ECS 30

Practice Midterm Key

The questions in this document may have appeared in midterms in previous years when this class was taught by different instructors.

1. UNIX commands

   a. (10 points) You just printed something to printer `hexus`, and find that it's printing a bunch of garbage. What are the steps you take to cancel the print job? **Use lpq -Phexus to find the # of my print job and then use lprm -Phexus job# to remove the job from the queue**.

   b. (5 points) What UNIX command would you type to see a list of all of the files (including the hidden files) in directory hw1? **ls -a hw1**

   c. (5 points) You just wrote and saved the file test2.c. Now you want to make sure that only you can read and write it. What UNIX command would you type to make test.c only readable and writeable by you? **chmod 600 test.c or chmod g-rw o-rw test.c**

   d. (5 points) You want to move all of your C source files (ending in .c) and header files (ending in .h) from your hw1 subdirectory to your mid1 subdirectory. What UNIX command should you type? **mv hw1/*.c hw1/*.h mid1**

   e. (5 points) You want to view the file test3.c a screen at a time. What UNIX command would you type? **more test3.c**

2. (50 points) Write a complete, warning-free, C program that lists all of the common factors of two positive integers. The program will prompt the user for the two numbers, and then list on one line all the factors that the numbers have in common. A factor of a number divides into that number with a remainder of zero. The program will continue to ask the user for more pairs of numbers until the first number entered is zero. Your prompts and formats should match that shown. User input is in **bold.**

```
Please enter two positive integers: 20 12
1 2 4

Please enter two positive integers: 60 120
1 2 3 4 5 6 10 12 15 20 30 60

Please enter two positive integers: 8 9
1

Please enter two positive integers: 0 230
Done.
```

Pts	
2	`#include <stdio.h>`
2	`int main()`
	`{`
3	`  int i, num, num1;`
1	`  do`
	`  {`
2	`    printf("Please enter two positive integers: ");`
5	`    scanf("%d%d", &num, &num1);`
3	`    if(num > 0)`
	`    {`
7	`      for(i = 1; i <= num; i++)`
12	`        if(num % i == 0 && num1 % i == 0)`
3	`          printf("%d ", i);`
3	`      printf("\n\n");`
	`    }`
	`  }`
4	`  while(num != 0);`
2	`  printf("Done\n");`
1	`  return 0;`
	`}`

3. (15 points) Consider the following recursive function. What does the function do as whole? (Do not describe what each line does.)

```
void pb(int n) {
   if (n != 0) {
         pb(n / 2);
         putchar('0' + n %2);
   }
}
```

It prints out the binary representation of n starting with the leftmost 1. For example, if n = 43 then pb would print 101011.

4. (20 points) Assuming that a,b,c, and ans are ints, and a is 2, b is 4, and c is 7 at the beginning of each statement, write on the line the value of ans. There are no syntax errors in these statements.

```
-2_ans = 2 * 4 / 7 - 2 + 3 - 4 % 7;
22_ans = (++a == --b) + 3 * c++;
_6_ans = (7 / 4 + 4 / 7) * 3 / 4.0 * 8 ;
_1_ans = 7 < 8 && 8 > 6 || 4 == 4 && !(6 < 5);
_8_ans = (4 == 4) + ((7 > 7) * 3 || 6 % 3) * 5 + !(6 < 5) * 7;
```

5. (25 points) Given the following series of if statements, provide the outputs for each X. Note that more than one printf can be executed for each X.

```
if (X > 20)
     printf("First ");
else
     if(X < 5)
          printf("Second ");
     else
          printf("Third ");
if(X == 20)
     printf("Fourth ");
else
     if(X < 19 || X > 25)
          printf("Fifth ");
if(X > 5 && X < 22)
     printf("Sixth ");
```

a.) (5 points) X = 0 __**Second Fifth**______
b.) (5 points) X = 5 __**Third Fifth** _______
c.) (5 points) X = 17 _**Third Fifth Sixth**_
d.) (5 points) X = 20 _**Third Fourth Sixth**_
e.) (5 points) X = 28 _**First Fifth**_________

(10 points) Given the following makefile. After I edit two.h, when I type make, which files will be changed by gcc? **two.o one.o whole.out**

```
whole.out : one.o, two.o, three.o
   gcc -o whole.out one.o two.o three.o

one.o : one.c whole.h one.h two.h
   gcc -c one.c

two.o : two.c two.h whole.h
   gcc -c two.c

three.o : three.c three.h
   gcc -c three.c
```

6. (22 points) Data representation
   a.) (8 points) Provide the hexadecimal representation for the following int that is presented as a binary number:

```
1100 1010 0101 1111 1000 1110 0111 1011
  C    A    5    F    8    E    7    B   (1 point each)
```

   b.) (8 points) Provide the binary representation for the following int that is presented as a hexadecimal number (to help the graders place a space between every four digits):

   D9AC3E6B **1101 1001 1010 1100 0011 1110 0110 1011 (1 point each)**

   c.) (6 points) Provide the decimal equivalent of the following unsigned char that is presented as a binary number:
   10101011 = 128 + 32 + 8 + 2 + 1 = **171**