KING FAHD UNIVERSITY OF PETROLEUM AND MINERALS

Information and Computer Science Department

2013 Summer Semester (Term 123)

ICS103 Computer Programming in C (2-3-3)

**Final Exam**

JULY 28, 2013

120 Minutes

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Exam Code | **001** | | | | | | | | | | | | | |
| Student Name |  | | | | | | | | | | | | | |
| KFUPM ID |  |  | |  |  | |  |  | |  | |  | |  |
| CLASS SECTION | DR. SALAH ADAM | | SECN 01 | | |  | UMTW 9: 20 AM | |  | |  | |  | |
| DR. SALAH ADAM | | SECN 03 | | |  | UMTW 10: 30 AM | |  | |  | |  | |
| DR. FARAG EZZEDIN | | SECN 02 | | |  | UMTW 9: 20 AM | |  | |  | |  | |
| DR. FARAG EZZEDIN | | SECN 04 | | |  | UMTW 10: 30 AM | |  | |  | |  | |
| DR. FARAG EZZEDIN | | SECN 05 | | |  | UMTW 1: 10 PM | |  | |  | |  | |

**IMPORTANT NOTES**

* Fill-in your information on the answer sheet.
* **Answer all fifty (50) questions.**
* **Mark your answers on the answer sheet.**
* **The answer sheet is the only one that will be graded.**
* Do NOT start the exam until you are instructed to do so.
* This is a closed material exam. So, remove any relevant material.
* Calculators are NOT allowed. If you have one, put it on the floor.
* Mobile phones are NOT allowed. If you have one, switch it off NOW.

1. How many times will “KFUPM” be printed by the following C program segment?

**int j,k;**

**for(k = 0; k < 5; k++)**

**for(j = k; j < 5; j++)**

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

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

1. 50
2. 25
3. 30
4. 16
5. What is the output of the following program?

**#include <stdio.h>**

**void fun(int y, int \*x);**

**int main(){**

**int x = 2;**

**int y = 3;**

**fun(x, &y);**

**printf("%d, %d\n", x, y);**

**system("pause");**

**return 0;**

**}**

**void fun(int y, int \*x){**

**y = 40;**

**\*x = 80;**

**}**

1. 2, 3
2. 2, 80
3. 80, 3
4. 80, 40
5. What is the output of the following C code segment? Assume the variables x and y are located in memory address 2000 and 4000 respectively.

**int x = 4;**

**int \*y = &x;**

**\*y = 7;**

**printf("%d, %d, %d, %d, %d\n", x, &x, y, &y, \*y);**

1. 4, 2000, 2000, 4000, 4
2. 4, 2000, 4000, 4000, 4
3. 4, 2000, 4000, 2000, 4
4. 7, 2000, 2000, 4000, 7
5. What is the output of the following C code segment?

**char s1[] = "10";**

**char s2[] = "2";**

**if(strcmp(s1, s2) == 0)**

**strcat(s1, s2);**

**else**

**strcat(s2,s1);**

**printf("%s, %s\n", s1, s2);**

1. 0, 210
2. 0, 0
3. 102, 102
4. 10, 210
5. Use the following code to answer this and the next question. The following C code segment is supposed to print the sum of each row. Where should the statement **sum = 0;** be written;(The number on the beginning of each statement is the statement number and is not part of the code.)

**1: double x[][3] = {{1,2,3}, {4,5,6}, {7,8,9}};**

**2: double sum;**

**3: inti, j;**

**4: for(i=0; i < 3; i++) {**

**5: for(j=0; j < 3; j++) {**

**6: sum += x[i][j];**

**7: }**

**8: }**

**9: return 0;**

1. Between statements 3 and 4
2. Between statements 4 and 5
3. Between statements 5 and 6
4. Between statements 7 and 8
5. Where should the statement **printf(“%f \n”, sum);** be written;
6. Between statements 4 and 5
7. Between statements 5 and 6
8. Between statements 7 and 8
9. Between statements 8 and 9
10. Use the following code to answer this and the next question. The following C code segment is supposed to swap the values of v1 and v2. What should be written as statement 3?

**1: int v1, v2, temp;**

**2: scanf(“%d %d”, &v1, &v2);**

**3: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**4: v1 = v2;**

**5: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**9: return 0;**

1. v2 = v1;
2. temp = v2;
3. temp = v1;
4. v2 = temp;
5. What should be written as statement 5?
6. v2 = v1;
7. temp = v2;
8. temp = v1;
9. v2 = temp;
10. Which of the following input strings will give the output “BBB”. (The function strrev reverses a string.)

**#include <stdio.h>**

**#include <string.h>**

**int main()**

**{**

**char a[100], b[100];**

**printf("Enter a string\n");**

**gets(a);**

**strcpy(b,a);**

**strrev(b);**

**if(strcmp(a,b) == 0 )**

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

**else**

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

**system("pause");**

**return 0;**

**}**

1. BBB
2. AABAA
3. AABB
4. ABABA
5. Which of the following C statements is wrong?
6. int x[3][3] = {{1}, {2}, {3}};
7. int x[][3] = {{1}, {2}, {3}};
8. int x[2][3] = {{1}, {2}};
9. int x[3][] = {{1}, {2}, {3}};

11) What will be the output of the following code fragment:

int x[3][3]={{1,2,3},{4,5,6},{7,8,9}}, i, j;

for(j=2; j>=0; j--)

for(i=2; i>j; i--)

printf("%d ",x[i][j]);

1. 8 4 7
2. 4 7 8
3. 8 7 4
4. 6 3 2

**The next 3 questions are based on the following incomplete code fragment:**

Assume that array x is declared with **5** rows and **7** columns and initialized with values. The objective of the incomplete code fragment shown below is to swap the values of the last column of array x with the values of the first column.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_// Statement 1 {

temp = x[i][6];

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_// Statement 2

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_// Statement 3

}

12) Statement 1 should be:

1. for(i=0;i<7;i++)
2. for(i=0;i<5;i++)
3. for(i=5;i>=0;i--)
4. for(i=7;i>=0;i--)

13) Statement 2 should be:

1. x[i][0]=temp;
2. x[i][0]=x[i][6];
3. x[i][6]=x[i][0];
4. x[i][6]=temp;

14) Statement 3 should be:

1. x[i][0]=temp;
2. x[i][0]=x[i][6];
3. x[i][6]=x[i][0];
4. x[i][6]=temp;

15) What is the output of the following code fragment?

int m = 5, y = 3;

int \*p1, \*p2;

p1 = &m;

p2 = &y;

\*p1 = m + y;

\*p2 = m + y;

printf("%d %d\n",m,y);

1. 8 8
2. 11 11
3. 11 8
4. 8 11

**The next 2 questions are based on the following code fragment:**

int A[5] = {7,8,15,4,3},B[5];

int i,j=0,k=4;

for (i=0; i<5; i++)

if (A[i] % 2 == 0)

B[j++]=A[i];

else

B[k--]=A[i];

printf("%d %d",j,k);

16) What is the output after executing the above code

1. 4 0
2. 2 2
3. 2 1
4. 3 1

17) What will be the values of array B (from B[0] to B[4]) after executing the above code?

1. 8 4 7 15 3
2. 8 4 3 15 7
3. 7 15 3 8 4
4. 7 15 3 4 8

18) What is the output of the following C code fragment?  
 int i=4,j,k;

for(j=5;j>i;j--) {

for(k=1;k<=i;k=k+3)

i++;

j--;

}

printf("%d %d %d \n",i,j,k);

A. 5 3 4

B. 6 4 7

C. 5 4 7

D. 6 3 7

19) Consider the following code fragment

char a[]="72ABTM%(;atqhn";

int i, count=0;

for(i=0; i<14; i++) {

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ // if statement

count++;}

Which of the following is the right **if statement** so that the value of the variable

count becomes **9** at the end of the loop.

1. if(!isupper(a[i]))
2. if(!islower(a[i]))
3. if(!isdigit(a[i]))
4. if(!isalnum(a[i]))

20) Assume the following declaration and missing function call

double m[5][4];

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_// call to function test**

The function **test** has the following prototype:

void test (int rows, int cols, double a[ ][4]);

Which one of the following is the correct call to the function **test?**

1. test (5,4,m);
2. test (5,4,m[][]);
3. test (5,4,m[][4]);
4. void test (5,4,m);

21) If the function fun is defined as follows:

void fun (int \*a, int b, int \*c) {

\*a = \*a - b;

b = b + \*c;

\*c = \*c + 1;

}

What is the output of thefollowing program fragment ?

int x[3]={2,5,9};

fun(&x[2], x[1], &x[0]);

printf("%d %d %d\n", x[0],x[1],x[2]);

1. 9 5 2
2. 3 5 3
3. 3 5 4
4. 2 5 9

22) Which code fragment will initialize array x as shown?

|  |  |  |  |
| --- | --- | --- | --- |
| 0 | 1 | 2 | 3 |
| 1 | 2 | 3 | 4 |
| 2 | 3 | 4 | 5 |
| 3 | 4 | 5 | 6 |

|  |  |
| --- | --- |
| 1. int k, m, x[4][4];   for(m = 0; m <= 3; m++)  for(k = 3; k >= 0; k--)  x[m][k] = 6 - (k + m); | 1. int k, m, x[4][4];   for(k = 0; k <= 3; k++)  for(m = 0; m <= 3; m++)  x[k][m] = k + m; |
| 1. int k, m, x[4][4];   for(k = 3; k >= 0; k--)  for(m = 0; m >= 0; m--)  x[k][4- m] = k + m; | 1. int j = 0;   for(k = 0; k <= 3; k++)  for(m = 0; m <= 3; m++)  x[k][m] = j++; |

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

23) Consider the **selection sort** function. Select the correct order of the elements of the array after each pass to sort the following array in increasing order: {6,8,4,3,7};

1. {8,6,4,3,7} {8,7,4,3,6} {8,7,6,3,4} {8,7,6,4,3}
2. {3,8,4,6,7} {3,4,8,6,7} {3,4,8,6,7} {4,5,6,7,8}
3. {6,4,3,7,8} {4,3,6,7,8} {3,4,6,7,8} {3,4,6,7,8}
4. {3,8,4,6,7} {3,4,8,6,7} {3,4,6,8,7} {3,4,6,7,8}

24) Consider the following code fragment:

int k, m;

for(k=1; k<=8; k++)

for(m = 4; m >= k; m--)

printf("ICS 103\n");

How many times is “ICS 103” printed?

1. 10
2. 12
3. 28
4. 4

25) If the function mystery is defined as follows:

int mystery(int a, int b) {

if(a == -1)

return 3;

else {

printf("%d%d\t",b, a);

return mystery(a - 3,b + 2);

}}

What is the output generated by the following call?

printf("%d ",mystery(8,2));

1. 28 45 62 3
2. 82 54 26 3
3. 28 45 62
4. 82 54 26

26) Which one of the following is the correct code to print the elements of an integer array y of

size 8?

A. for(i = 0; i <= 8; ++i)

printf(“%d ”, y[i]);

B. for(i = 1; i <= 8; i++)

printf(“%d ”,&y[i]);

C. for(i = 0; i < 8; i++)

printf(“%d ”,&y[i]);

D. for(i = 0; i < 8; i++)

printf(“%d ”,y[i]);

27) What is the output of the following code fragment?

int x[3][3]={{1,2,3},{4,5,6},{7,8,9}},k;

int sum = 0;

for(k = 0; k < 3; k++){

sum += x[k][2-k];

printf("%d ",sum);

}

1. 3 5 7
2. 3 8 15
3. 1 5 9
4. 9 5 1

28) What is the output of the following code fragment?

int flag = 2;

switch(flag){

case 2: printf("Apple");

case 1: printf("Orange");

break;

case 3: printf("Banana");

default: printf("Guava");

}

1. AppleOrange
2. AppleBananaGuava
3. Apple
4. AppleGuava

29) What is the output of the following code fragment?

int k, m = 1;

printf("%d ",++m);

k = m++\*2;

printf("%d %d",m,k);

A. 1 3 4   
B. 2 2 4  
C. 1 2 4

D. 2 3 4

**The next 2 questions are based on the following code fragment:**

scanf("%d%d",&x,&y);

if(x > y)

if(y < 20)

if(x > 10)

printf("1");

else

printf("2");

else

printf("3");

else

printf("4");

30) What is the output if the user has typed **7 3** as input

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

31) What is the output if the user has typed **13 17** as input

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

32) What is the output after executing the following code fragment?

char line[] = "Exams123";

int i;

for(i = 0; line[i] != '\0'; i+=2)

printf("%c", toupper(line[i]));

1. EXAMS123
2. ExAmS123
3. Exams123
4. EAS2

33) Consider the following declaration and initialization:

char cities[3][15]={"Dammam","Taif","Dhahran"};

What is the correct way to display the length of string “Dammam” ?

1. printf("%d ",strlen(cities[0][6]));
2. printf("%d ",strlen(cities[0]));
3. printf("%d ",strlen(cities[1][15]));
4. printf("%d ",strlen(cities[1]));

34) What is the output of the code fragment shown below?

int x[] = {5, 6, 7, 8, 9, 10}, sum = 0, i = 0;

while(i < 4) sum += x[++i];

printf("%d ",sum);

1. 15
2. 30
3. 45
4. 60

35) What will be printed by the following statement?

printf("%d ",strcmp("apple","AppleMacintosh"));

1. Positive Value
2. Negative Value
3. 0
4. None of the above

36) Consider **the bubble sort** function. Select the correct order of the elements of the array after each pass to sort the following array in increasing order: {6,4,3,8,0}

1. {0,6,4,3,8} {0,3,6,4,8} {0,3,4,6,8} {0,3,4,6,8}
2. {0,6,4,3,8} {0,3,4,8,6} {0,3,4,8,6} {0,3,4,6,8}
3. {4,3,6,0,8} {3,4,6,0,8} {0,3,6,4,8} {0,3,4,6,8}
4. {4,3,6,0,8} {3,4,0,6,8} {3,0,4,6,8} {0,3,4,6,8}

**The next 5 questions are based on the following code**

#include <stdio.h>

#include <math.h>

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_; // Statement 1 (prototype)**

int main() {

int x[3]={1,2,3};

double avg, dist;

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ // Statement 2 (function call)**

printf("Average: %f, Distance: %f", avg, dist);

return 0;

}

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ { //statement 1 (function header)**

int i, sum = 0, sumOfSquares = 0;

for(i=0; i<m; i++) {

sum += a[i];

sumOfSquares += a[i]\*a[i];

}

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //Statement 4 (Average Calculation)**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ //Statement 5 (Distance Calculation)**

}

The above code is made of the main and stat functions. stat receives one integer array of size m, and returns the average value of the array and the distance of the array as follows:

* The average value is defined as avg = (a[0]+a[1]+a[2]+…+a[m–1])/m
* The distance is defined as

distance =

37) Statement 1 should be:

1. void stat(int x[], int m, double \*avg)
2. void stat(int x[], int m, double \*dist)
3. void stat(int x[], int m, double \*avg, double \*dist)
4. void stat(int x[], double \*avg, double \*dist)

38) Statement 2 should be:

1. stat(x, 3, avg, dist);
2. stat(x, 3, &avg, \*dist);
3. stat(x[3], 3, &avg, &dist);
4. stat(x, 3, &avg, &dist);

39) Statement 4 should be:

1. \*avg = (double) sum/m;
2. &avg = (double) (sum/m);
3. avg = sum/(double) m;
4. \*avg = \*sum/m;

40) Statement 5 should be:

1. \*dist = sqrt(sum);
2. dist = sqrt(sumOfSquares);
3. &dist = sqrt(sum);
4. \*dist = sqrt(sumOfSquares);

41) Statement 6 should be:

1. printf("Average: %d, Distance: %f", &avg, &dist);
2. printf("Average: %f, Distance: %d", \*avg, \*dist);
3. printf("Average: %f, Distance: %f", avg, dist);
4. printf("Average: %d, Distance: %d", avg, dist);

**The next 2 questions are based on the following code fragment:**

char \*token, delim[] = "!";

char str[] ="!Thisis!me!";

int count=0, total=0;

token = strtok(str, delim);

while(token != NULL && count < 2){

count++;

total = total + strlen(token);

token = strtok(str, delim);

}

42) What will be the value of total after the while loop?

1. 12
2. 8
3. 2
4. 6

43) What will be the value of token after the while loop?

1. me
2. Thisis
3. Thisisme
4. !

**The next 2 questions are based on the following code fragment:**

char line[20],text[3][7]={"ICS103","428","Great!"};

int i, j, count=0, k=0;

for(i=0; i<3 ;i++)

for(j=0; j < 3; j++)

if(isalpha(text[j][i])){

line[k] = text[i][j];

k++;

}

else

count++;

line[k]='\0';

puts(line);

44) What is the output of the above code fragment?

1. IS28Gr
2. IS48Ge
3. IS48Gr
4. IS28Ge

45) What will be the final value of count after executing the code?

1. 4
2. 2
3. 3
4. 9

46) Which of the following codes will print the pattern shown below?

\*\*

\*\*\*

\*\*\*\*

\*\*\*\*\*

|  |  |
| --- | --- |
| A. for(i=1; i <= 5; i++) {  for(m=0; m <= i; m++)  printf("\*");  printf("\n");  } | B. for(i=1; i < 5; i++)  for(m=0; m <= i; m++)  printf("\*");  printf("\n"); |
| C. for(i=1; i < 5; i++) {  for(m=0; m <= i; m++)  printf("\*");  printf("\n");  } | D. for(i=1; i <= 5; i++)  for(m=0; m <= i; m++) {  printf("\*");  printf("\n");  } |

47) Consider the following code fragment

int A[3][3]={{1},{4,5},{0}};

int i,j,sum;

for(j = 0; j < 3; j++) {

sum = 0;

for (i = 0; i < 3; i++)

if ( sum = j )

sum = sum + A[j][i];

printf("%d ", sum);

}

What is the output of the above code?

1. 0 9 0
2. 5 0 0
3. 0 5 0
4. 0 1 2

48) Which statement is true for searching algorithms (linear search and binary search) covered in class.

1. Binary search is better than linear search
2. It depends of the number and type of the items to sorted
3. Selection search is better than binary search
4. None of the above

49) Consider the following recursive binary search function covered in class:

int binary\_search (double x[], int low, int high, double target) {

int middle;

if (low > high) /\*base case1:target not found\*/

return -1;

middle = (low + high)/2;

if ( Statement 1 )

return (middle); /\*base case2:target found\*/

else if ( Statement 2 )

return binary\_search(x, middle+1,high,target);

else

return binary\_search(x, low, middle-1,target);

}

Statement 2 is

1. If (middle < target)
2. If (middle <= target)
3. If (middle < x[target])
4. If (x[middle] < target)

50) Consider the following array:

int x[7]={10, 13, 23, 25, 30, 35, 50};

and consider searching for the target value **0** by the **recursive binary search** covered in the class. The last 2 values of the array compared to the **target** value are:

1. 23 and 10
2. 25 and 30
3. 13 and 10
4. 25 and 13