Nathan Ha SID 862377326

CS 111 ASSIGNMENT 2

due February 5

Problem 1: Prove the following statement:

If $p > 5$ and $gcd(p, 20) = 1$, then $(p^2 - 21)(p^2 + 16) \equiv 0 \pmod{20}$.

Hint: The product of any k consecutive integers is divisible by k.

Solution 1:

We are trying to prove that $(p^2 - 21)(p^2 + 16)$ is divisible by 20. First, we should remember one of the properties of modular arithmetic: $ab \pmod{c} \equiv (a \pmod{c})(b \pmod{c})$. Notice that within each factor, we can add or subtract c as much as we like, and the value would not change. We will use this to simplify the original problem:
$(p^2 - 21)(p^2 + 16) \equiv (p^2 - 21 + 20)(p^2 + 16 - 20) \equiv (p^2 - 1)(p^2 - 4) \pmod{20}$.
We can simplify this further: $(p^2-1)(p^2-4) = (p+1)(p-1)(p+2)(p-2) = (p-2)(p-1)(p+1)(p+2)$.

For now, let's look at a different, but similar product: $(p - 2)(p - 1)(p)(p + 1)(p + 2)$. This is a product of 5 consecutive integers. We know that a product of 5 consecutive integers is divisible by 5 (by the hint). We also know that p is not divisible by 5 and 4, since p is coprime with 20. This means that one of the factors other than p must be divisible by 5. following similar logic, we also know that one of he non-p factors must be divisible by 4, since there is also a product of 4 consecutive integers. Going back to $(p - 2)(p - 1)(p + 1)(p + 2)$, Since there is a factor that is divisible by 4, and another by 5, the product must be divisible by 20.
Therefore, the statement $(p^2 - 21)(p^2 + 16) \equiv 0 \pmod{20}$ holds.

Problem 2:

Alice's RSA public key is $P = (e, n) = (7, 4453)$. Bob sends Alice the message by encoding it as follows. First he assigns numbers to characters: A is 8, B is 9, ..., Z is 33, a blank is 34, quotation marks: 35, a coma: 36, a period: 37, an apostrophe: 38. Then he uses RSA to encode each number separately.

Bob's encoded message is:

```
1400 2218   99 2088 4191   84  843   99 4191 3780  764 4191 2979 2269   99  764
2218 2269 2088  843 3015   99 2970 1443 1655   99 3237 2979   99  447 1443 3237
1032 2382  871  843 1655   99  871 1443   99 4242  843   99 4191 2269   99  843
4191 2269 2979   99  871 1443   99 2382 2269  843   99 4191 2269   99 3237 2979
  99  871  843 3780  843 1032 2088 1443 2962  843 2916   99 3237 2979   99  764
2218 2269 2088   99 2088 4191 2269   99  447 1443 3237  843   99  871 1655 2382
 843   99 4242  843  447 4191 2382 2269  843   99 2218   99  447 4191 2962   99
2962 1443   99 3780 1443 2962 1294  843 1655   99 2970 2218 1294 2382 1655  843
  99 1443 2382  871   99 2088 1443  764   99  871 1443   99 2382 2269  843   99
3237 2979   99  871  843 3780  843 1032 2088 1443 2962  843 2916 1400
```

Decode Bob's message. Notice that you only know Alice's public key, but don't know the private key. So you need to "break" RSA to decrypt Bob's message. For the solution, you need to provide the following:

(a) Describe step by step how you arrived at the solution: show how to find p and q, $\phi(n)$ and d.

(b) Show your work for one integer in the message (M = 2218): the expression, the decrypted integer, the character that it is mapped to.

(c) To decode the remaining numbers, you need to write a program in C++ (see below), test it in Gradescope, and append the code to HW 2, Problem 2 solutions.

(d) Give the decoded message (in integers).

(e) Give Bob's message in plaintext. What does it mean and who said it?

For part (c). Your program should :

(i) Take three integers, e, n (the public key for RSA), and m (the number of characters in the message) as input to your program. Next, input the ciphertext.

(ii) Test whether the public key is valid. If not, output a single line "Public key is not valid!" and quit the program.

(iv) If the public key is valid, decode the message.

(v) Output p and q, $\phi(n)$ and d.

(vi) On a new line, output the decoded message in integers.

(vii) On a new line, output the decoded message in English. The characters should be all uppercase. You can assume that the numbers will be assigned to characters according to the mapping above.

More information and specifications will be provided separately.

Upload your code to Gradescope to test. There will be 15-16 (open and hidden) test cases. Your score for the RSA code will be based on the score that you received in Gradescope. If you have any questions, post them on Slack.

Solution 2:
(a)

In order to break the RSA, we first have to find p and q. Since n is a relatively small value, we can brute force its divisors. Using brute force, we can find p, and then $q = \frac{n}{p}$. We find that $(p, q) = (61, 73)$. Now that we have p and q, we can calculate $\phi(n) = \phi(p)\phi(q) = (p-1)(q-1) = 4320$. At this point, we have everything we need to get the private key. We just need to find
$d = e^{-1} \pmod{\phi(n)}$. There are several ways to compute this inverse, but I just did it by listing multiples. Now that we have d, we just need to raise the encrypted message to the power of p, modulo n $\implies d = 3703$. Even though the exponent is huge, we can just simplify it by squaring. Once you simplify it, you will get the integer that corresponds to the message.

(b)
To decode 2218, we will raise it to the power of d (which we found to be 3703), modulo n. We can then simplify it by squaring:
$2218^{3703} \equiv 2218(2218)^{1851} \pmod{4453}$
$\equiv 2218(4919524)^{1851} \pmod{4453}$
$\equiv 2218(3412)^{1851} \pmod{4453}$
$\vdots$
$\equiv 266(3249)^3 \pmod{4453}$
$\equiv 266(3249)(3249^2) \pmod{4453}$
$\equiv 352 \cdot 2391 \pmod{4453}$
$\equiv 15 \pmod{4453}$. Now we have that the decoded message is 15. According to the character assignments, 15 is the letter 'I'.

(c)
The code for main is on the last page. I didn't include the other functions that I wrote because they took up too much space, but they are on gradescope.

(d)
The decoded message in integers:
34 15 33 14 7 28 11 33 7 18 29 7 31 25 33 29 15 25 14 11 10 33 12 21 24 33 19 31 33 9 21 19 22 27 26 11 24 33 26 21 33 8 11 33 7 25 33 11 7 25 31 33 26 21 33 27 25 11 33 7 25 33 19 31 33 26 11 18 11 22 14 21 20 11 36 33 19 31 33 29 15 25 14 33 14 7 25 33 9 21 19 11 33 2 6 24 27 11 33 8 11 9 7 27 25 11 33 15 33 9 7 20 33 20 21 33 18 21 20 13 11 24 33 12 15 13 27 24 11 33 21 27 26 33 14 21 29 33 26 21 33 27 25 11 33 19 31 33 26 11 18 11 22 14 21 20 11 36 34

(e)
In plaintext, the message is:
"I HAVE ALWAYS WISHED FOR MY COMPUTER TO BE AS EASY TO USE AS MY TELEPHONE. MY WISH HAS COME TRUE BECAUSE I CAN NO LONGER FIGURE OUT HOW TO USE MY TELEPHONE." This is a quote by Bjarne Stroustrup creator of C++. He is talking about the fact that technology has gotten so advanced so quickly.

Problem 3:

(a) Compute $5^{1627} \pmod{12}$. Show your work.

(b) Compute $8^{-1} \pmod{17}$ by listing the multiples. Show your work.

(c) Compute $8^{-1} \pmod{17}$ using Fermat's Little Theorem. Show your work.

(d) Compute $8^{-11} \pmod{17}$ using Fermat's Little Theorem. Show your work.

(e) Find an integer x, $0 \leq x \leq 40$, that satisfies the following congruence: $31x + 54 \equiv 16 \pmod{41}$. Show your work. You should not use brute force approach.

Solution 3:

(a) $5^{1627} \pmod{12}$
$\equiv 5 \cdot (5^2)^{813} \pmod{12}$
$\equiv 5 \cdot (25)^{813} \pmod{12}$
$\equiv 5 \cdot (1)^{813} \pmod{12}$
$\equiv 5 \pmod{12}$.

(b) $8^{-1} \pmod{17} \implies 8a \equiv 1 \pmod{17} \implies 8a = 17b + 1$. We need to find an a to make this equation true.
Multiples of 8: $8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120$
Multiples of 17 (and then +1): $18, 35, 52, 69, 86, 103, 120$
We can see that the equation is true when $a = 15$ and $b = 7$. This means that $8^{-1} \pmod{17} \equiv 15$.

(c) According to Fermat's Little Theorem, $a^{p-1} \equiv 1 \pmod{p}$, where p is prime. Since $8^{-1} \pmod{17}$, and 17 is prime, then $8x \equiv 1 \pmod{17}$, for some a. By Fermat's Little Theorem, we can say: $8^{16} \equiv 8^{17-1} \equiv 1 \pmod{17} \implies 8 \cdot 8^{15} \equiv 1 \pmod{17}$
$\implies 8^{-1} \equiv 8^{15} \pmod{17}$
$\equiv 8(8^2)^7 \pmod{17}$
$\equiv 8(64)^7 \pmod{17}$
$\equiv 8(13)^7 \pmod{17}$
$\equiv (8)(13)(13^2)^3 \pmod{17}$
$\equiv 104(169)^3 \pmod{17}$
$\equiv 2(16)^3 \pmod{17}$
$\equiv 2(16)(16)^2 \pmod{17}$
$\equiv 32(256) \pmod{17}$
$\equiv 15(256) \pmod{17}$
$\equiv 15(1) \pmod{17}$
$\equiv 15 \pmod{17}$.
Therefore, $8^{-1} \pmod{17} = 15$.

(d) $8^{-11} \pmod{17}$ is the same as $(8^{-1})^{11} \pmod{17}$. we know that, by the result found in parts b and c, that $8^{-1} \pmod{17} = 15$. This means that $(8^{-1})^{11} \equiv 15^{11} \pmod{17}$. From here, since we want to use Fermat's Little Theorem, we should multiply by $(8^5)(8^{-1})^5 \equiv (8^5)(15)^5 \equiv 1 \pmod{17}$. We know this to be true because of the properties of inverses. We now have $8^5(15^5)(15^{11})$
$\equiv 8^5(15)^{16} \pmod{17}$. By Fermat's Little Theorem (as stated in part c), we can substitute 15^{16} with 1: $8^5(15)^{16} \pmod{17} \equiv 8^5 \pmod{17}$. Now, we can simplify by squaring:
$8^5 \equiv 8 \cdot 64^2 \pmod{17}$
$\equiv 8 \cdot 13^2 \pmod{17}$
$\equiv 8 \cdot 169 \pmod{17}$
$\equiv 8 \cdot 16 \pmod{17}$
$\equiv 128 \pmod{17}$
$\equiv 9 \pmod{17}$.

(e) $31x + 54 \equiv 16 \pmod{41}$. First, we can subtract both sides by 54: $31x \equiv -38 \pmod{41}$
$\equiv 3 \pmod{41}$. Now, we have to find the inverse $31^{-1} \pmod{41}$. We can list multiples to find the solution of $31a = 41k + 1$:
multiples of 31: 31, 62, 93, 124
multplies of 41(+1): 42, 83, 124
As we can see, the equation is true when $a = 4$ and $k = 3$. This means that the inverse of 31 is 4. Going back to the original equation, we can multiply both sides by 31^{-1}:
$31 \cdot 31^{-1}x \equiv 31^{-1} \cdot 3 \pmod{41} \implies x \equiv 4 \cdot 3 \pmod{41} \equiv 12 \pmod{41}$. Therefore, $x = 12$.

Academic integrity declaration. I did this homework by myself. I got help from the professor and the TA during office hours, though.

```
int main()
{
    int e = 7;
    int n = 4453;
    int char_count = 158;
    cin >> e >> n >> char_count;
    vector<int> data;
    // get message
    for (int i = 0; i < char_count; i++) {
        int temp_msg = 0;
        cin >> temp_msg;
        data.push_back(temp_msg);
    }
    // find p,q
    int p = solve_for_p(n); // 61
    int q = n / p;          // 73
    // find phi
    int phi = (p - 1) * (q - 1); // 4320
    // test public key validity:
    // e and phi(n) are coprime, n has two prime divisors
    if (__gcd(e, phi) != 1 or !is_prime(p) or !is_prime(q) or p == q) {
        cout << "Public key is not valid!";
        return 1;
    }
    // find d
    int d = mod_inv(e, phi); // 3703
    // decrypt message
    for (auto &current_letter : data) {
        int decrypted_data = pow_mod(current_letter, d, n);
        current_letter = decrypted_data;
    }
    // Output p and q, phi(n) and d
    cout //<< endl << p << ' ' << q << ' '<< phi << ' '<< d << endl;
    // print decrypted message in integers
    for (auto &current_letter : data) {
        cout << current_letter << ' ';
    }
    cout << endl;
    // printed decryped message in English
    for (auto &current_letter : data) {
        cout << int_to_char(current_letter);
    }
}
```