
C Programming – Strings and Bit Vectors

Write the code in C. I want to see the use of `git` and `Makefile`. Work on each part of the project in its own subdirectory. Use the `README` file to indicate what is being done, especially if you have done something new, and how to execute the program.

1. The *Sieve of Eratosthenes* is a method used to determine all primes less than a given number N . It is very fast. Initially, we write down all integers from 2 to N . Begin P at 2. Cross out all multiples of P starting with $2 \times P$; this is easily done by crossing out every P th entry. Then increment P to the next non-crossed out integer. Again, cross out all multiples of P . Repeat these steps until P is greater than \sqrt{N} . The numbers that have not been crossed out represent all of the prime numbers smaller than or equal to N .

Write a program using a bit vector that implements the sieve for $N = 100,000$. The program should be in file `sieve.c` and the executable should be called `sieve`.

2. Write a function that asks the user to input a telephone number as a string containing a three-digit area code, followed by a seven-digit number. Also write an accompanying test (driver) program. The function and driver must be in separate files. The function should scan the input string, reading only the first ten numeric digits ignoring all other characters in the string. It should report the telephone number in the format (123) 456-7890. Note that the user may choose any input format, yet the program should maintain a consistent output format.

The function should be called `phone_fmt`. Your executable will be called `phone`. The function and executable should be in the files `phone_fmt.c` and `phone.c`, respectively.

3. Create a program to shuffle a deck of cards. The suits and cards are to be defined as

```
typedef enum { clubs, diamonds, hearts, spades} suit_t;
typedef struct
{
    unsigned int number    : 4;
    suit_t        suit      : 2;
} card_t;
```

Declare an array of cards with a complete deck of 52 cards. Populate the cards using a loop such that all the cards of a single suit are together in order as Ace, 2, 3, 4, ..., King. Shuffle the cards using the following algorithm.

Generate a random number between 0 and 51, and call it r . Swap the cards at position 0 and position r . Repeat the loop for a total of 52 iterations.

At the end of the shuffle, print the array.

Then, sort the array using `qsort(3C)` such that all the cards with the same number are together.

This program will be called `shuffle`.

4. Use the same structure as last program for each card. Create a bit vector of 52 bits. It will be achieved by using an array of `unsigned char` with $\lceil 52/8 \rceil$ elements. Clear all the bits by using `memset(3C)` function. In a loop with 52 iterations, create a random number between 0 and 51. Put the card corresponding to this random number in the next location in the array if the corresponding bit in the bit vector is zero, and set that bit to 1. If the bit is already 1, scan the bit vector for the next bit (given by `r % 51`) and put the card corresponding to that bit in the next location, setting that bit to 1.

At the end of the loop, print the array.

Then, sort the array using `qsort(3C)` such that all the cards with the same suit are together.

This program will be called `sb`.

Submission

Create a directory `username.5` in your home where `username` is your user name on delmar. Keep all programs and datafiles for this assignment in this directory.

You do not have to submit any hard copy of the code. Write the code in `bash` using `delmar`. Follow good programming principles and document your scripts well. Do not forget to take care of issues that can cause a wrong utility to execute than the one you intended.

After you are done with the assignment, execute the following commands:

```
% cd
% chmod 755 ~
% ~bhatias/bin/handin cs2750 5
% chmod 700 ~
```