

```

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

#define N 5
#define M 30

/*      Υλοποίηση δυαδικού ψαξίματος σε πίνακα συμβολοσειρών, που έχει
υλοποιηθεί σαν πίνακας δεικτών σε χαρακτήρες.
Επιστρέφει θέση στοιχείου αλλιώς -1.
Ο πίνακας συμβολοσειρών υλοποιείται με δυναμική διαχείριση μνήμης
(malloc).
*/

int binarySearch(char * arr[], int low, int high, char * key);

int main() {
    int i, len, result;
    char temp[M];

    char *arr[N]={0};

    /* Εισαγωγή συμβολοσειρών */

    printf("Enter the strings:\n");
    for ( i = 0; i < N; i++) {
        /* compute Len */
        /* printf("Give Length of word %d: ", i);
scanf("%d", &len);*/
        scanf("%s", temp);
        len=strlen(temp);
        arr[i]=(char *)malloc((len+1)*sizeof(char));
        strcpy(arr[i],temp);
        // arr[i]=temp; ΛΑΘΟΣ
    }

    printf("Give element: ");
    scanf("%s", temp);

    result=binarySearch(arr, 0, N-1, temp);

    if (result==-1) printf("The element does not exist");
    else printf("The element exists is position %d\n", result);

    for ( i = 0; i < N; i++) free(arr[i]); /* Ελευθέρωση pointers */

    return 0;
}

int binarySearch(char * arr[], int low, int high, char * key) {

```

```
int left, right;

left=low;
right = high - 1;

while (left <= right) {
    int comp;
    int middle;

    middle=(left+right)/2;  /* Εύρεση μεσαίου στοιχείου */
    comp=strcmp(key, arr[middle]);
    if (comp==0) return middle;
    if (comp<0) right=middle-1;
    else left=middle+1;
}
return -1;
}
```