CSC 117

Project 7, Due Monday, August 12, 2002

40 points

Please email the program to the grader by midnight on the due date. You can use the command:

mail liuq@eng.unlv.edu -s "Project 7" < proj7.f90

where 'proj7.f90' is the name of the file you want to send.

(Problem 5, page 606 of your textbook.) A *prime number* is an integer greater than 1 whose only positive divisors are 1 and the integer itself. One method for finding all the prime numbers in the range 2 through n is known as the *Sieve of Eratosthenes*. Consider the list of numbers from 2 through n. Here 2 is the first prime number, but the multiples of 2 (4, 6, 8. ...) are not, and so they are "crossed out" in the list. The first number after 2 that was not crossed out is 3, the next prime. We then cross out all higher multiples of 3 (6, 9, 12, ...) from the list. The next number not crossed out is 5, the next prime; we cross out all higher multiples of 5 (10, 15, 20, ...). We repeat this procedure until we reach the first number in the list that has not been crossed out and whose square is greater than n. Write a program that uses this sieve method to find all the prime numbers from 2 through n. Your program should ask the user to input the value of n, a positive integer not to exceed 4 digits. Your program must use run-time arrays, and before the program terminates, you must deallocate the run-time arrays to free memory. Your output must look like the sample output shown below. The output should be on as many lines as necessary so that no output line has length greater than 79.

Sample Output:

```
 This program finds all primes from 2 through n.
 Enter a positive value for n (not to exceed 9999): 100

Primes in the range 2 - 100 are:
    2    3    5    7   11   13   17   19   23   29   31   37   41   43   47
   53   59   61   67   71   73   79   83   89   97
```