

Assignment 3

100 points

Purpose

This assignment should give you experience with input loops and file I/O.

Assignment

Create two programs, one for generating a file of random integers, the other for counting integers greater than a specified threshold.

generate

Create a program called "generate" which generates a file of random integers.

This program asks for four values. They are, in order

1. A random number seed. This is an integer value explained below.
2. A maximum value. The random values should be less than this maximum. This value is an integer.
3. The number of values to be generated
4. The name of an output file to store the values

The integer values should all be checked for negative values. If a negative value is given, the program should print an error message and exit.

Random number generation

Random numbers on a computer are not truly random. They just look that way. The `rand()` function, found in `<stdlib.h>` returns a random positive integer every time it is called.

In scientific programming, it is often desirable to reproduce a random number sequence. The `srand(int)` function takes an integer called the *random number seed* which is used to initialize the random number generator. Subsequent calls to `rand()` produce a random sequence which is uniquely tied to its seed. Typically in a program, `srand()` is called once, while `rand()` is called multiple times.

A common technique for producing random numbers of a particular size is to use the remainder of dividing `rand()` by the maximum upper bound.

thresh

Create a program called `thresh` which should ask the user for a file name and an integer threshold value. The program should then count the number of values in the file greater than the threshold and report the results. For example,

```
43 of 300 values in mydatafile are greater than 17
```

When asking for the threshold value, if the user inputs a negative value, the program should print an error and loop on the request until a proper value is obtained.

If the input file does not exist, an error message should be printed and the program should exit.

Hints

- Create `generate` first.
- Test your programs with small values that can be checked by hand. For example, create a file of 20 values less than 10.
- A particularly good test of `thresh` is to use the last value of the data file as a threshold value. Then use the last value minus one as a threshold value.

Other Points

- A `Makefile` is required as part of your submission.
- If the project has multiple executables, you should have a default makefile rule to build all of the executables.
- You should have a rule to clear out the project to a pristine state.
- Your `Makefile` should use appropriate variables.

Submission

Place your source code files and `Makefile` in a directory name `assign3`. Use the `mailprog` script on this directory as described on the course web site to submit the assignment.