CS101 Spring 2012
Final Exam (120 minutes)
Friday, May 4th, 2012

Name: ____________________________________________ Section: ________________________

READ and complete the following:

• Bubble your Scantron only with a No. 2 pencil.

• On your Scantron (shown in the figure below), bubble:

  1. Your Name
  2. Your NetID
  3. Form letter "A"
  4. Bubble the corresponding 3-digit code (shown below) for your lab section on your Scantron.

<table>
<thead>
<tr>
<th>Time</th>
<th>Monday</th>
<th>Tuesday</th>
<th>Wednesday</th>
<th>Thursday</th>
</tr>
</thead>
<tbody>
<tr>
<td>9:00-10:50</td>
<td>101</td>
<td>104</td>
<td>107</td>
<td>109</td>
</tr>
<tr>
<td>11:00-12:50</td>
<td>102</td>
<td></td>
<td>110</td>
<td></td>
</tr>
<tr>
<td>1:00-2:50</td>
<td></td>
<td>105</td>
<td></td>
<td>111</td>
</tr>
<tr>
<td>3:00-4:50</td>
<td>103</td>
<td>106</td>
<td>108</td>
<td>112</td>
</tr>
</tbody>
</table>

• No electronic devices, books, notes, or cheat sheets are allowed while taking this exam.

• Please fill in the most correct answer on the provided Scantron sheet.

• We will not answer any questions during the exam.

• Each question has only ONE correct answer.

• You must stop writing when time is called by the proctors.

• Hand in both these exam pages and the Scantron.

• DO NOT turn this page UNTIL the proctor instructs you to.
STOP!!! Please ensure that your NETID is bubbled in correctly on your scantron sheet. If it is not, you will AUTOMATICALLY be deducted 5 points from your final score.
1. What is the output of the following program that compiles and runs without errors?

```
#include <stdio.h>

int x = 1; /* Global Variable */
int y = 2; /* Global Variable */

void swap(int x, int y)
{
    int tmp;
    tmp = x;
    x = y;
    y = tmp;
}

void main(void)
{
    int x = 10;
    int y = 20;

    swap(x, y);

    printf("x: %i,y: %i\n", x, y);
}
```

(a) x: 1, y: 2
(b) x: 20, y: 10
(c) x: 10, y: 20
(d) x: 2, y: 1
2. The following program compiles and runs without errors, what is the output?

```c
#include <stdio.h>

void accumulate(int * x, int * y)
{
    *x = *y + *x;
}

void main(void)
{
    int x = 20;
    int y = 10;

    accumulate(&x, &y);
    accumulate(&x, &y);
    accumulate(&x, &y);

    printf("x: %i, y: %i\n", x, y);
}

(a) x: 10, y: 10

(b) x: 20, y: 30

(c) x: 50, y: 10

(d) x: 10, y: 30
3. How many distinct variables does the following program have?

```c
#include <stdio.h>

int x = 10;
int y = 20;
int z = 30;

void swap(int *x, int *y)
{
    int tmp;
    tmp = *x;
    *x = *y;
    *y = tmp;
}

int acc(int x)
{
    int sum = 0;
    sum += x;
    z = sum;
    return sum;
}

void main(void)
{
    double z = 1.0;
    swap(&x, &y);
    x = acc(y);

    printf("%i %lf\n", x, z);
}
```

(a) 5
(b) 9
(c) 11
(d) 3
4. The following program compiles and runs without errors. What is the output of the program?

```c
#include <stdio.h>

void main(void)
{
    int arr[5] = {2, 4, 6, 8, 10};
    int *ptr = arr;

    printf("%i ", *ptr + 1);
    printf("%i\n", *(ptr + 1));
}
```

(a) 4 4  
(b) 3 5  
(c) 3 4  
(d) 4 6

5. The following C program compiles and runs without errors. What is the output produced by this program?

```c
#include <stdio.h>

void main(void)
{
    int x[2] = {10, 20};
    int *ptrX;

    ptrX = x;
    (*ptrX)++;
    --x[1];
    --(*ptrX);

    printf("(%i, %i)\n", x[0], x[1]);
}
```

(a) (9, 19)  
(b) (9, 18)  
(c) (10, 18)  
(d) (10, 19)
6. The following program compiles and runs without errors. What would be the output of the following program?

```c
#include <stdio.h>

void swap1(int a, int b) {
    int temp = a;
    a = b;
    b = temp;
}

void swap2(int a[ ], int b[ ]) {
    int temp = a[0];
    a[0] = b[0];
    b[0] = temp;
}

void swap3(int *a, int *b) {
    int temp = *a;
    *a = *b;
    *b = temp;
}

void main(void) {
    int a[2] = {1, 2};
    int b[2] = {4, 5};
    swap1(a[0], a[1]);
    swap2(a, b);
    swap3(&b[0], &b[1]);
    printf("%i, %i, %i, %i \n", a[0], a[1], b[0], b[1]);
}
```

(a) 4, 1, 5, 2
(b) 4, 2, 5, 1
(c) 4, 5, 2, 1
(d) 4, 5, 1, 2
7. Which of the following commands can be used to declare two pointer variables to two objects of type float?

(a) float *fptr1, fptr2;

(b) float *fptr1, *fptr2;

(c) float *(fptr1, fptr2);

(d) None of the above.

8. The following program compiles and runs without errors. When it is run var1 is stored in memory address 2000, and var2 is stored in memory address 2006.

```c
#include <stdio.h>

void main(void)
{
    int var1 = 5;
    int var2 = 20;

    int *ptr1 = &var1;
    int *ptr2 = &var2;
    int *ptr3 = ptr2;
}
```

What value was assigned to ptr3?

(a) 20

(b) 2000

(c) 2006

(d) 5
9. Which one of the following is an incorrect C programming language statement? That is, which line would generate a compile error from gcc? Note that the line numbers are not part of the code.

1. #include <stdio.h>
2. void main(void)
3. {
4.   int x = 1, y = 2;
5.   int * ptr =&y;
6.   y = ++x;
7.   y = *ptr;
8.   ptr = &x;
9.   y = ptr->x;
10. }

(a) Line 5
(b) Line 7
(c) Line 8
(d) Line 9
The following C program compiles and runs without errors. What is the output of this program?

```c
#include <stdio.h>

void func(int *x, int y, int z)
{
    int i;

    for(i = 0; i < z; i++)
        x[i] = x[i] + (i + 1) * y;
}

void main(void)
{
    int a[4] = {1, 2, 3, 4};
    int b[3] = {3, 2, 1};
    int *c;

    func(&a[0], a[1], 3);
    printf("%i %i %i\n", a[0], a[1], a[2]);

    func(&b[1], b[2], 2);
    printf("%i %i %i", b[0], b[1], b[2]);
    c = &a[1];
}
```

(a) 3 6 9
   3 3 3

(b) 3 6 9
    2 3 4

(c) 3 3 3
    2 3 4

(d) 2 3 4
    3 6 9
Suppose we need to manage a number of peanut butter and jelly sandwich orders for a party. We are not expecting more than 500 guests, so we will declare an array of 500 sandwich structures as follows:

(Use the following structure definition and array declaration to answer the next two questions.)

```c
typedef struct
{
    char type[25];
    char jelly[25];
    char bread[25];
    int layers;
} pbj;

pjb sandwiches[500];
```

11. Which line of code below, will assign "smooth" to the **type** field of the first element of the array named `sandwiches`?

   (a) `strcpy(sandwiches[0].type, "smooth")`;

   (b) `sandwiches[0].type = "smooth"`;

   (c) `strcpy(sandwiches.type[0], "smooth")`;

   (d) `sandwiches[1].type = "smooth")`;

12. Which line of code below, will assign 7 to the **layers** field of the seventh element of the array named `sandwiches`?

   (a) `strcpy(sandwiches[6].layers, 7)`;

   (b) `sandwiches[6].layers = 7`;

   (c) `sandwiches[6].layers -> 7`;

   (d) `sandwiches[7].layers.7`;
Answer the next two questions using the following structure definition and array declaration:

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>

#define teamSize 30

typedef struct
{
    char name[40];
    char team[20];
    char position[15];
    int number;
    double salary;
} Player;

Player roster[teamSize];
```

13. Assuming all variables have been initialized correctly, which for-loop will assign "Tigers" to the team field of every entry in the array `roster`?

(a) for(i = 0; i<teamSize; i++)
    roster[i].team = "Tigers";

(b) for(i = 0; i<teamSize; i++)
    strcpy(roster[i].team,"Tigers");

(c) for(i = 0; i<teamSize; i++)
    roster->team = "Tigers";

(d) for(i = 0; i<teamSize; i++)
    strcpy(roster[i]->team,"Tigers");

14. What line of C code will assign 19 to the number field of the 20th element in the array `roster`?

(a) roster[20].number = 19;

(b) roster[20][number] = 19;

(c) roster[19].number = 19;

(d) roster[19]->name = 19;
The next question uses the following structure that is part of a music store’s inventory tracking software.

typedef struct {
    int    cdno;    /* cd number */
    char   title[30];
    char   artist[20];
    int    year;
    int    num_tracks; /* number of tracks */
    int    quantity;
    float  price;
    float  sales[12]; /* cd sales ($US) over 12 months */
} Album;

15. Fill in the blank below for the program fragment that computes the net asset value for the music store. That is to say, complete the code below that assigns the product of the quantity and price fields of each element in the array albums to the variable value. The code then prints out the variable networth (the summation of all album values). You can assume that all fields of each element in the array albums have been initialized. You may also assume that num_albums has been initialized correctly.

float assetsValue(Album *album)
{
    float value;

    value = _________________________;
    return value;
}

void add(Album albums[], int num_albums)
{
    int i;
    float networth = 0.00;

    for(i = 0; i < num_albums; i++)
        networth += assetsValue(&albums[i]);

    printf("%7.2f", networth);
}

(a) (album->quantity)*(album->price)

(b) (album[i].quantity)*(album[i].price)

(c) (album.quantity)*(album.price)

(d) ((*album)->quantity)*((*album)->price)
16. Suppose that we are given the following structure and incomplete comparison function:

```c
#include <stdio.h>

typedef struct
{
    int UIN;
    char name[20];
} student;

int compareUinAsc(student *l, student *r)
{
    /* code for comparing UINs goes here */
}
```

Assuming that the above comparison function is used as the fourth argument of the function `qsort()`, which line of code below correctly completes the comparison function such that an array of data type `student` is sorted by the `UIN` field in ASCENDING order?

(a) return *l.UIN - *r.UIN;

(b) return l->UIN - r->UIN;

(c) return *r.UIN - *l.UIN;

(d) return r->UIN - l->UIN;
17. What is the correct way to call the function \texttt{qsort()}, so that it sorts a non-empty array called \texttt{arr} that has 1000 elements of data type \texttt{double}.

(a) \texttt{qsort(arr[],1000,double,comparefunc)};

(b) \texttt{qsort(arr,1000,sizeof(double),comparefunc)};

(c) \texttt{qsort(arr[],1000,sizeof(arr[0]),comparefunc)};

(d) \texttt{qsort(arr,sizeof(double),1000,comparefunc)};

18. Which of the following statements about \texttt{fopen()} is NOT true?

(a) When the mode "r" is used, and the specified file does not exist, \texttt{fopen()} returns NULL.

(b) When the mode "w" is used, and the specified file does not exist, \texttt{fopen()} returns NULL.

(c) When the mode is "a" is used, and the specified file does not exist, \texttt{fopen()} creates a file.

(d) When the mode is "a" is used, and the specified file exists, \texttt{fopen()} writes at end of the file.
19. Consider the following C code below. The line numbers are not part of the code.

```c
#include <stdio.h>
void main(void)
{
    FILE *fileIn;
    int value;
    int total;
    fileIn = fopen("infile.dat", "r");
    while (fscanf(fileIn, "%i", value))
        total += value;
    fclose(fileIn);
}
```

Which of the following statements is NOT True?

(a) Since fopen() may return NULL at line 8, a proper error handling is necessary.

(b) The condition at line 10 would be problematic if end of file (EOF) is encountered.

(c) At line 10, value should be replaced by &value.

(d) fclose(fileIn); must be replaced with free(fileIn); .
20. Which of the following will correctly open a file with the name data.txt for writing.

(a) `FILE *fin;`  
`FILE = fopen("data.txt","w");`  

(b) `FILE *fin;`  
`*fin = fopen("data.txt","w");`  

(c) `FILE *fin;`  
`fin = fopen("data.txt","r");`  

(d) `FILE *fin;`  
`fin = fopen("data.txt","w");`  

21. Complete the following C program by filling in the blank to read in

The Beatles; The Long and Winding Road; Let It Be... Naked; 214

into ‘artist’, ‘song_name’, ‘album_title’ and ‘track_length’ defined below.

```c
#include <stdio.h>

void main(void) {

    char artist[40];
    char song_name[40];
    char album_title[40];
    int track_length;

    printf(" Please enter the artist, song_name, album_title, and track_length\n");
    scanf("%[^;] %[^;] %[^;] %i",artist,song_name, album_title,&track_length);

    printf(" %s \n %s \n %s \n %i\n",artist,song_name, album_title,track_length);
}
```

(a) " %s %s %s %i"
(b) " %s; %s; %s; %i"
(c) " %[^;] %[^;] %[^;] %i"
(d) " %[^;]; %[^;]; %[^;]; %i"
What is the output of the following program which compiles and runs without errors?

```
#include <stdio.h>

int max(int x, int y);
int min(int x, int y);
int rec(int x, int y);

void main(void)
{
  printf("%i", rec(512, 44));
}

int max(int x, int y)
{
  if(x > y)
    return x;
  else
    return y;
}

int min(int x, int y)
{
  if(x < y)
    return x;
  else
    return y;
}

int rec(int x, int y)
{
  int a = max(x, y);
  int b = min(x, y);
  int c = a % b;
  if(c == 0)
    return b;
  else
    return rec(b, c);
}
```

(a) 2
(b) 4
(c) 8
(d) 16
Complete the following recursive function named `find` by filling in the blank. The function named `find` finds a value in an array and returns the index of the value in the array. If the value is not found in the array the function returns -1.

```c
#include <stdio.h>

int find(int arr[], int value, int left, int right)
{
    int midpt = (left+right)/2;
    if (left > right)
        return -1;
    if (arr[midpt] == value)
        return midpt;
    else if (arr[midpt] < value)
        return ____________________________;
    else
        return find(arr,value, left,midpt-1);
}

void main(void)
{
    int arr[] = {4,5,6,9,11};
    printf("%i \n", find(arr,4,0,4)); /* displays 0 */
    printf("%i \n", find(arr,9,0,4)); /* displays 3 */
    printf("%i \n", find(arr,10,0,4)); /* displays -1 */
}

(a) find(arr,value, right, midpt+1)
(b) find(arr,value, midpt+1,right)
(c) find(arr,value, left,midpt+1)
(d) find(arr,value, right,midpt-1)
```
The following program compiles and runs without errors. What is the output of the program?

```c
#include <stdio.h>

void rec(char str[], int n)
{
    if(n < 1)
        return;

    printf("%c", str[n - 1]);
    rec(str, n - 1);
}

void main(void)
{
    char str[] = "0123456789"
    rec(str, 5);
}

(a) 01234

(b) 56789

(c) 98765

(d) 43210
25. The following program compiles and runs without errors. What is the output of this program?

```
#include <stdio.h>

int power(int x, int n)
{
    if (n == 1) /* Voom! */
        return x;

    return power(x, n/2) * power(x, n-n/2);
}

void main(void)
{
    printf(" %i ", power(3, 3));
}

(a) 3
(b) 9
(c) 27
(d) 81
```
Extra Credit

Answering, the questions below, correctly will add points to your exam total. Answering incorrectly or not answering will not add points to your exam total.

26. What is the result when you try to compile and run this program?

```c
#include <stdio.h>

int x = 2;
int y = 4;

void func(int y)
{
    x = y*y;
}

void main(void)
{
    func(2);
    printf("%i \n",x);
}
```

(a) The program prints out 2
(b) The program prints out 4
(c) The program prints out 16
(d) A compile error occurs since x and y were not initialized correctly inside func and main.

27. Which of the problems below could be solved using recursive function(s) in the C programming language?

(a) The Fibonacci numbers: \( f(n) = f(n-1) + f(n-2) \) where \( f(0) = 0 \) and \( f(1) = 1 \)
(b) The factorial: \( n! = n(n-1)! \) where \( 0! = 1 \)
(c) The Towers of Hanoi puzzle
(d) All of the above