**King Fahd University of Petroleum and Minerals**

**Information and Computer Science Department**

ICS 103: Computer Programming in C

**Summer Semester 2009-2010 (Term-093)**

##### Final Exam

**Time: 120 minutes Wednesday August 25, 2010**

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Name:** |  | | | | | | | | | | | |
| **ID#:** |  |  |  |  |  |  |  |  |  |  |  |

PLEASE CIRCLE YOUR SECTION BELOW:

|  |  |  |  |
| --- | --- | --- | --- |
| Section | 01 | 02 | 03 |
| Instructor | Mr. AHMAD IRFAN | Dr. FARAG AZZEDIN | Dr. AIMAN EL-MALEH |
| Time | SUMT  9:20-10:10am | SUMT  10:30-11:20am | SUMT  10:30-11:20am |

|  |  |  |
| --- | --- | --- |
| Question # | Maximum Marks | Obtained Marks |
| 1 | 36 |  |
| 2 | 20 |  |
| 3 | 20 |  |
| 4 | 24 |  |
| Total | 100 |  |

**Notes.** 1. Make sure you have **TEN** pages including the cover page.

2. Closed book and notes

3. Write clearly, briefly and precisely

4. Cheating will result in ZERO grade

## Good Luck

## Question 1: (36 points)

**Determine the output of each of the following programs:**

|  |  |
| --- | --- |
| #include <stdio.h> // P1: 6 points  #include <string.h>  int main() {  char s[]="seaqwwa901qwa0";  for(int i=0; s[i]!='\0'; i++) {  if (s[i]=='a')  printf("%d\n", strlen(s)-i);  }  return 0;  } |  |
| #include <stdio.h> // P2: 6 points  int main() {  int a[3][3],b[3][3];  int i,j;  for(i=0;i<3;i++) {  for(j=0;j<3;j++)  a[i][j]=i+j;  }  for(i=0;i<3;i++) {  for(j=0;j<3;j++) {  b[j][i] = a[i][j] + i;  printf("%4d ",b[j][i]);  }  printf("\n");  }  return 0;  } |  |

|  |  |
| --- | --- |
| #include <stdio.h> // P3: 6 points  int main() {  int num = 120123456;    while(num!=0){  int rem = num%10;  num = num/10;  if (rem%2)continue;  if(rem == 0)break;  printf("%d\n", rem);  }  return 0;  } |  |
| #include <stdio.h> // P4: 6 points  #include <string.h>  #include <ctype.h>  int main() {  char a[10],b[]="keq",c[]="jdr";  int i,len;  len=strlen(b);  for(i=0;i<len;i++) {  if(b[i] > c[i])  a[i]=toupper(c[i]-1);  else  a[i]=toupper(c[i]+1);  }  a[i]='\0';  puts(a);  return 0;  } |  |

|  |  |
| --- | --- |
| #include <stdio.h> // P5: 6 points  #include <string.h>  int main(void) {  char str[4][80]={"ICS", "SWE", "SE", "COE"};  char tstr[80];  int i, p=1, s;  do{  s = 0;  for(i = 1; i <= 4-p; i++) {  if (strcmp(str[i-1],str[i])<0){  strcpy(tstr,str[i-1]);  strcpy(str[i-1],str[i]);  strcpy(str[i],tstr);  s=1;  }  }  p++;  for (i=0; i<4; i++)  puts(str[i]);  } while (s && p <= 3);  return 0;  } |  |
| #include <stdio.h> // P6: 6 points  #include <string.h>  #include <ctype.h>  int main() {  char str1[]={"ICS 103 is Interesting"};  char \*t;  int count=0;  t = strtok(str1, " ");  while ( t != NULL ) {  if (tolower(t[0])=='i'){  puts(t);  count++;  }  t = strtok(NULL, " ");  }  printf("%d\n", count);  return 0;  } |  |

**Question 2: ( 20 points )**

Write a complete C program that prompts a user to enter 64 integer numbers; the program stores them in a 1D array. Then, it converts the 1D array into an 8x8 2D array and prints the 2D array row-wise in the following format (as an example):

Row 0: 3 4 5 6 8 2 4 1

Row 1: 4 7 4 3 7 5 8 4

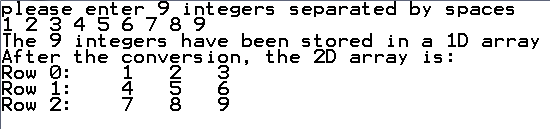
…

Row 7: 4 2 7 5 7 9 3 7

*You must use loop to fill the arrays.*

Your solution must use a function called **read\_1D\_array** to prompt the user to enter 64 integers and stores them in the ID array. You should also use a function called **convert** that takes the 1D array and converts it into an 8x8 2D array. The function **convert** will need arguments such as the input **1D\_array** and the resulting **2D\_array**. Finally, you should write a function called **print\_2D\_array** that takes the **2D\_array** as an argument **and prints it to the screen.** Define the dimensions of the 1D and 2D arrays using define preprocessor directive.

*Your program should produce the following output. This output is only for 9 integers as an example. Your program should work for 64 integers as specified above.*

****

**Question 3: ( 20 points )**

Write a C program that accepts a string from user and counts the number of one-letter words, two-letter words, three-letter words, etc, appearing in the text. Assume that there is no word greater than 10 letters. Then, your program prints the count of words whose count is not 0. Use an array to store the various words counts. Your program should then reformat and print the string such that every word starts with a capital letter.

*Sample execution of the program is given below:*

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

**Question 4: ( 24 points )**

Write a C program that does the following:

1. Reads a list of names and scores from the file “input.txt” and stores them in appropriate arrays. Assume that your program does not know the size of data and that the maximum number of names and scores is 100 and the maximum name length is 20.
2. The **selection** **sort algorithm** to sort an array of integers in decreasing order is given below as a function. Modify the function and use it to sort the data read in part (i) from highest score to lowest score and store the results in file “sorted\_scores.txt”.

void selection\_sort(int x[], int size){

int k,j,pos,temp;

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

pos = k;

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

if(x[j] > x[pos])

pos = j;

}

temp = x[pos];

x[pos] = x[k];

x[k] = temp;

}

}

1. Write a function to sort an array of strings in alphabetical increasing order based on modifying the selection sort algorithm given in (ii). Then use this function to sort the data read in part (i) according to names and store the results in the file “sorted\_names.txt”.

*Sample execution of the program is given below:*

|  |  |  |
| --- | --- | --- |
| **input.txt** | **sorted\_scores.txt** | **sorted\_names.txt** |
|  |  |  |