Assignment 6
RSA Key Cryptography
Design Document

Name: Arad Levin
CSE13S - Fall Quarter
Prof. Long
11/19/21

Description of Program:

The purpose of this program is to implement a RSA encrypter and decrypter, with the encrypter reading a given file and encrypting it and the decrypter reading an encrypted file and decrypting it, with the file being the same as it was before encryption. This will be done using keys generated using keygen, using math functions that create prime numbers and perform various other operations. The generated keys will be used to encrypt the provided files and then create a file containing the encrypted data. Thereafter, the decrypt program can be used. With full functionality, a file that is encrypted by the encryption program will be able to be successfully decrypted back to its original state by the decryption program. A help page can also be printed in order to aid with usage of the program, and there are multiple command line options that can be used as well which will be discussed later on.

How Each Part of the Program Should Work:

In numtheory.c, math functions will be created that create and verify prime numbers, as well as finding the modular inverse of some values.

In keygen.c, keys will be created using the math function defined in numtheory.c, the keys will then be printed to outfiles called rsa.priv and rsa.pub. Then everything gets deleted.

In rsa.c, the encryption is done and the information from the encryption is printed/read/verified.

In randstate.c the random state extern variable is created called state using a seed provided in the keygen file.

In encrypt.c, a help function is first defined that will print out the help page that describes how to use the program. After this, there is the main file. Within the main file, all of the necessary variables are defined and then a while loop parses the command

line options, setting the infile/outfile based on them and optionally printing out the help page and exiting the program. Then everything gets deleted.

In decrypt.c, a help function is first defined that will print out the help page that describes how to use the program. After this, there is the main file. Within the main file, all of the necessary variables are defined and then a while loop parses the command line options, setting the infile/outfile based on them and optionally printing out the help page and exiting the program. Then everything gets deleted.

Files/Libraries/Binaries:

DESIGN.pdf (pushed to git)
- This file

README.md (pushed to git)
- Contains information about the program and how to run it.

rsa.h (pushed to git) (provided)

rsa.c (pushed to git)

numtheory.h (pushed to git) (provided)

numtheory.c (pushed to git)

randstate.h (pushed to git) (provided)

randstate.c (pushed to git)

encrypt.c (pushed to git)

decrypt.c (pushed git)

keygen.c (pushed to git)

Makefile (pushed to git)
- Contains instructions used to create a binary that can be run in order to use the program, instructions for removing unnecessary files,
and instructions for formatting all of the files.