Important: This is an open book test. You can use any books, notes, or paper but no electronic device. *Do not log into the computer during the test, or use any electronic or communications device. Switch off your cell phones.* Any calculations and rough work can be done on the back side of the test pages. If there is a syntax error in any program segment, just write it down and you will get full credit for the problem. You will lose five points for not writing your name.

1. [6 pt] You are given a file named `datafile` that has some lines that are made up exclusively of numbers. The other lines could be a mix of letters, numbers, and special characters. Give the sed command to split the file into two: `datanum` and `data`. The file `datanum` should have all the lines that have nothing but numbers. The file `data` should have all the other lines.

2. [6 pt] What compiler flags will you use in `gcc` to prepare a file for debugging? What is the difference between `break` and `tbreak` commands in `gdb`?

3. [10 pt] You are preparing a prototype header file called `protos.h`. All your functions in the C source files start in column 1 with the function headers limited to one line. Give me a command to automatically build the file `protos.h` from your C source files. You can assume that the function body does not have any line starting at column 1 except for the opening and closing curly braces. The functions look like:

```
int func ( int argv1, float argv2 )
{
    // function body
}
```

where `int` and `float` could be substituted by other data types and the number of arguments could change as well.

4. [4 pt] Why do you profile a given piece of code?

5. [8 pt] You have released some program in the field, and there are multiple versions of this program out there. Somebody reported a bug with a given version. You ask this person to tell you the revision number of the code being used by using the option `-v`. What should you have done using RCS keywords to get the revision number to be printed. (Just describe the process giving specifics only for RCS keywords. You do not have to give me any C code.)

6. [10 pt] When you `fork()` a process, how do you differentiate between a child and a parent? Typically, the child can call a function to perform a job and `exit`, or call `execl` and `exit`. What is the difference between calling a function and performing `execl`?