

```

1  #include "apue.h"
2  #include <pthread.h>
3
4  int coord[2]={-1,-1};
5  int key, righe, colonne;
6  pthread_mutex_t coord_mutex;
7  int **matrice;
8  pthread_t *threads;
9  void crea_matrice();
10 void stampa_matrice();
11
12 void *ricerca(void *tid)
13 {
14     int k,j;
15     pthread_t *mytid;
16
17     mytid = (pthread_t *) tid;
18
19     printf ("Thread %d doing search %d \n",*mytid, *mytid);
20
21     for (j=0; j < colonne; j++) {
22         if (matrice[*mytid][j] == key){
23             pthread_mutex_lock(&coord_mutex);
24             coord[0]=*mytid;
25             coord[1]=j;
26
27             for (k=0;k<righe;k++){
28                 if(k!=*mytid){
29                     printf("%d %d\n",k,*mytid);
30                     pthread_cancel(threads[k]);
31                 }
32             }
33
34             pthread_mutex_unlock(&coord_mutex);
35             pthread_exit(NULL);
36         }
37     }
38
39     pthread_exit(NULL);
40 }
41
42 int main(int argc, char**argv){
43     pthread_t *tids, i;
44     righe = atoi(argv[1]);
45     colonne = atoi(argv[2]);
46     crea_matrice(righe,colonne);
47     stampa_matrice();
48     printf("Elemento da ricercare: ");
49     scanf("%d",&key);
50     threads = calloc(righe,sizeof(pthread_t));
51     tids = calloc(righe, sizeof(pthread_t));
52
53     for (i=0; i<righe; i++) {
54         tids[i] = i;
55         pthread_create(&threads[i], NULL, ricerca, (void *) &tids[i]);
56     }
57
58     for (i=0; i<righe; i++) {
59         pthread_join(threads[i], NULL);
60     }
61
62     printf ("Fatto. Coordinate = %d %d\n", coord[0],coord[1]);
63     exit(0);
64 }
65
66 void crea_matrice(){
67     int i,j;
68     matrice = calloc(righe,sizeof(int *));
69

```

```
70     for(i=0;i<righe;i++)
71     matrice[i] = calloc(righe,sizeof(int));
72
73     for(i=0;i<righe;i++)
74     for(j=0;j<colonne;j++)
75         matrice[i][j] = rand() % 20;
76 }
77
78 void stampa_matrice(){
79     int i,j;
80
81     for(i=0;i<righe;i++){
82     for(j=0;j<colonne;j++)
83         printf("%d\t",matrice[i][j]);
84
85     printf("\n");
86     }
87 }
88
```