ICS 103, Term 093

Computer Programming in C

**HW# 4 Solution**

**Due date: Wednesday, August 18, 2010**

# You are required to write a C program to do the following:

## Ask the user to enter an input file name and read it.

## Count the number of occurrences of each of the alphabetic characters in the input file i.e. ‘a’ to ‘z’ regardless whether the character is small letter or capital.

## Print the characters in descending order of their occurrence along with their number of occurrences printing only characters with non-zero count.

*A sample execution of the program is shown below:*

|  |  |
| --- | --- |
| *Input file: hw4.txt* | *Histogram Display:* |
|  |  |

**#include <stdio.h>**

**#include <stdlib.h>**

**#define SIZE 26**

**void sort(int CC[], char C[], int size);**

**int main(void)**

**{**

 **FILE \*inf;**

 **char infname[40], ch;**

 **int CC[SIZE]={0}, i;**

 **char C[SIZE];**

 **printf("Enter the input file name: ");**

 **gets(infname);**

 **inf = fopen( infname, "r");**

 **if (inf == NULL){**

 **printf("Cannot open %s for reading \n", infname);**

 **system("pause");**

 **exit(1);**

 **}**

 **while ( fscanf(inf,"%c",&ch) != EOF ) {**

 **if ((ch>='a' && ch<='z')||(ch>='A' && ch<='Z')){**

 **ch = ch | 32; // convert chararcters to lower case**

 **C[(int)ch-97]=ch;**

 **CC[(int)ch-97]++;**

 **}**

 **}**

 **sort(CC, C, SIZE);**

 **for (i=0; i<SIZE;i++)**

 **if (CC[i] !=0)**

 **printf("%c: %d\n",C[i], CC[i]);**

 **system("pause");**

 **return 0;**

**}**

**void sort(int CC[], char C[], int size){**

 **int k,j,minpos,temp;**

 **char ctemp;**

 **for (k=0; k < size - 1; k++) {**

 **minpos = k;**

 **for(j = k+1; j < size; j++){**

 **if(CC[j] > CC[minpos])**

 **minpos = j;**

 **}**

 **temp = CC[minpos]; ctemp = C[minpos];**

 **CC[minpos] = CC[k]; C[minpos] = C[k];**

 **CC[k] = temp; C[k] = ctemp;**

 **}**

**}**

# Write a C program that displays the following menu:

1. Read Array
2. Print Array
3. Reverse a row
4. Reverse a column
5. Exit

Assume that the entered array will be a two dimensional array of integers and that the maximum number of rows and columns in the array is 15. Implement each of the menu options 1 to 4 as separate functions. Print the array after reversing a row or reversing a column. The menu should continue to be displayed as long as choice 5 is not selected. If a choice other than between 1 and 5 is entered, the statement “Invalid Choice” should be displayed.

*A sample execution of the program is shown below:*

|  |  |
| --- | --- |
|  |  |

**#include <stdio.h>**

**#include <stdlib.h>**

**#define MAX 15**

**void read\_array (int a[][MAX], int nr, int nc);**

**void print\_array (int a[][MAX], int nr, int nc);**

**void reverse\_row (int a[][MAX], int nc, int r);**

**void reverse\_col (int a[][MAX], int nr, int c);**

**void menu();**

**int main (void){**

**int a[MAX][MAX], nr, nc, i, j, r, c, ch;**

 **do{**

 **menu();**

 **scanf("%d", &ch);**

 **switch (ch){**

 **case 1:**

 **printf("Enter number of rows: ");**

 **scanf("%d",&nr);**

 **printf("Enter number of columns: ");**

 **scanf("%d",&nc);**

 **printf("Enter %d integers:\n",nr\*nc);**

 **read\_array (a, nr, nc);**

 **break;**

 **case 2:**

 **print\_array(a, nr, nc);**

 **break;**

 **case 3:**

 **printf("Enter a row number: ");**

 **scanf("%d",&r);**

 **reverse\_row(a, nc, r);**

 **print\_array(a, nr, nc);**

 **break;**

 **case 4:**

 **printf("Enter a column number: ");**

 **scanf("%d",&c);**

 **reverse\_col(a, nr, c);**

 **print\_array(a, nr, nc);**

 **break;**

 **case 5: break;**

 **default: printf("Invalid Choice \n");**

 **}**

 **} while (ch != 5);**

 **system ("pause");**

 **return 0;**

**}**

**void menu(){**

 **printf("Select a choice:\n");**

 **printf("1. Read Array \n");**

 **printf("2. Print Array \n");**

 **printf("3. Reverse a row \n");**

 **printf("4. Reverse a column \n");**

 **printf("5. Exit \n");**

**}**

**void read\_array (int a[][MAX], int nr, int nc){**

 **int i, j;**

 **for(i=0; i<nr; i++)**

 **for (j=0; j<nc; j++)**

 **scanf("%d", &a[i][j]);**

**}**

**void print\_array (int a[][MAX], int nr, int nc){**

 **int i, j;**

 **for(i=0; i<nr; i++){**

 **for (j=0; j<nc; j++)**

 **printf("%3d", a[i][j]);**

 **printf("\n");**

 **}**

**}**

**void reverse\_row (int a[][MAX], int nc, int r){**

 **int i, t;**

 **for(i=0; i<nc/2; i++){**

 **t=a[r][i];**

 **a[r][i]=a[r][nc-i-1];**

 **a[r][nc-i-1]=t;**

 **}**

**}**

**void reverse\_col (int a[][MAX], int nr, int c){**

 **int i, t;**

 **for(i=0; i<nr/2; i++){**

 **t=a[i][c];**

 **a[i][c]=a[nr-i-1][c];**

 **a[nr-i-1][c]=t;**

 **}**

**}**

# Write a C program that asks the user to enter a string of characters, str1, and another string of characters, str2. Then the program replaces all occurrences of str2 in str1 by \*. Assume that the maximum length of str1 and str2 is 80.

*A sample execution of the program is shown below:*

**

**#include <stdio.h>**

**#include <stdlib.h>**

**#include <string.h>**

**#define MAX 80**

**int main (void){**

 **char str1[MAX], str2[MAX], \*index;**

 **printf("Enter a string: ");**

 **gets(str1);**

 **printf("Enter another string: ");**

 **gets(str2);**

 **index=strstr(str1, str2);**

 **while (index != NULL){**

 **for (int i=0; i<strlen(str2); i++)**

 **index[i]='\*';**

 **index=strstr(index+strlen(str2), str2);**

 **}**

 **printf("Updated string: ");**

 **puts(str1);**

 **system ("pause");**

 **return 0;**

**}**