PRACTICE EXAM 02 for
CS16 Midterm Exam 2
E02, 09F, Phill Conrad, UC Santa Barbara
Actual exam scheduled for: 11/18/2009

Name: _____

Umail Address: _____@ umail.ucsb.edu

Circle Lab section: 8AM 10AM 11AM noon

Link to Printer Friendly PDF Version

Please write your name **only** on this page. That allows me to grade your exams without knowing whose exam I am grading.

This exam is **closed book, closed notes**, **closed mouth, cell phone off**,
except for:

- You are permitted **one sheet of paper** (max size 8.5x11") on which to write notes
- These sheets will be collected with the exam, and might not be returned
- Please write your name on your notes sheet

There are 100 points worth of questions on the exam, and you have 50 minutes to complete the exam.

A hint for allocating your time:

- if a question is worth 10 points, spend no more than 5 minutes on it
- if a question is worth 20 points, spend no more than 10 minutes on it
- etc.

---

1. (10 pts) Write the definition of a C function that takes two parameters—an array of integers, and a second integer indicating the size of the array.

   The function should be called countMax. The function should count how many elements of array have the maximum value in the array.

   For example, if the array contains {10,40,15,40,15,40,20} the value returned is 3, because the maximum value is 40, and 40 occurs three times in the array.

   If the array contains {40,34,20,70,40}, then the value returned is 1, because 70 occurs once in the array.

   If you use any helper functions—for example, if you use function calls to `countOccurences(a,n)`, or `maxValue(a,n)` —then include definitions of those functions also.

   Write ONLY the function definition—for this question, I do NOT want a complete C program, so do NOT include any extraneous stuff such as #include <stdio.h> or a main function. (See solution)

2. (10 pts) Write the definition of a C function that takes two parameters—an array of integers, and a second integer indicating the size of the array.

   The function should be called sumPositive. The function should return the sum of all the integers in the array that are greater than 0.

   For example, if the array contains {-3,10,-5,20} the value returned is 30, because the positive elements are 10 and 20, and their sum is 30.

   If you use any helper functions—for example, if you call a function called `isPositive(i)`—then include definitions of those functions also.

   Write ONLY the function definition—for this question, I do NOT want a complete C program, so do NOT include any extraneous stuff such as #include <stdio.h> or a main function. (See solution and an alternate correct solution)

3. (10 pts) Write the definition of a C function that takes two parameters—an array of integers, and a second integer indicating the size of the array.

   The function should be called averageOfIntArray. The function should return the average of all the integers in the array.

   If you use any helper functions—for example, if you call a function `sumOfArray(a,n)`—then include definitions of those functions also.

   Write ONLY the function definition—for this question, I do NOT want a complete C program, so do NOT include any extraneous stuff such as #include <stdio.h> or a main function. (See solution and compare it with this incorrect solution)

   Hints:

   - Note that the average is not necessarily a integer—for example the average of a two element array of integers containin 3 and 4 is 3.5—so choose the return type appropriately.
   - Remember to be careful about type—int vs. double—when dividing.
   - If a variable—say, x—is an integer, you can convert it to a double in several ways:
     - Using a typecast
     - Assigning its value to another variable of type double
     - Multiplying by 1.0

4. (10 pts) Given the definition of a struct for Point:

```
struct Point {
  double x;
  double y;
};
```

Write the definition of a function that matches this prototype:

```
int countPointsCloserThan(struct Point p,
                          struct Point *others,
                          double distance, int numPts);
```

This function counts the number of points in an array of points called `others` that are closer to the point `p` than the value of the parameter `distance` (i.e. the distance between p and the point from others is strictly less than `distance`. The parameter `numPts` indicates the size of the array `others`)

Assume that the function matching the prototype below has already been defined, so that you can —and should—use a call to that function to determine the distance between points.

```
double distanceBetween(struct Point p1, struct Point p2);
```

Notes:

- You will lose points if you recreate the distance formula from scratch rather than using a call to the distanceBetween function.
- You do NOT need to provide the definition of `distanceBewteen`—and *will not receive any credit for doing so (not even partial credit.)*

See a sample solution

5. Here are several function definitions—doIt1, doIt2, etc..

In each case, **indicate what would be printed when the function is called.**

If the value is unpredictable, because the array element is outside the bounds of the array, please write the word "**undefined**" (solution)

a. (3 pts)

```
void doIt1()
{
  int a[] = {10,20,30,40};
  printf("%d\n",a[3]);
}
```

b. (3 pts)

```
void doIt2()
{
  int a[5] = {0};
  printf("%d\n",a[5]);
}
```

c. (3 pts)

```
void doIt3()
{
  int a[5] = {1};
  printf("%d\n",a[4]);
}
```

d. (3 pts)

```
void doIt4()
{
  int a[5];
  int i;
  for (i=0; i<5; i++)
    a[i] = 5-i;
  printf("%d\n",a[2]);
}
```

6. (25 pts) Together with this exam, there is a program (on a separate handout).

Assuming each of the expressions below appeared in this program, indicate the type they would have, or write error if the expression is not valid, e.g.

- dereferencing something with * or -> that isn't a pointer
- a reference to a struct member that doesn't exist (e.g. d.foo where there is no foo)

The first few are done for you as an example.

Hints--for full credit:

- don't write *pointer to character*; instead, write **char ***
- don't write *address of int*; instead, write **int ***
- don't write *address of int* * or *address of pointer to int*, instead write **int ****

See solution

Expression	Type		Expression	Type
a	int		f->radius	
&a	int *		f->center.x	
*a	error		f->center->y	
*b	int		&g	
e	struct Circle		*h	
b			&(h->x)	
& b			*(h.y)	
c			i.m	
*c			j.d	
d			i->y	
&d			j->m	
*e			(*j).m	
e.x			argc	
e.radius			argv[0]	
f.y			argv[0][0]	

7. There is a Unix command you can type that will give you information about the permissions that apply to accessing files in a format such as `rwxr-xr-x`

   a. (2 pts) What is this command?
   Give the full command, exactly as you would type it at the Unix command prompt.

   (Solution to this quesiton, and all the rest of the parts of question 7)

   b. (1 pts) Suppose you use the command from question 7, and you see that the current permission mode for a file is `rw-r--r--`.

   Does the owner of this file have permission to make changes to the file?

   c. (1 pts) If the mode is `rw-r--r--`, can the owner of this file run this file as a program?

   d. (1 pts) If the mode is `rwxr--r--`, can users other than the owner of the file make changes to it?

   e. (2 pts) If you want to change the permissions of a file, there is a unix command you can use to do this of the form *commandName octalNumber filename*.

   Give this command (the octal number form) for changing the permissions of the file `foobar` to have the permission string `rwxr-xr-x`

   f. (2 pts) If you use the same command as the previous problem but with the octal number 711, what would the resulting rwx format string look like?

   g. (2 pts) In that same scenario (changing the file foobar to have permission mode corresponding to 711), who would have permission to execute the file?

   Select one:

   - all users,
   - no users
   - only the owner of the file?

   h. (2 pts) There is a Unix command you can type that will give a brief summary of whether a file contains executable machine instructions, ASCII text, or something else. How would you run that command on the file with filename `foobar`?

8. (10 pts) Briefly explain the following statement:

   "There are two senses in which a file can be executable—one having to do with the contents of the file, and another having to do with the chmod command."

   (No solution given—but if you review what we did in lab07a, you'll understand...)

End of Exam

Total Points: 100

PRACTICE EXAM 02 for
CS16 Midterm Exam 2

Extra Handout

Program for question about types

```
// types.c  Code for exam question, 11/15/2009
// P. Conrad for CS16, 09F, UCSB


#include <stdio.h>


struct Point {
  double x;
  double y;
};


struct Date {
  int d;
  int m;
  int y;
};


struct Circle {
  struct Point center;
  double radius;
};


int main(int argc, char *argv[])
{
  int a;
  int *b;
  double c;
  double *d;
  struct Circle e;
  struct Circle *f;
  struct Point g;
  struct Point *h;
  struct Date i;
  struct Date *j;


  // Program does no useful work
  // It is just the basis of a homework assignment about types



  // Pretend there is useful code here, and then
  // answer questions about the types of various expressions
  // as if they appeared right here.


  return 0;


}
```