Information and Computer Science Department

Second Semester 162

ICS 103 – Computer Programming in C

Major 02 Examination

Saturday, May 6, 2017 (6:30 - 8:30 PM)

Duration: 120 minutes

|  |  |  |  |
| --- | --- | --- | --- |
| Name: | KEY | Lecture Serial # |  |

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ID#: |  |  |  |  |  |  |  |  |  |

Please tick your section:

|  |  |
| --- | --- |
| Instructor | Section |
| Dr. Louai Al-Awami | [ ] 01 (UT 7:00 – 7:50) [ ] 02 (UT 7:00 – 7:50)[ ] 03 (UT 8:00 – 8:50) [ ] 04 (UT 8:00 – 8:50) |
| Dr. Samer Arafat | [ ] 05 (UT 11:00 – 11:50) [ ] 06 (UT 11:00 – 11:50) |
| Dr. Mohammed Balah | [ ] 07 (UT 13:10 – 14:00) [ ] 08 (UT 13:10 – 14:00)  |
| Mr. Said Abdallah Muhammad | [ ] 09 (MW 7:00 – 7:50) [ ] 10 (MW 7:00 – 7:50)  |
| Dr. Tarek El-Basuny | [ ] 11 (MW 8:00 – 8:50) [ ] 12 (MW 8:00 – 8:50) |
| Dr. Rafiul Hassan | [ ] 13 (MW 9:00 – 9:50) [ ] 14 (MW 9:00 – 9:50)[ ] 15 (MW 10:00 – 10:50) [ ] 16 (MW 10:00 – 10:50) |
| Dr. El-Sayed El-Alfy | [ ] 17 (MW 11:00 – 11:50) [ ] 18 (MW 11:00 – 11:50)[ ] 19 (MW 13:10 – 14:00) [ ] 20 (MW 13:10 – 14:00) |

Instructions:

1. Answer all questions. Make sure your answers are clear and readable.
2. Make sure there are 5 questions in 13 pages.
3. The exam is closed book and closed notes. No calculators or any helping aides are allowed. Make sure to turn off your mobile phone and keep it in your pocket.
4. If there is no space on the front of a question’s page, use the back of the page. Indicate this clearly.

|  |  |  |  |
| --- | --- | --- | --- |
| Question # | Max Mark | Obtained Mark | Remarks |
| 1 | 15 |  |  |
| 2 | 30 |  |  |
| 3 | 15 |  |  |
| 4 | 20 |  |  |
| 5 | 20 |  |  |
| Total | 100 |  |  |

Question 1 (15 points): Fill out the following table by selecting the best answer for each of the following 10 Multiple Choice questions. BE SURE TO WRITE THE FINAL ANSWER IN THE TABLE.

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Question | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| Answer | D | A | C | D | A | C | C | B | C | C |

1. A function named *test* that receives one integer value and does not return a value has which one of the following prototypes?
2. int test(int x);
3. int test;
4. test (int x);
5. void test(int x);
6. Which of the following is a valid function call for a function that does not receive any value?
7. fnc( );
8. fnc;
9. void fnc( );
10. void fnc(void);
11. Which of the following statements will read and store 4 input values into an integer array *arr1* of size 4 in reverse order?
12. for (i = 0; i < 4; i++) scanf("%d",&arr1[i]);
13. for (i = 3; i <= 0; i--) scanf("%d",& arr1[i]);
14. for (i = 3; i >= 0; i--) scanf("%d",&arr1[i]);
15. for (i = 0; i <= 4; i++) scanf("%d",&arr1[4-i]);
16. If an array name is passed as an argument to a function, what actually gets passed?
	1. The value of the first element of the array
	2. The values of all the elements of the array
	3. The address of the last element of the array
	4. The address of the first element of the array
17. What is the prototype for a function that computes a multiplication of array elements A and B each of size n and puts the result back in array A ?
	1. void mult(int A[ ], int B[ ], int n);

|  |
| --- |
| * 1. void mult (int A[n], int B[n], int n);
	2. int mult (int A[n], int B[n], int n);
	3. void mult (int\* A[ ], int\* B[ ], int n);
 |
|  |

1. Consider the following array:

 int m[3] ={7,2};

What is the value of m[2]?

1. 2
2. 7
3. 0
4. NULL
5. What is the output of the following program?

#include <stdio.h>

int main(){

 int k, m, count = 0;

 for(k = 0; k <= 4; k++)

 for(m = 3; m > 0; m--)

 count++ ;

 printf("%d", count);

return 0;

}

* 1. 20
	2. 16
	3. 15
	4. 12
1. What is the output of the following program?

 #include <stdio.h>

 int main(){

int x = 30, y = 15;

int \*p1, \*p2;

p1 = &x;

p2 = &y;

x = \*p1 + \*p2;

\*p1 = 2 \* \*p1;

printf("%d",x);

 return 0;

 }

1. 60
2. 90
3. 30
4. 45
5. What is the output of the following program fragment?

int i=5, j=10;

printf("%d",i++ + --j);

1. 15
2. 16
3. 14
4. 13
5. What is the output of the following program?

#include <stdio.h>

void cube (int i){

 i=i\*i\*i;}

int main (void){

 int i = 2;

 cube(i);

 printf("%d",i);

 return 0;

 }

1. 8
2. 4
3. 2
4. 1

Question 2 (30 points)

Question 2 Part 1 (25 points): Show the output of each of the following programs:

|  |  |
| --- | --- |
| #include <stdio.h> // 7 pointsint fnc(int \*a, int \*b, int c){ int d = \*a; \*a = \*b + c ; c = \*a + \*b ; return d;}int main() { int a=5,b=10,c=20, d=0; int \*p1,\*p2; p1=&a; p2=&b; \*p1=2+\*p2; d = fnc(&a , &b, c) ; printf("%d %d %d %d",a,b,c,d); return 0;} | 30 10 20 12 |
| #include <stdio.h> // 6 points int main(){ int count=1,m,n; m=2; do{ n=5; while(n>=3){ count++; n=n-1; } m=m+3; }while(m<=3); printf("%d %d %d\n",m,n,count);return 0;} | 5 2 4  |
| #include <stdio.h> // 6 pointsint main (void){int i, j,sum;for (i=0; i<3; i++){ sum=0; for (j=0; j<3; j++) {  if (j%2 != 1){ sum += i-j; }  } printf("%d\n",sum);} return 0;} | -202 |
| #include <stdio.h> // 6 pointsvoid pick(int a, int \*b) { if(a < \*b) { \*b=\*b+3; a=a-10;  } else { \*b=\*b-3; a=a+10; }}int main() { int i; int a[]={2,4,8,10,20,100,25,15,7,3,1}; for(i=0;i<10;i=i+2) { pick(a[i],&a[i+2]); } for(i=0;i<10;i=i+2) printf("%d ",a[i]);return 0; } | 2 11 23 28 4 |

Question 2 Part 2 (5 points): Convert the following for loop into an equivalent while loop:

|  |  |
| --- | --- |
| int x; // 5 pointsfor (x = 0; x < 10; x++)  printf( "%d\n", x ); | int x = 0; while ( x < 10 ) {  printf( "%d\n", x )  x++; }  |

Question 3 (15 points):

An internet service provider charges customers 8.00 Saudi Riyals per hour for the first 10 hours of connection time. Each hour or partial hour in excess of 10 hours is charged 2.50 Saudi Riyals. Write a complete C program that calls in which the main function calls a single function getChargeAndAverage that receives the number of hours for a customer, it then returns the total charge and the average charge per hour for the customer.

Given a texfile input.txt of the form:

|  |
| --- |
| 15362 4.242768 11.111111 9.923454 12.040333 1.033578 15.0 |

 where each line contains a customer ID and his connection time in hours, write the main function that reads the input textfile and uses the function getChargeAndAverage to generate an output file charges.txt of the form:

|  |
| --- |
| CustomerID Hours used Average charge(SAR) Total charge(SAR) 15362 4.2 8.00 33.60 42768 11.1 7.66 85.00 11111 9.9 8.00 79.20 23454 12.0 7.08 85.00 40333 1.0 8.00 8.00 33578 15.0 6.17 92.50 TOTAL 383.30 |

 Hint: Your getChargeAndAverage function should use the standard mathematical function ceil

|  |  |  |  |
| --- | --- | --- | --- |
| C function | Mathematical Notation | Example | Comment |
| ceil(x) | ⎡ x ⎤ | ceil(45.1) = 46.0ceil(-7.9) = -7.0 | Returns the smallest integral value (of type double) greater or equal to x |

Note:

* The function getChargeAndAverage must not contain scanf and printf calls.
* The function getChargeAndAverage must be written after the main function.
* Your program must be general; it must work for any number of lines in the input file.

#include <stdio.h>

#include <stdlib.h>

#include <math.h>

#define FLATRATE 8.00

#define ADDITIONALRATE 2.50

void getChargeAndAverage(double hours, double\* charge, double\* average);

int main(void){

 int customerID;

 double hours, charge, average, totalCharge = 0;

 FILE \*inptr, \*outptr;

 inptr = fopen("usage.txt", "r");

 if(inptr == NULL){

 printf("Error: cannot open usage.txt\n");

 exit(1) ;

 }

 outptr = fopen("charges.txt", "w");

 fprintf(outptr, "%10s%14s%23s%21s\n",

 "CustomerID", "Hours used", "Average charge(SAR)", "Total charge(SAR)");

 while(fscanf(inptr,"%d%lf", &customerID, &hours) != EOF){

 getChargeAndAverage(hours, &charge, &average);

 totalCharge += charge;

 fprintf(outptr,"%10d%14.1f%23.2f%21.2f\n",customerID,hours,average,charge);

 }

 fprintf(outptr, "\n%58s%10.2f\n", "TOTAL", totalCharge);

 fclose(inptr);

 fclose(outptr);

 printf("Output file charges.txt created . . .");

 return 0;

}

void getChargeAndAverage(double hours, double\* charge, double\* average){

 if(hours <= 10)

 \*charge = hours \* FLATRATE;

 else

 \*charge = 10.0 \* FLATRATE + ceil(hours - 10.0)\* ADDITIONALRATE;

 \*average = \*charge / hours;

}

Question 4 (20 points):

Assume two non-zero vectors in n-dimensional space are stored in two 1D-arrays

 *a* = {*a*0, *a*1, . . . , *an*-1} and b = {*b*0, *b*1, . . . , *b*n-1} each of type double. The cosine of the angle between these two vectors is computed by the equation:

$$\cos(θ)= \frac{\sum\_{k=0}^{n-1}a\_{k}b\_{k}}{\sqrt{\sum\_{k =0}^{n-1}a\_{k}^{2}}\sqrt{\sum\_{k=0}^{n-1}b\_{k}^{2}}}$$

Where:

 $\sum\_{k=0}^{n-1}a\_{k}b\_{k}= a\_{0}b\_{0}+ a\_{1}b\_{1}+ a\_{2}b\_{2}+ . . . + a\_{n-1}b\_{n-1}$

 $\sum\_{k=0}^{n-1}a\_{k}^{2}= a\_{0}^{2}+ a\_{1}^{2}+ a\_{2}^{2}+ . . . + a\_{n-1}^{2}$

 $\sum\_{k=0}^{n-1}b\_{k}^{2}= b\_{0}^{2}+ b\_{1}^{2}+ b\_{2}^{2}+ . . . + b\_{n-1}^{2}$

Write a complete C program that uses the following four functions:

 readArray - it receives an output 1D-array parameter x of type double and its size. It then reads values

 typed by the user from the keyboard into the array.

 dotProduct – it receives two input 1D-array parameters x and y, each of type double, and their size. It

 then computes and returns the dot product of two vectors as given by $\sum\_{k=0}^{n-1}x\_{k}y\_{k}$

 magnitude – it receives an input 1D-array parameter x of type double and its size. It then computes

 and returns the magnitude of the vector given by $\sqrt{\sum\_{k=0}^{n-1}x\_{k}^{2}}$

 radiansToDegrees - it receives an angle in radians and converts it to degrees.

The program prompts the user to enter the number of elements n in each vector, this number must be between 1 and 100 inclusive. Your program must display an error message and loop as long as n is not in the interval [1, 100]. If n is valid, the program creates two 1D-arrays a and b each of type double and size n it then uses the readArray function to read values into each of the two vectors a and b and then it uses the functions dotProduct and magnitude to calculate the cosine of the angle between the vectors using the above equation. Your program must print both the cosine the angle in degrees.

Hint: 180o = π radians

 The C standard function for cos-1(x) is acos(x)

Note:

* The function readArray must contain no printf call.
* The functions dotProduct, magnitude, and radiansToDegrees must contain no scanf and printf calls.
* The four functions must be defined after the main function.
* Your function must define a constant π with the value 3.141592
* You program must be general and behave as in the sample program runs below.

Sample program runs:

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

#include <stdio.h>

#include <math.h>

#define PI 3.141592

void read\_array(double x[ ], int n);

double dotProduct(double x[ ], double y[ ], int n);

double magnitude(double x[ ], int n);

double radiansToDegrees(double radians);

int main(void){

 int n;

 double cosTheta, radians;

 do{

 printf("Enter the number of elements in each vector [should be <=100]:\n");

 scanf("%d", &n);

 if(n <= 0 || n > 100)

 printf("Error: Invalid vector size\n");

 } while(n <= 0 || n > 100);

 double a[n], b[n];

 printf("Enter %d elements of the first vector: ", n);

 read\_array(a, n);

 printf("Enter %d elements of the second vector: ", n);

 read\_array(b, n);

 cosTheta = dotProduct(a, b, n) / (magnitude(a, n) \* magnitude(b, n));

 printf("The cosine of the angle between the vectors is: %.4f", cosTheta);

 radians = acos(cosTheta);

 printf("The angle between the vectors is: %.4f", radiansToDegrees(radians));

 return 0;

}

void read\_array(double x[ ], int n) {

 int i;

 for(i = 0; i < n; i++)

 scanf("%lf", &x[i]);

}

double dotProduct(double x[ ], double y[ ], int n){

 double z = 0;

 int i;

 for(i = 0; i < n; i++)

 z += x[i] \* y[i];

 return z;

}

double magnitude(double x[ ], int n) {

 double mg = 0;

 int i;

 for(i = 0; i < n; i++)

 mg += x[i] \* x[i];

 return sqrt(mg);

}

double radiansToDegrees(double radians){

 return 180 \* radians / PI;

}

Question 5 (20 points):

Write a C program that uses two integer 1D-arrays x and frequency, each of size 20, to solve the following problem: Prompt the user to enter the number of integers N to read. Assume that N ≥1 and N ≤ 20. Prompt the user to enter N integers. As each integer is read assign it to the first current empty location of array x only if it is not a duplicate of a number already read. Use the array frequency to count the frequency of each integer read (that is, the number of times that integer is read). Your program finally prints a table of the non-duplicate values read and their frequencies.

Sample program runs:

|  |
| --- |
|  |
|  |

#include <stdio.h>

#define SIZE 20

int main( void ) {

 int a[SIZE], frequency[SIZE] = {0};

 int i, j, count = 0, n, isDuplicate, value;

 printf("Enter the number of elements N to read [1 <= N <= 20]: ");

 scanf("%d", &n);

 printf( "Enter %d integers: ", n );

 for ( i = 0; i <= n - 1; i++ ) {

 isDuplicate = 0;

 scanf( "%d", &value );

 for ( j = 0; j < count; ++j ){

 if ( value == a[ j ] ) {

 isDuplicate = 1;

 frequency[j]++;

 break;

 }

 }

 if ( ! isDuplicate ){

 a[count] = value;

 count++;

 frequency[j] = 1;

 }

 }

 printf ("\n%5s%15s\n", "Value", "frequency" );

 for ( i = 0; i < count ; i++ )

 printf( "%5d%15d\n", a[ i ], frequency[i] );

 return 0;

}