

```

runner$ cat freq_struct.c
/* Calculate frequencies of letters in an input file */
#include <stdio.h>
#include <ctype.h>
struct table_tag {
    char letter;
    int count;
};

void bubble(struct table_tag table[]);
void swap(struct table_tag *table1,
          struct table_tag *table2);
void printfreq(struct table_tag table[], int tot);

void main(void)
{
    struct table_tag table[26];
    int i;
    /* index for lc array */
    int tot = 0; /* total number of alpha characters */
    int ch; /* int so that EOF will work */
    for (i = 0; i < 26; i++) {
        table[i].letter = (char) (i + 'a');
        table[i].count = 0;
    }
    while ((ch = getchar()) != EOF)
        if (isalpha(ch = tolower(ch))) {
            tot++;
            table[ch - 'a'].count++;
        }
    bubble(table);
    printfreq(table, tot);
}

/* bubble: sort lc array into decreasing order.
   Carry alf along */
void bubble(struct table_tag table[])
{
    int i, dum;
    for (dum = 0; dum < 25; dum++)
        for (i = 0; i < 25; i++)
            if (table[i].count < table[i+1].count)
                swap(&table[i], &table[i+1]);
}

/* swap: interchange two structs */
void swap(struct table_tag *table1,
          struct table_tag *table2)
{
    struct table_tag temp;
    temp = *table1;
    *table1 = *table2;
    *table2 = temp;
}

```

```

/* printfreq: print out the frequency table */
void printfreq(struct table_tag table[], int tot)
{
    int i;
    printf("Frequency of letters, out of total: %d\n\n",
           tot);
    printf(" Letter Frequency (%%)\n");
    for (i = 0; i < 26; i++)
        printf("%6c %13.3f%%\n", table[i].letter,
              (double)table[i].count/tot*100.0);
}

runner$ freq_struct
AaAaA
bbbBBbbb
Zzzzzzzzzzzzzzzzzzzzz (ctrl-D)
Frequency of letters, out of total: 30

Letter Frequency (%)
Z 50.000%
b 33.333%
a 16.667%
-----
runner$ cat freq_typedef.c
/* Calculate frequencies of letters in an input file */
/* Variation using typedef */
#include <stdio.h>

typedef struct {
    char letter;
    int count;
} table_type;

void bubble(table_type table[]);
void swap(table_type *table1,
          table_type *table2);
void printfreq(table_type table[], int tot);

void main(void)
{
    table_type table[26];
    int i;
    /* index for lc array */
    int tot = 0; /* total number of alpha characters */
    int ch; /* int so that EOF will work */
    for (i = 0; i < 26; i++) {
        table[i].letter = (char) (i + 'a');
        table[i].count = 0;
    }
}

```

```

while ((ch = getchar()) != EOF)
    if (isalpha(ch = tolower(ch))) {
        tot++;
        table[ch - 'a'].count++;
    }
    bubble(table);
    printfreq(table, tot);
}

/* bubble: sort 1c array into decreasing order.
   Carry alf along */
void bubble(table_type table[])
{
    int i, dum;
    for (dum = 0; dum < 25; dum++)
        for (i = 0; i < 25; i++)
            if (table[i].count < table[i+1].count)
                swap(&table[i], &table[i+1]);
}

/* swap: interchange two structs */
void swap(table_type *table1, table_type *table2)
{
    table_type temp;
    temp = *table1;
    *table1 = *table2;
    *table2 = temp;
}

/* printfreq: print out the frequency table */
void printfreq(table_type table[], int tot)
{
    int i;
    printf("Frequency of letters, out of total: %d\n\n",
        tot);
    printf(" Letter Frequency (%%)\n");
    for (i = 0; i < 26; i++)
        printf("%6c %13.3f%%\n", table[i].letter,
            (double)table[i].count/tot*100.0);
}

}

runner% cat freq_simp.c
/* Calculate frequencies of letters in an input file */
/* Variation using simple global arrays */
#include <stdio.h>
#include <ctype.h>
char letter[26];
int count[26];

/* bubble: sort 1c array into decreasing order. */
void bubble(void)
{
    int i, dum;
    for (dum = 0; dum < 25; dum++)
        for (i = 0; i < 25; i++)
            if (count[i] < count[i+1]) {
                int temp = count[i];
                char tempc = letter[i];
                count[i] = count[i+1];
                letter[i] = letter[i+1];
                count[i+1] = temp;
                letter[i+1] = tempc;
            }
}

/* printfreq: print out the frequency table */
void printfreq(int tot)
{
    int i;
    printf("Frequency of letters, out of total: %d\n\n",
        tot);
    printf(" Letter Frequency (%%)\n");
    for (i = 0; i < 26; i++)
        printf("%6c %13.3f%%\n", letter[i],
            (double)count[i]/tot*100.0);
}

}

void main(void)
{
    int i;
    int tot = 0; /* index for 1c array */
    int tot = 0; /* total number of alpha characters */
    int ch; /* int so that EOF will work */
    for (i = 0; i < 26; i++) {
        letter[i] = (char) (i + 'a');
        count[i] = 0;
    }
    while ((ch = getchar()) != EOF)
        if (isalpha(ch = tolower(ch))) {
            tot++;
            count[ch - 'a']++;
        }
    bubble();
    printfreq(tot);
}

```