

```

/*
A) Γράψτε ένα πρόγραμμα σε C σύμφωνα με το οποίο το σύστημα θα
διαβάζει ένα
αλφαριθμητικό από την βασική είσοδο και θα τυπώνει για κάθε
χαρακτήρα του το πόσες φορές αυτός
εμφανίζεται στο αλφαριθμητικό. <-- το συγκεκριμένο πρόβλημα
έχει επιλυθεί σε προηγούμενη
άσκηση (thema1,c)

B) Δώστε μια άλλη έκδοχή του παραπάνω προγράμματος ορίζοντας
και αξιοποιώντας μια
συνάρτηση η οποία δέχεται ως όρισμα ένα αλφαριθμητικό και
επιστρέφει το πόσες φορές
εμφανίζεται στο αλφαριθμητικό ο κάθε χαρακτήρας του.
(thema1_b.c)
*/

```

```

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

#define N 30

typedef struct info
{
    char ch;
    int plithos;
} INFO;

INFO * occurrences(char word[], int *size)
{
    int TOP=-1;
    int len; int i;
    len=strlen(word);
    INFO temp[len];          INFO *result;

    for (i=0; i<len; i++)
    {
        int count=0;
        int j;
        int flag=0;

        for (j=0; j<i; j++)
            if (word[j]==word[i]) flag=1;

        if (flag==1) continue;
        TOP=TOP+1;
        temp[TOP].ch=word[i];

        for (j=i; j<len; j++) if (word[i]==word[j]) count++;
        temp[TOP].plithos=count;
    }
}

```

```

    *size=TOP+1;

    result=(INFO *)malloc((TOP+1)*sizeof(INFO)); // INFO
    result[TOP+1];

    for (i=0; i<=TOP; i++) result[i]=temp[i];

    return result;
}

int main()
{
    char word[N+1];

    INFO * information; // INFO information[];
    int distinct;

    int i;
    int len;

    printf("Give word: ");
    scanf("%s", word);

    len=strlen(word);

    information=occurrences(word, &distinct);

    for (i=0; i<distinct; i++)
    printf("The character %c appears %d times\n",
    information[i].ch, information[i].plithos);

}

```