COMP-4476 - Assignment 2

Name: Srijan Ravisankar
Student #: 1302850

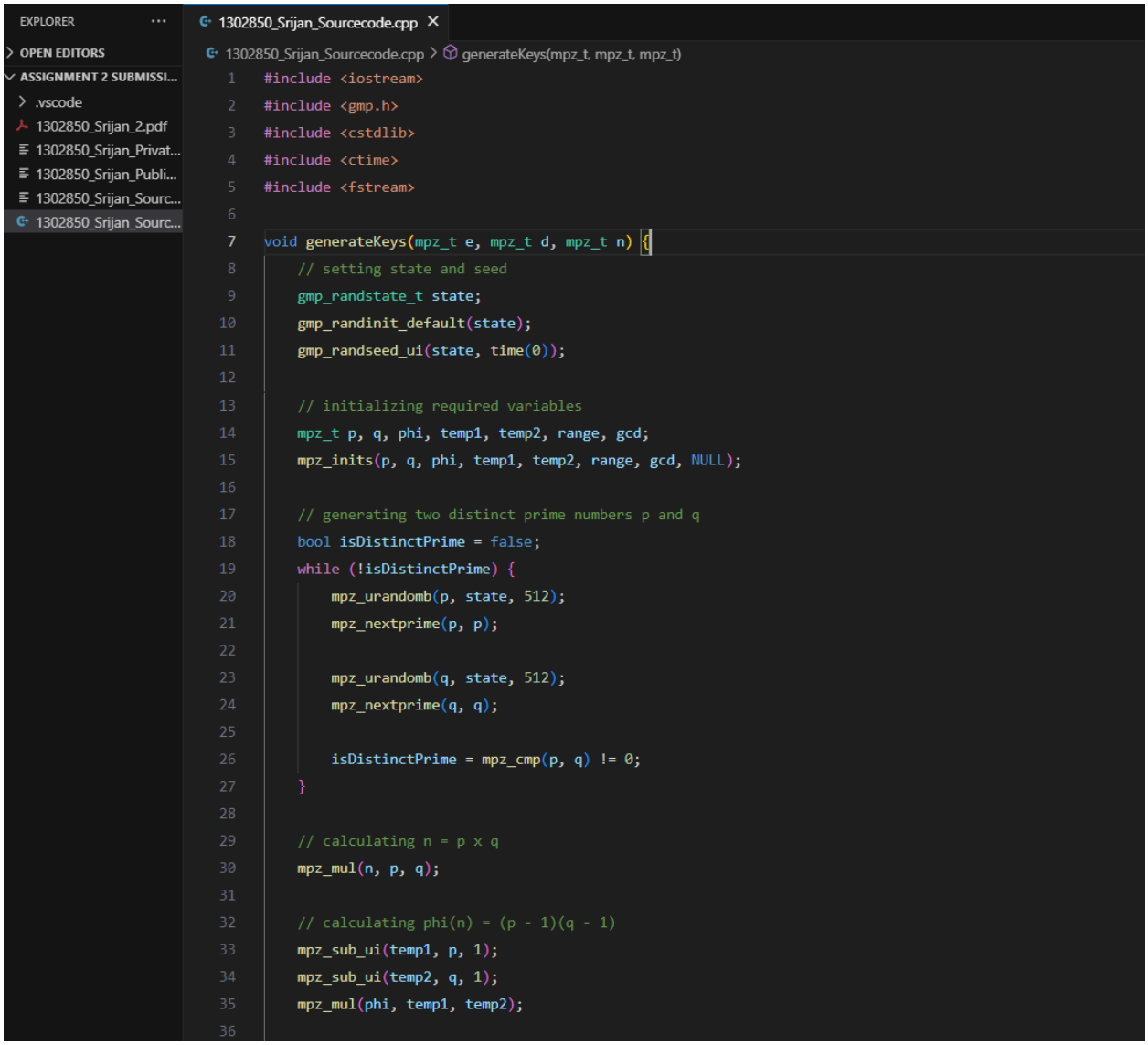
Editor/Compiler used for the task:

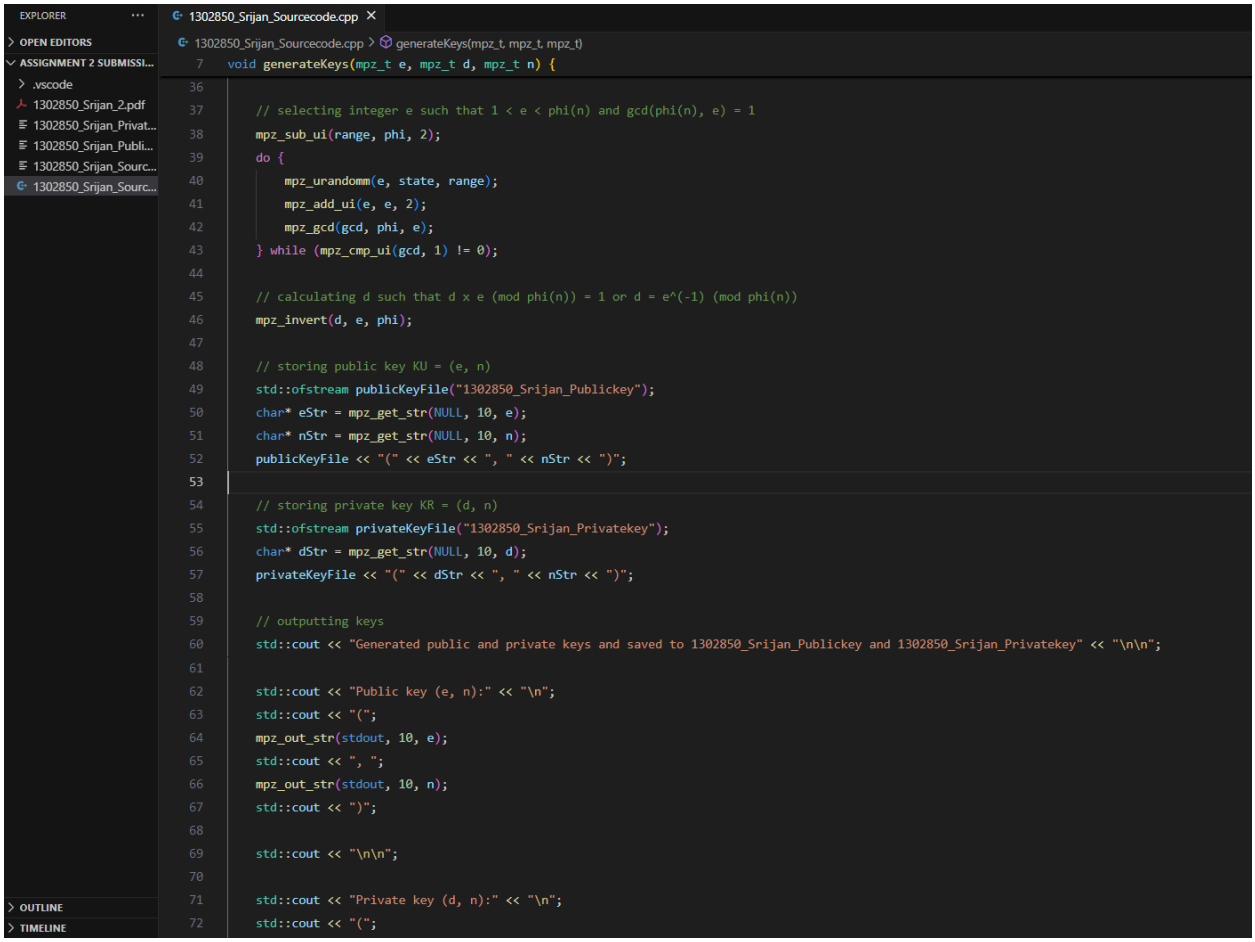
- ★ Editor: Visual Studio Code (VS Code) was used as the primary code editor for development.
- ★ Compiler: The code was compiled and executed using the GNU Compiler Collection (GCC) within the Windows Subsystem for Linux (WSL) environment.

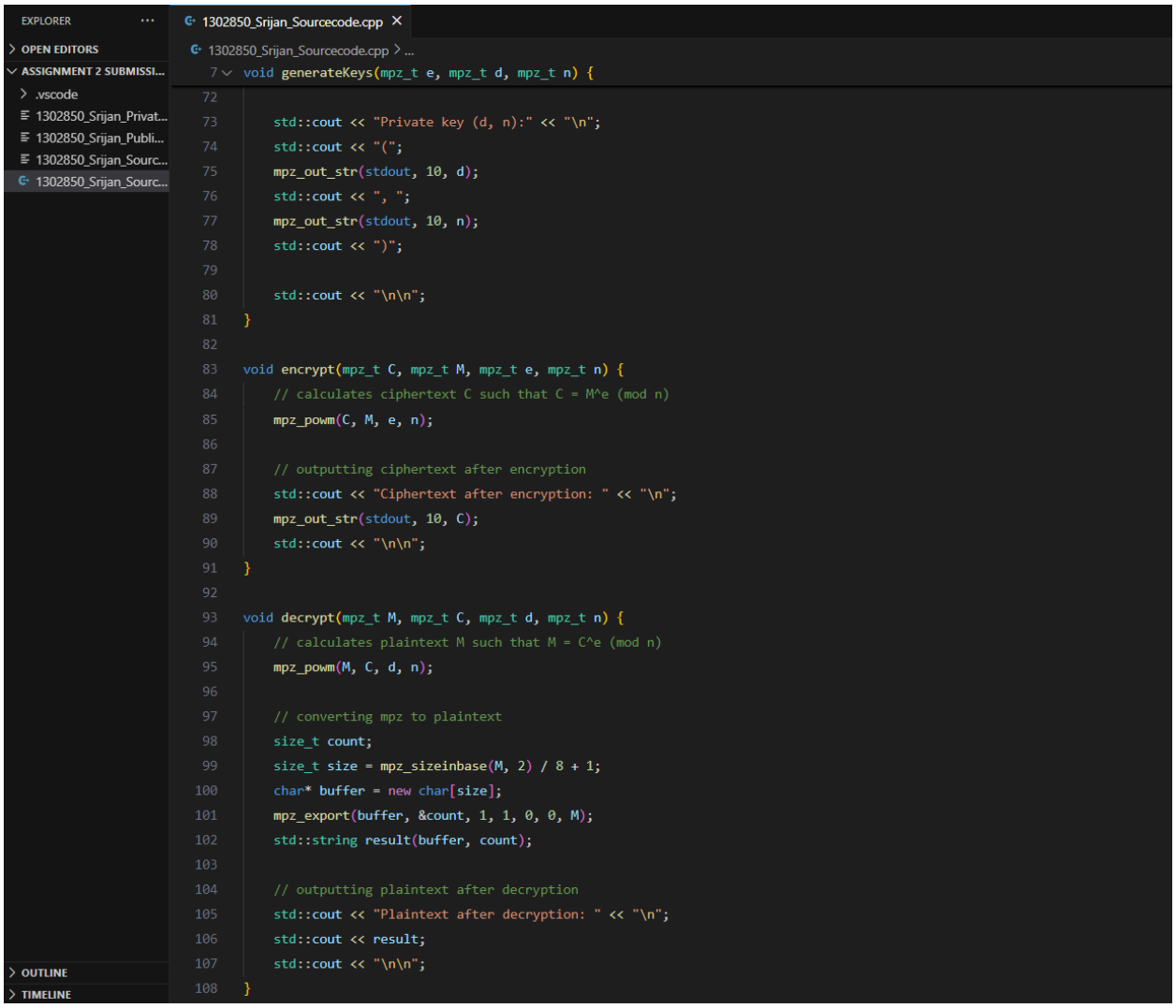
How to Run the Code:

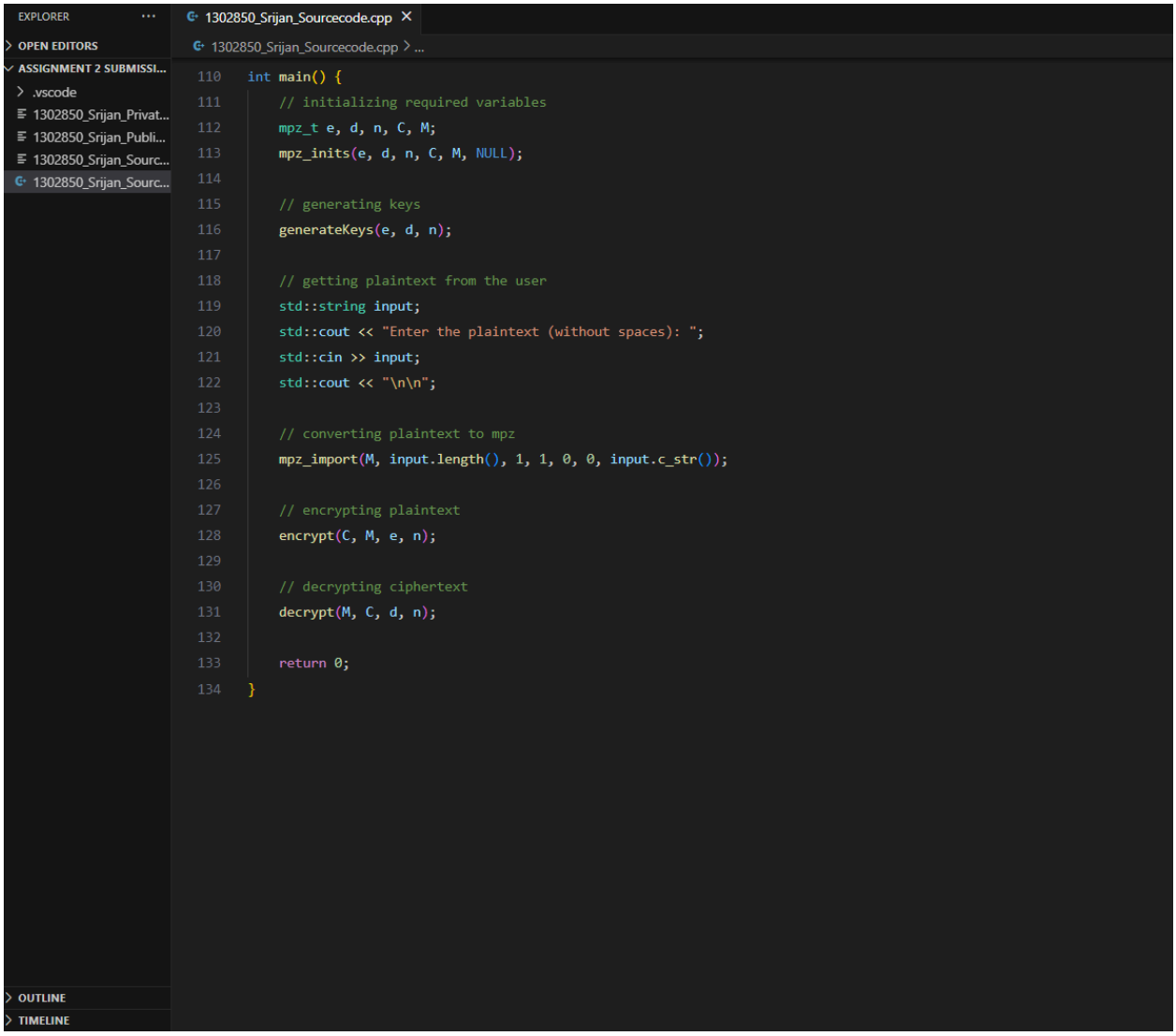
- ★ This C++ program is a single-file source code that can be compiled and executed using g++.
- ★ Prerequisites:
  - Ensure you have g++ installed. You can check by running:
    sh g++ --version
  - If g++ is not installed, you can install it using:
    - On Ubuntu/Debian:
      sh sudo apt update && sudo apt install g++
    - On macOS (via Homebrew):
      sh brew install gcc
    - On Windows (via MinGW):
      - Install MinGW from https://www.mingw-w64.org/
      - Ensure g++ is added to your system's PATH
- ★ Compilation and Execution:
  - Use Ubuntu for compilation and execution.
  - Update Your Package Lists:
    sudo apt update
  - Install GMP Development Library:
    sudo apt install libgmp-dev
  - Verify Installation:
    gmp version
  - Compile and then run Sample C++ Program Using GMP:
    g++ test_gmp.cpp -o outputfile -lgmp
    ./test_gmp
- ★ Opening public key and private key files:
  - Open the output files with VS Code or any Browser.

Code Screenshots:



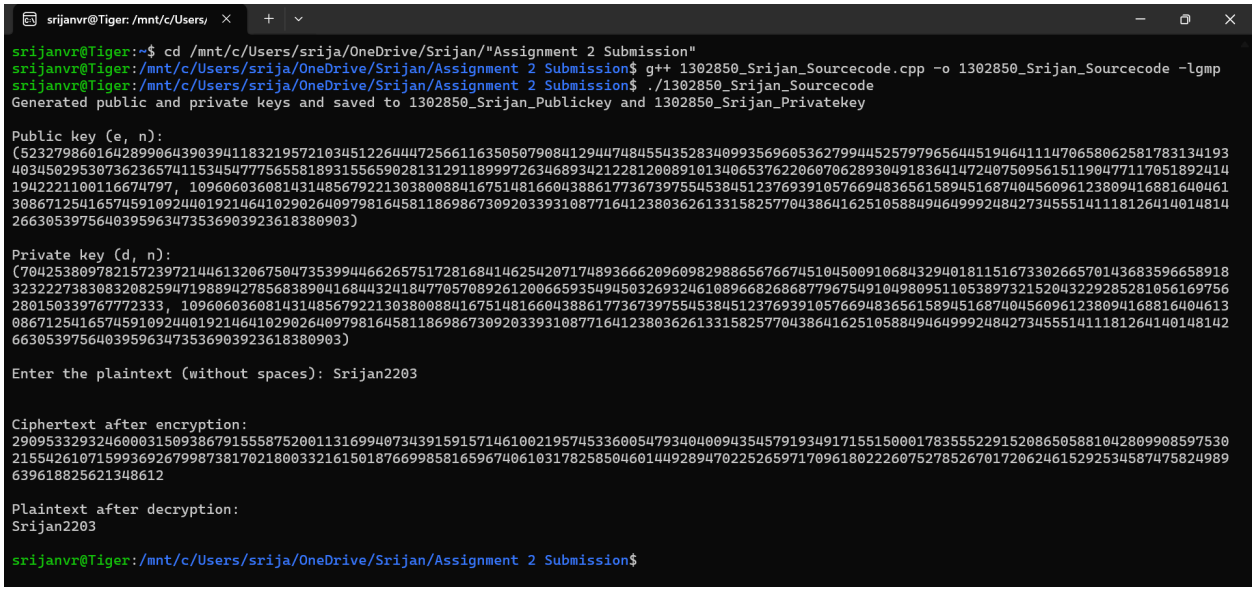

EXPLORER
1302850_Srijan_Sourcecode.cpp
OPEN EDITORS
ASSIGNMENT 2 SUBMISSI...
.vscode
1302850_Srijan_2.pdf
1302850_Srijan_Privat...
1302850_Srijan_Publi...
1302850_Srijan_Sourc...
1302850_Srijan_Sourc...
OUTLINE
TIMELINE
1302850_Srijan_Sourcecode.cpp > generateKeys(mpz_t, mpz_t, mpz_t)
7 void generateKeys(mpz_t e, mpz_t d, mpz_t n) {
36
37 // selecting integer e such that 1 < e < phi(n) and gcd(phi(n), e) = 1
38 mpz_sub_ui(range, phi, 2);
39 do {
40 mpz_urandomm(e, state, range);
41 mpz_add_ui(e, e, 2);
42 mpz_gcd(gcd, phi, e);
43 } while (mpz_cmp_ui(gcd, 1) != 0);
44
45 // calculating d such that d x e (mod phi(n)) = 1 or d = e^(-1) (mod phi(n))
46 mpz_invert(d, e, phi);
47
48 // storing public key KU = (e, n)
49 std::ofstream publicKeyFile("1302850_Srijan_Publickey");
50 char* eStr = mpz_get_str(NULL, 10, e);
51 char* nStr = mpz_get_str(NULL, 10, n);
52 publicKeyFile << "(" << eStr << ", " << nStr << ")";
53
54 // storing private key KR = (d, n)
55 std::ofstream privateKeyFile("1302850_Srijan_Privatekey");
56 char* dStr = mpz_get_str(NULL, 10, d);
57 privateKeyFile << "(" << dStr << ", " << nStr << ")";
58
59 // outputting keys
60 std::cout << "Generated public and private keys and saved to 1302850_Srijan_Publickey and 1302850_Srijan_Privatekey" << "\n\n";
61
62 std::cout << "Public key (e, n):" << "\n";
63 std::cout << "(";
64 mpz_out_str(stdout, 10, e);
65 std::cout << ", ";
66 mpz_out_str(stdout, 10, n);
67 std::cout << ")";
68
69 std::cout << "\n\n";
70
71 std::cout << "Private key (d, n):" << "\n";
72 std::cout << "(";


EXPLORER
1302850_Srijan_Sourcecode.cpp
OPEN EDITORS
ASSIGNMENT 2 SUBMISSI...
.vscode
1302850_Srijan_Privat...
1302850_Srijan_Publi...
1302850_Srijan_Sourc...
1302850_Srijan_Sourc...
OUTLINE
TIMELINE
1302850_Srijan_Sourcecode.cpp > ...
7 void generateKeys(mpz_t e, mpz_t d, mpz_t n) {
72
73 std::cout << "Private key (d, n):" << "\n";
74 std::cout << "(";
75 mpz_out_str(stdout, 10, d);
76 std::cout << ", ";
77 mpz_out_str(stdout, 10, n);
78 std::cout << ")";
79
80 std::cout << "\n\n";
81 }
82
83 void encrypt(mpz_t C, mpz_t M, mpz_t e, mpz_t n) {
84 // calculates ciphertext C such that C = M^e (mod n)
85 mpz_powm(C, M, e, n);
86
87 // outputting ciphertext after encryption
88 std::cout << "Ciphertext after encryption: " << "\n";
89 mpz_out_str(stdout, 10, C);
90 std::cout << "\n\n";
91 }
92
93 void decrypt(mpz_t M, mpz_t C, mpz_t d, mpz_t n) {
94 // calculates plaintext M such that M = C^e (mod n)
95 mpz_powm(M, C, d, n);
96
97 // converting mpz to plaintext
98 size_t count;
99 size_t size = mpz_sizeinbase(M, 2) / 8 + 1;
100 char* buffer = new char[size];
101 mpz_export(buffer, &count, 1, 1, 0, 0, M);
102 std::string result(buffer, count);
103
104 // outputting plaintext after decryption
105 std::cout << "Plaintext after decryption: " << "\n";
106 std::cout << result;
107 std::cout << "\n\n";
108 }

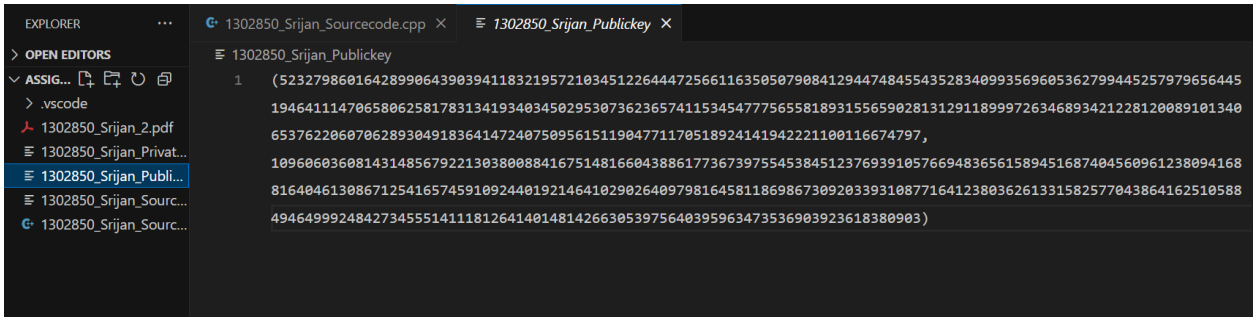

EXPLORER
OPEN EDITORS
ASSIGNMENT 2 SUBMISSI...
.vscode
1302850_Srijan_Privat...
1302850_Srijan_Publi...
1302850_Srijan_Sourc...
1302850_Srijan_Sourc...
OUTLINE
TIMELINE
1302850_Srijan_Sourcecode.cpp
1302850_Srijan_Sourcecode.cpp > ...
110 int main() {
111 // initializing required variables
112 mpz_t e, d, n, C, M;
113 mpz_inits(e, d, n, C, M, NULL);
114
115 // generating keys
116 generateKeys(e, d, n);
117
118 // getting plaintext from the user
119 std::string input;
120 std::cout << "Enter the plaintext (without spaces): ";
121 std::cin >> input;
122 std::cout << "\n\n";
123
124 // converting plaintext to mpz
125 mpz_import(M, input.length(), 1, 1, 0, 0, input.c_str());
126
127 // encrypting plaintext
128 encrypt(C, M, e, n);
129
130 // decrypting ciphertext
131 decrypt(M, C, d, n);
132
133 return 0;
134 }

Output Screenshots:

Terminal output:



Public key file output:



Private key file output: