

```

/* Πρόγραμμα χειρισμού στοιχείων βαθμολογίας φοιτητών με χρήση αρχείων
    APXEIA KEIMENOU
*/

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

#define N 5
#define M 31

typedef struct student /* Δομή φοιτητής */
{
    char eponymo[M];
    char name[M];
    float degree;
} STUDENT;

void input(STUDENT x[], int size); /* Εισαγωγή φοιτητών */
void output(STUDENT x[], int size); /* Εκτύπωση φοιτητών */
void statistics(STUDENT x[], int size); /* Στατιστικά στοιχείων φοιτητών */

int main()
{
    // int x[N];

    FILE * fp;

    int i;

    char onoma[M]; /* Όνομα για αναζήτηση */
    STUDENT etos[N]; /* Πίνακας φοιτητών */
    input(etos, N);

    /* fp=fopen("example.txt", "r");
    for (i=0; i<N; i++)
        fscanf(fp, "%s%s%f", etos[i].eponymo, etos[i].name, &etos[i].degree);

    fclose(fp);

    output(etos, N);

    statistics(etos, N); */

    fp=fopen("example.txt", "w");
    for (i=0; i<N; i++)

```

```
        fprintf(fp, "%s %s %f\n", etos[i].eponymo, etos[i].name,
etos[i].degree);
```

```
    fclose(fp);
```

```
    return 0;
```

```
}
```

```
void input(STUDENT x[], int size)
```

```
{
```

```
    int i;
```

```
    for (i=0; i<size; i++)
```

```
    {
```

```
        printf("Give the %d-th student: ", i);
```

```
        scanf("%s", x[i].eponymo);
```

```
        scanf("%s", x[i].name);
```

```
        scanf("%f", &x[i].degree);
```

```
    }
```

```
}
```

```
void output(STUDENT x[], int size)
```

```
{
```

```
    int i;
```

```
    printf("\n The students are: \n");
```

```
    for (i=0; i<size; i++)
```

```
    {
```

```
        printf("The %d-th student is: ", i);
```

```
        printf("%s ", x[i].eponymo);
```

```
        printf(" %s ", x[i].name);
```

```
        printf("%.2f\n", x[i].degree);
```

```
    }
```

```
}
```

```
void statistics(STUDENT x[], int size)
```

```
{
```

```
    float min, max, average;
```

```
    int i;
```

```
    min=x[0].degree;
```

```
    max=x[0].degree;
```

```
    average=x[0].degree;
```

```
    for (i=1; i<size; i++)
```

```
    {
```

```
        average=average+x[i].degree;
```

```
        if (x[i].degree<min) min=x[i].degree;
```

```
        if (x[i].degree>max) max=x[i].degree;
```

```
    }
```

```
        printf("\nThe values are (min,max,average): %.2f, %.2f. %.2f\n", min, max,  
average/N);  
    }
```