;
}
```

```
clang-7 -pthread -lm -o main main.c
./main
enter the value of n
4
the factorial of 4=24
```

main.c

```c
#include <stdio.h>
int hcf(int n1, int n2);
int main() {
    int n1, n2;
    printf("Enter two positive integers: ");
    scanf("%d %d", &n1, &n2);
    printf("G.C.D of %d and %d is %d.", n1, n2, hcf(n1, n2)
    );
    return 0;
}

int hcf(int n1, int n2) {
    if (n2 != 0)
        return hcf(n2, n1 % n2);
    else
        return n1;
}

```


clang-7 -pthread -lm -o main main.c
./main
Enter two positive integers: 2
4
G.C.D of 2 and 4 is 2.

main.c

```c
#include <stdio.h>
int fib(int n)
{
  if(n==0)
  return 0;
  if(n==1)
  return 1;
  return fib(n-1)+fib(n-2);
}
int main()
{
  int i,n;
  printf("Enter the value of n\n");
  scanf("%d",&n);
  printf("%d fibonacci numbers are \n",n);
  for(i=0;i<n;i++)
  {
    printf("fib(%d)=%d\n",i,fib(i));
  }
  return 0;
}
```

```
$ clang-7 -pthread -lm -o main main.c
$ ./main
Enter the value of n
12
12 fibonacci numbers are
fib(0)=0
fib(1)=1
fib(2)=1
fib(3)=2
fib(4)=3
fib(5)=5
fib(6)=8
fib(7)=13
fib(8)=21
fib(9)=34
fib(10)=55
fib(11)=89
$
```

main.c

```c
#include <stdio.h>

void towers(int, char, char, char);

int main()
{
    int num;

    printf("Enter the number of disks : ");
    scanf("%d", &num);
    printf("The sequence of moves involved in the Tower of
    Hanoi are :\n");
    towers(num, 'A', 'C', 'B');
    return 0;
}
void towers(int num, char frompeg, char topeg, char auxpeg)
{
    if (num == 1)
    {
        printf("\n Move disk 1 from peg %c to peg %c",
        frompeg, topeg);
        return;
    }
    towers(num - 1, frompeg, auxpeg, topeg);
    printf("\n Move disk %d from peg %c to peg %c", num,
    frompeg, topeg);
    towers(num - 1, auxpeg, topeg, frompeg);
}
```

```
clang-7 -pthread -lm -o main main.c
./main
Enter the number of disks : 3
The sequence of moves involved in the Tower of Hanoi are :

 Move disk 1 from peg A to peg C
 Move disk 2 from peg A to peg B
 Move disk 1 from peg C to peg B
 Move disk 3 from peg A to peg C
 Move disk 1 from peg B to peg A
 Move disk 2 from peg B to peg C
 Move disk 1 from peg A to peg C
```

main.c

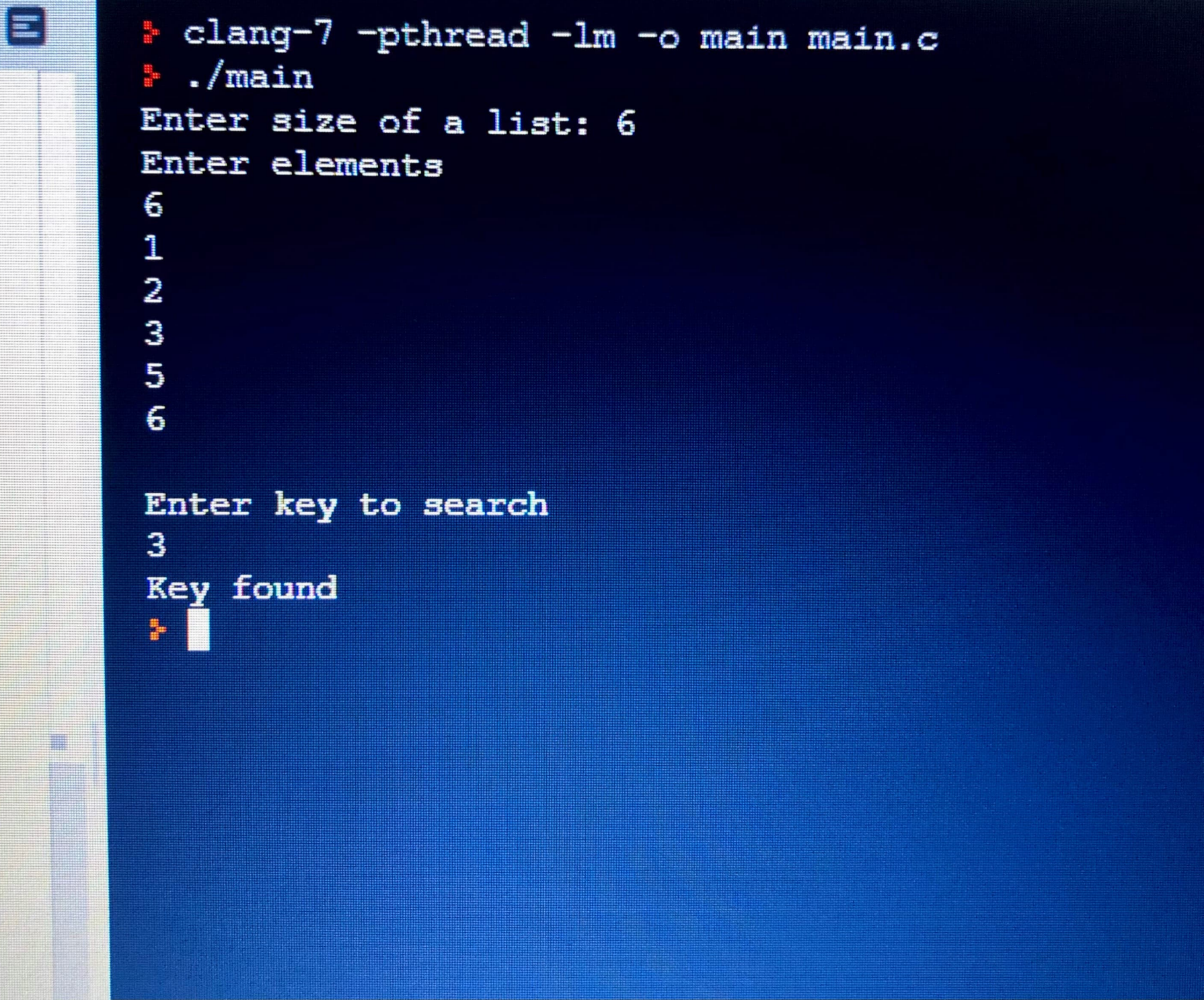
```c
#include <stdio.h>

void binary_search(int [], int, int, int);
void bubble_sort(int [], int);

int main()
{
    int key, size, i;
    int list[25];

    printf("Enter size of a list: ");
    scanf("%d", &size);
    printf("Enter elements\n");
    for(i = 0; i < size; i++)
    {
        scanf("%d",&list[i]);
    }
    bubble_sort(list, size);
    printf("\n");
    printf("Enter key to search\n");
    scanf("%d", &key);
    binary_search(list, 0, size, key);

}

void bubble_sort(int list[], int size)
{
    int temp, i, j;
    for (i = 0; i < size; i++)
```

main.c

```
27  {
28      int temp, i, j;
29      for (i = 0; i < size; i++)
30      {
31          for (j = i; j < size; j++)
32          {
33              if (list[i] > list[j])
34              {
35                  temp = list[i];
36                  list[i] = list[j];
37                  list[j] = temp;
38              }
39          }
40      }
41  }
42
43  void binary_search(int list[], int lo, int hi, int key)
44  {
45      int mid;
46
47      if (lo > hi)
48      {
49          printf("Key not found\n");
50          return;
51      }
52      mid = (lo + hi) / 2;
53      if (list[mid] == key)
54      {
55          printf("Key found\n");
```

main.c

```
46
47      if (lo > hi)
48      {
49          printf("Key not found\n");
50          return;
51      }
52      mid = (lo + hi) / 2;
53      if (list[mid] == key)
54      {
55          printf("Key found\n");
56      }
57      else if (list[mid] > key)
58      {
59          binary_search(list, lo, mid - 1, key);
60      }
61      else if (list[mid] < key)
62      {
63          binary_search(list, mid + 1, hi, key);
64      }
65  }
66
```


clang-7 -pthread -lm -o main main.c
./main
Enter size of a list: 6
Enter elements
6
1
2
3
5
6
Enter key to search
3
Key found