Information and Computer Science Department

2011/2012 Summer Semester (Term 113)

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

**MIDTERM EXAM KEY**

Sunday, July 8th 2012, 07:00 PM – 09:00 PM

120 MINUTES

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Name: | K | | | | | | | | |
| ID: |  |  |  |  |  |  |  |  |  |

Circle your section:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1. Bagais   Section 01 | R. Putu  Section 02 | M. Aslam  Section 03 | M. Said  Section 04 | 1. Salahdin   Section 05 |
| 9:20 – 10:10 | 9:20 – 10:10 | 10:30 – 11:20 | 10:30 – 11:20 | 10:30 – 11:20 |

|  |  |  |
| --- | --- | --- |
| Question No. | Maximum Score | Score |
| 1 | 20 |  |
| 2 | 8 |  |
| 3 | 8 |  |
| 4 | 8 |  |
| 5 | 8 |  |
| 6 | 8 |  |
| 7 | 20 |  |
| 8 | 20 |  |
| Total | 100 |  |

printf(“Good Luck”);

## Question 1 (20 points):

Choose the correct answer for each of the following questions:

1. What is the code that produces the following formatted output?

4

-

5

0

4

3

8

.

.

8

5

0

4

3

.

8

1. double x = 8.337;

int y = -54;

printf(“%6.2f%4d”, x, y);

1. double x = 8.34;

int y = 54;

printf(“%4d%6.3f”, y, x);

1. **double x = 8.34;**

**int y = -54; (C)**

**printf(“%6.3f%4d”, x, y);**

1. double x = 8.337;

int y = -54;

printf(“%7.2f%4d”, x, y);

1. Consider this C code fragment:

int x;

scanf("%d",&x);

if(x > 3 ) {

if(x < 15){

if(x > 8)

printf("A");

else

printf("B");

}

else{

if ( x <= 0)

printf("C");

else

printf("D");

}

}

else

printf("E");

Which of the following statement is correct?

1. A is printed if the scanned value for x is in the range (3 – 15) not inclusive
2. **printf(“C”) will never be executed no matter what the scanned value of x is. (B)**
3. D is printed if the scanned value of x is greater than 0
4. B is printed when the scanned value for x is 2
5. Which statement is true about the following code:

char c = 'c';

switch(c) {

default:

case 'a': printf("a"); break;

case 'b': printf("b"); break;

}

* 1. this switch is illegal because only integers can be used in switch statement
  2. **when this code is executed, the letter a is printed (B)**
  3. this switch is illegal because default must come as the last case
  4. when this code is executed, nothing is printed out

1. Given the following code:

int i = 1;

do for(i = -5; i > 10; i++)

printf("ICS 103\n");

while(i == -5);

Which of the following is true:

* 1. The code will not compile because it is not correct
  2. ICS 103 will be printed 5 times when the code is executed
  3. **the code is fine but nothing is printed (C)**
  4. ICS 103 will be printed infinitely

1. The text in /\* \*/ is:
   1. A preprocessor directive
   2. **Ignored by the compiler (B)**
   3. A file name in fopen function
   4. none of the above
2. For any value of y, which of the following pair of statements are ***NOT*** equivalent:
   1. x++; x += 15 % 2;
   2. x = 15 < y < -35; x = 4 – (int) 4.8;
   3. x = pow(3, 2) / 2; x = 9 / (6.0 – 4);
   4. **double a = 7 / 2; double a = 3.5; (D)**
3. For these declarations:

int x; double y; char z;

which of the following is the correct way to scan values for x, y and z?

1. **scanf(“%c%lf%d”, &z, &y, &x); (A)**
2. scanf(“%d%lf%c”, &y, &z, &x);
3. scanf(“%c%d”, &x, &y, &z);
4. scanf(“%lf%d%c”, y, x, z);
5. For this function prototype:

int fun(int a, double b);

which of the following is the correct way to call the corresponding function?

1. int x = fun(3 / 5);
2. **double x = fun(4, sqrt(33)); (B)**
3. fun(10, 2.5, 6);
4. char c = fun();
5. Which of the following statements is true?
   1. continue statement prevent the loop from doing more iterations
   2. sentinel loops are the ones that runs for infinite number of times
   3. **any while loop can be written as a for loop (C)**
   4. do while loop always makes one more iteration than the while loop
6. Which of the following opens data.txt file for writing?
   1. fopen(data.txt, “w”);
   2. fopen(“data.txt”, “r”);
   3. fopen(data.txt, a);
   4. **fopen(“data.txt”, “w”); (D)**

## Question 2 (8 points):

What is the output of the following code fragment?

int k, m, p;

Write the output here:

**11 7 8**

p = 0;

for (k=5; k<=10; k+=3)

{

for(m=3; m<=6; m+=2)

++p;

p += 2;

2.5 each and add 0.5.   
-1 for each extra output

}

printf("%d %d %d\n", k, m, p);

**Question 3 (8 points)**:

Consider the following program. What will be the output for the different values of x inputted by a user?

#include <stdio.h>

int main() {

int x, sum=0;

printf("Enter a value for x: ");

scanf("%d", &x);

|  |  |
| --- | --- |
| Value of x typed by the user. | Program Output |
| 1 | **2** |
| 6 | **-6** |
| 5 | **10** |
| 9 | **9** |

switch(x) {

case 1:

case 2:

case 5: sum += x;

case 7: sum += x;

case 8: break;

case 9: sum += x;

break;

default : sum –= x;

}

printf("%d\n", sum);

return 0;

}

**2.0 each**

**Question 4 (8 points)**:

Given the following program:

|  |
| --- |
| 7  5  9  8  6 |
| file1.txt |

#include <stdio.h>

int main(){

FILE \*infile, \*outfile;

int ct = 0, num;

double s = 0, mean;

infile = fopen("file1.txt", "r");

outfile = fopen("file2.txt", "w");

while(fscanf(infile, "%d", &num)!=EOF){

s += num;

ct++;

}

7.00

9

8

mean = s/ct;

fprintf(outfile, "%0.2f\n", mean);

rewind(infile);

while(fscanf(infile, "%d", &num)!= EOF){

2.5 each and add 0.5. -1 for each extra line

if(num > mean)

fprintf(outfile, "%d\n", num);

}

fclose(infile);

file2.txt

fclose(outfile);

return 0;

}

If the content of **file1.txt** is as shown, write the content of **file2.txt** after running the program

**Question 5 (8 points):**

Consider the following output.

AAAAAAAAAA

AAAAAAAA

AAAAAA

AAAA

AA

|  |  |
| --- | --- |
| Which one of the following code fragment will produce the above output? (Tick one box) | |
| int k, n;  for(n = 5; n>=1; n--){  for(k=1; k<=2\*n; k++)  printf("A");  printf("\n");  } | int k, n;  for(n = 1; n<=5; n++){  for(k=1; k<=2\*n; k+=2)  printf("A");  printf("\n");  } |
| int k, n;  for(n = 5; n>=1; n--){  for(k=1; k<=n; k+=2)  printf("A");  printf("\n");  } | int k, n;  for(n = 1; n<=5; n++){  for(k=1; k<=n; k+=2)  printf("A");  printf("\n");  } |

**Question 6 (8 points)**

Consider the following code fragment:

int k=15, p=1, n;

while (k >= 2){

n = -k;

while (n <= k){

p \*= n;

n += k/3;

}

k = k - 3;

}

printf("%d\n", p);

|  |  |
| --- | --- |
| Which of the following code fragment is equivalent to the above code? (Tick one box) | |
| int k, p = 1, n;  for (k=2; k<=15, k+=3)  for (n= -k; n<=k; n+=k/3)  p \*= n;  printf("%d\n", p); | int k, p = 1, n;  for (k=15; k>=2; k-=3)  for (n= -k; n<= k; n+=k/3)  p \*= n;  printf("%d\n", p); |
| int k, p = 1, n;  for (k=2; k<=15, k+=3){  for (n= -k; n<=k; n+=k/3)  p \*= n;  k -= 3;  }  printf("%d\n", p); | int k, p = 1, n;  for (k=15; k>=2; k-=3){  for (n= -k; n<=k; n+=k/3)  p \*= n;  k -= 3;  }  printf("%d\n", p); |

**Question 7 (20 points)**

Write a complete interactive C program that displays the following menu:

1. **Slope of line joining two points**
2. **Distance between two points (cm)**

**Enter your choice**

The program then reads the choice. If the choice is invalid the program displays an error message and terminates; otherwise the program prompts for and reads the coordinates of two points (x1, y1) and (x2, y2) and then it either displays the slope or the distance between the two points based on the user’s menu selection.

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

Sample run s of the program:

|  |  |
| --- | --- |
| distance.jpg | line.jpg |

**#include <stdio.h>**

**#include <math.h> //(1) includes and program structure**

**int main(void){**

**int choice;**

**double x1, y1, x2, y2, slope, distance; //(1) declerations**

**printf("1. Slope of line joining two points\n"); // (1) menu**

**printf("2. Distance between two points\n"); //**

**printf("\nEnter your choice\n"); //**

**scanf("%d", &choice); // (1) scanf**

**if(choice != 1 && choice != 2) // (4) outer if-structure**

**printf("Error: Invalid choice\n"); // (1)**

**else{**

**printf("Enter the coordinates of point1: "); // 2 points**

**scanf("%lf%lf", &x1, &y1); //**

**printf("Enter the coordinates of point2: "); //**

**scanf("%lf%lf", &x2, &y2); //**

**if(choice == 1){ // (4) inner if**

**slope = (y2 - y1)/ (x2 - x1); // (1)**

**printf("slope = %.2f\n", slope); // (1)**

**}**

**else{**

**distance = sqrt(pow(x1-y1, 2) + pow(x2 - y2, 2)); //(2)**

**printf("distance = %.2f cm\n", distance); //(1)**

**}**

**}**

**system("pause");**

**return 0;**

**}**

**Question 8 (20 points)**

Each line of a text file **patients.txt** consists of a patient ID, the weight of the patient in kilograms, and his height in meters.

|  |
| --- |
| 567789 50.5 1.75  450088 82.4 1.63  326517 55.5 1.52  864321 70.0 1.37  256675 60.5 1.93 |

Write a complete C program that reads data from the **patients.txt** file and outputs on a file **results.txt** the classification of each patient as: under-weight, normal-weight, over-weight, or obese based on the BMI index of that patient:

BMI index = (weight in kilograms) / (height in meters)2

|  |  |
| --- | --- |
| **BMI index** | **Classification** |
| < 18.5 | under-weight |
| >= 18.5 and < 25 | normal- weight |
| >= 25 and < 30 | over -weight |
| >= 30 | obese |

**Note:**

* Assume that the number of patients in the file is not known (i.e., your program must work for any number of patients).
* Your program must take care of all text file I/O errors.

Sample output in results.txt:

|  |
| --- |
| Patient ID Category  567789 under-weight  450088 obese  326517 normal-weight  864321 obese  256675 under-weight |

**#include <stdio.h>**

**int main(void){**

**int ID;**

**double weight, height, bmi; // (1) declarations and program structure**

**FILE \*infileptr, \*outfileptr; // (2)**

**infileptr = fopen("patients.txt", "r"); // (2)**

**if(infileptr == NULL){ //(1)**

**printf("Error in opening patients.txt\n"); //**

**system("pause"); //**

**exit(1); //**

**}**

**outfileptr = fopen("results.txt", "w"); // (2)**

**fprintf(outfileptr,"Patient ID\t\tCategory\n\n"); //(1)**

**// (4) loop**

**while(fscanf(infileptr,"%d%lf%lf", &ID, &weight, &height)!=EOF){**

**bmi = weight / (height \* height); // (2)**

**// (4) if-structure**

**if(bmi < 18.5)**

**fprintf(outfileptr, "%d\t\t\tunder-weight\n", ID);**

**else if(bmi >= 18.5 && bmi < 25)**

**fprintf(outfileptr, "%d\t\t\tnormal-weight\n", ID);**

**else if(bmi >= 25 && bmi < 30)**

**fprintf(outfileptr, "%d\t\t\tover-weight\n", ID);**

**else**

**fprintf(outfileptr, "%d\t\t\tobese\n", ID);**

**}**

**fclose(infileptr); // (1)**

**fclose(outfileptr); //**

**system("pause");**

**return 0;**

**}**