```c
#include <stdio.h>
#include <string.h>

#define N 2
#define M 2

#define L 40

typedef struct date{
	int year;
	int month;
	int day;
	}Date;

typedef struct student
{
	char name[L];  //char *name;
	int AM;
	Date registration;
	}
 STUDENT;

//int tmima[N][M];



void input(STUDENT foitites[][M], int rows);

void output(STUDENT foitites[][M], int rows);

// void search(STUDENT foitites[][M], int rows);

STUDENT   find(STUDENT foitites[][M], int rows, int number);


int main()
{

	STUDENT tmima[N][M];

	int num;

	STUDENT x;

	input(tmima, N);

	output(tmima, N);

//	search(tmima, N);

	printf("Give AM to search: ");
	scanf("%d", &num);

//	x=find(tmima, N, num);
```

```
	printf("Name: %s\n", x.name);
	printf("AM: %d\n", x.AM);
	printf("Registration date: %d\%d\%d\n", x.registration.day,
x.registration.month, x.registration.year);

	return 0;


}


void input(STUDENT foitites[][M], int rows)
{
	int i,j;

	for (i=0; i<rows; i++)
	   {
	       for (j=0; j<M; j++)
	         {
	              printf("Give the (%d, %d)-student:", i, j);
	              scanf("%s",foitites[i][j].name);
	              scanf("%d", &foitites[i][j].AM);
	scanf("%d%d%d", &foitites[i][j].registration.day,
&foitites[i][j].registration.month, &foitites[i][j].registration.year);

	         }
	   }
}

void output(STUDENT foitites[][M], int rows)
{
	int i,j;

	for (i=0; i<rows; i++)
	   {
	       for (j=0; j<M; j++)
	         {
	              printf("The (%d, %d)-student is:", i, j);
	              printf("name: %s ",foitites[i][j].name);
	              printf("AM: %d  ", foitites[i][j].AM);
	printf("Registration %d\\%d\\%d  \n ", foitites[i][j].registration.day,
foitites[i][j].registration.month, foitites[i][j].registration.year);

	         }
	        printf("\n");
	   }
}

STUDENT   find(STUDENT foitites[][M], int rows, int number)
{

	int i,j;
	STUDENT result;
	/*
	char onoma[L];
```

```
        scanf("%s", onoma); */

        for (i=0; i<rows; i++)
           {
               for (j=0; j<M; j++)
                 {
        /* if (strcmp(onoma, foitites[i][j].name)==0) { result ... , return result;}
        */
                              if (number==foitites[i][j].AM) {
                        result=foitites[i][j];
                        return result;
                 }
           }
}
```