HOMEWORK 3

Due: May 7, 2003

Beginning of Class

(100 pts)

Introduction

For this assignment you will write a program that implements a portion of the Unix **touch** command. In doing so, you will have a chance to work with many of the system calls studied in chapter 3 of the textbook, e.g., *utime, getpwuid, stat*. There may also be other system calls we've studied that you will want to use.

Specific Details

Write a program called mytouch that modifies the access and modification times of files. Read the manpage for touch (1). Implement the case with no flags (**touch filename**) first. Then add functionality by implementing code to handle the following options: -a –m –t.

Note that in the manpage it says the following. "A user with write access to a file, but who is not the owner of the file or a superuser can change the modification and access times of that file only to the current time. Attempts to set a specific time with **touch** will result in an error." You may not want to implement this functionality on the first attempt, but need to think about the modularity of your program in order to add this functionality later. It is required in the final version of your program.

For 5 points of extra credit, implement the –r option.

Program Documentation

Create a `README` file that documents your program. The format of this file is shown in the file **`~jusakd/cprogs/sample_outline.txt`**. You may copy that file and complete it.

Turning In Your Work

Run the ***script*** command to capture your compilation step and execution, using as many sets of test data as you feel are appropriate. Turn in a copy of your source code, your test data, appropriately labeled, and the output of the ***script*** command.

Create a directory under cs352 called hw3. In that directory leave your source code and any necessary input files. Make sure the directory and files in it have the appropriate permissions and ACLs for me to access them and possibly write to the directory. (See instructions for turning in HW1.

Grading Criteria

For the program in this assignment, the following grading criteria will be used.

Program Design and Style	20%
Program Documentation	10%
Quality of Test Cases	10%
Correctness of Output	60%

By ***Program Style*** I mean the quality of the code: is it readable, are there appropriate comments, is the code as efficient as it could be (e.g., does it make two passes through the input data when one pass is sufficient with an appropriate algorithm and/or data structure, are normal C coding conventions used, .h files), are appropriate data structures and algorithms used, are variables, and language features used as well as they could be, are there meaningful identifier names. All of these characteristics contribute to high quality code, which is what you should be producing.

For ***Quality of Test Cases*** I'm looking for coverage of the code. Test cases should be considered before you start coding. What are you going to test and why? I also expect that the test data is clearly identified: input and corresponding output.

Finally, for ***Correctness of Output*** I will give partial credit for clean compilation. The output should be clearly identified and nicely formatted.