CS111 W'24 ASSIGNMENT 2

Ben Pham SID 862387254
Gokul Nookula SID 862366253

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: Given: $p > 5$
Find p: where
$gcd(p, 20) = 1$
$p = 7$ because
$\frac{20}{7} = 2rem6$
$\frac{7}{6} = 1rem1$
1 is where GCD = 7
We know:

(1) $(p^2 - 21) \equiv 0(mod20)$

(2) $(p^2 + 16) \equiv 0(mod20)$

(1) Solving for first
$p^2 - 21 \equiv 0(mod20)$
We can write 21 also as 20 + 1 so:
$p^2 - (20 + 1) \equiv 0(mod20)$
$p^2 - 20 - 1 \equiv 0(mod20)$
Here we can mod -20 by 20 so we end it using Modular arithmetic with negative numbers we get.
$p^2 \equiv 20(mod20)$
Formula:
$-20 \equiv a(20) + n($n must be a positive number
$-20 \equiv 1(20) + 0$
So back to the equation
$p^2 - 1 \equiv 0(mod20)$ form of $a^2 - b^2 = (a + b)(a - b)$
So:
$(p - 1)(p + 1) \equiv 0(mod20)$

(2) Solving the second:
$p^2 + 16 \equiv 0(mod20)$
We can rewrite 16 as 20 - 4 so:
$p^2 + (20 - 4) \equiv 0(mod20)$
$p^2 + 20 - 4 \equiv 0(mod20)$
Use the same logic above. $p^2 - 4 \equiv 0(mod20)$
$(p - 2)(p + 2) \equiv 0(mod20)$
Now we have: (Gathering both equations together
$(p - 1)(p + 1)(p - 2)(p + 2)(p + 2) \equiv 0(mod20)$
$(p - 2)(p - 1)(p + 2)(p - 1) \equiv 0(mod20)$
Now we can multiply the above sequence with p to get 5 - sequence
$(p - 2)(p - 1)p(p + 2)(p - 1) \equiv 0(mod20)$
For the lone variable p, we are trying to show that when we multiply the equation with p it does not change it.

$(p-2)(p-1)p(p+2)(p-1) \equiv 0(mod 20)$
So from (p-2) to (p-1) it is 2 sequences, same goes with (p+2) to (p-1) which is also two sequences.
But if you put (p-2) to (p+2) it is in total 4 sequence which is (p-2)(p-1)p(p+2).
Also the reason why it is 2 sequence is due to the LCM of 20 = 2,2 and 5 or 4 and 5.
So we know that LCM of 20 = 2,2 and 5 or 4 and 5.
Then from gcd(p,20) = 1, we can see that n does not introduce 4 and 5 as factors. Hence multiplying n does not change the divisibility by 20.
Hence we can conclude that, $(p^2 - 21)(p^2 + 16) \equiv 0 \pmod{20}$.

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 7, B is 8, ..., Z is 32, a blank is 33, quotation marks: 34, a coma: 35, a period: 36, an apostrophe: 37. 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)
First, we are given the public keys:
$e = 7$
$n = 4453$
We need to factorize n
Since $n = p * q$
We see that $61 * 73 = 4453$
So $p = 61$ and $q = 73$
Now let calculate $\phi(n)$:
Since 'p' and 'q' are primes we use the formula:
$\phi(n) = (p-1)(q-1) = (60) * (72) = 4320$
Now we must calculate d:
Formula:
$d = e^-1(\pmod{\phi(n)})$
$\Rightarrow$ In this case, $d \equiv 7^-1 \pmod{4320}$
$\equiv 7^{-1} \pmod{4320} = 1$
We need to find α, β such that: $\alpha * 7 + \beta * 4320 = 1$
Multiples of 7:
$7, 14, 21, ..., 25921$ (Listing it all the way to 7 * 3703)
Multiples of 4320:
$4320, 8640, 12960, 17280, 21600, 25920$
So $\alpha = 3703, \beta = -6 : 7 * 3703 + (-6) * 4320 = 1$
And this gives us that $7^{-1} \pmod{4320} = 3703$

(b)
We know that: $c = 2218$ (Replace M with C as this is decryption!)
We would use the Decryption Formula $= D(C) = C^d rem\ n$
Then as the problem goes on if the number is big enough we would keep modding it.
$2218^{3703} rem 4453$
$2219 * (2218^2)^{1851} rem 4453$
$2218(3412)^{1851} rem 4453$
$2218 * 3412(34126^2)^{925} rem 4453$
$2169 * 1602(1602^2)^{462} rem 4453$
$2169 * 1602(1602^2)^{462} rem 4453$
$1398(1476)^{462} rem 4453$
$1398(1059)^{231} rem 4453$
$1398 * 1059(1059)^{230} rem 4453$

$2086(1059^2)^{115} rem 4453$
$2086(3778)^{115} rem 4453$
$2086 * 3378(3778)^{114} rem 4453$
$3551(3778^2)^{57} rem 4453$
$3551(1419)^{57} rem 4453$
$3551 * 1419(1419)^{56} rem 4453$
$2526(1419^2)^{28} rem 4453$
$2526(805)^{28} rem 4453$
$2526(805^2)^{14} rem 4453$
$2526(2340^2)^{7} rem 4453$
$2526(2863)^{7} rem$
$2526 * 2863(2863)^{6} rem 4453$
$266(2863^2)^{3} rem 4453$
$266(3249)^{3} rem 4453$
$266 * 3249(3249^2) rem 4453$
$352(3249)^{2} rem 4453$
$352 * 2391 rem 4453$
$= 15$

Meaning 15 is the decrypted integer and from the look of it, it is pointing to the letter I!

(c)

```
#include <iostream>
#include <vector>
#include <cmath>
#include <algorithm>

using namespace std;

void decodedMessage(int);

int main() {
    int e = 0;
    int n = 0;
    int m = 0;
    int num = 0;
    int p = 0;
    int q = 0;
    int phi = 0;
    int d = 0;
    bool prime = true;
    vector <int> message;

    cin >> e >> n;
    cin >> m;

    //Reads in numbers from message and stores in vector
    for (int i = 0; i < m; i++) {
        cin >> num;
        message.push_back(num);
    }

    //Find p and q through brute force
    for (int i = 2; i < n - 1; i++) {
        if (n % i == 0) {
            p = i;
            q = n / i;
        }
    }
```

```
    //If they are prime, they should not be divisible by numbers other than 1 and itself
    for (int i = 2; i < p; i++) {
        if (p % i == 0) {
            prime = false;
            break;
        }
    }

    for (int i = 2; i < q; i++) {
        if (q % i == 0) {
            prime = false;
            break;
        }
    }

    //if p greater than q, we swap since we want p < q
    if (p > q) {
        int temp = p;
        p = q;
        q = temp;
    }

    phi = (p-1) * (q-1);

    if (p == q || (__gcd(e, phi) != 1)|| prime == false) {
        cout << "Public key is not valid!";
        return 0;
    }
    else {
    int e2 = e;
    int phi2 = phi;
    int count = 1;

    //We find d through listing multiples
    while(e2 != phi2 + 1) {
        if (e2 > phi2) {
            phi2 += phi;
        }
        e2 += e;
        count++;
    }

    d = count;

    cout << p << " " << q << " " << phi << " " << d << endl;

    int M = 1;
    int exponent = d;
    int base = 0;

    //We decode the message to an int using exponentiation by squaring
    for (int i = 0; i < m; i++) {
        M = 1;
        exponent = d;
        base = message.at(i);
        while (exponent > 0) {
            if (exponent % 2 == 1) {
                M = (M * base) % n;
            }
            base = (base * base) % n;
            exponent = exponent / 2;
        }
        message.at(i) = M;
        cout << M;
        if (i < m) {
```

```
            cout << "-";
        }
    }

    cout << endl;


    //Calls functions that would output letter depending on decoded integer
    for (int i = 0; i < m; i++) {
        M = message.at(i);
        decodedMessage(M);

}
    }
    return 0;
}

void decodedMessage(int integerMessage) {
    // Map the decoded integer to the corresponding ASCII value
    char decodedChar;

    if (integerMessage == 7) {
        decodedChar = 'A';
    } else if (integerMessage == 8) {
        decodedChar = 'B';
    } else if (integerMessage == 9) {
        decodedChar = 'C';
    } else if (integerMessage == 10) {
        decodedChar = 'D';
    } else if (integerMessage == 11) {
        decodedChar = 'E';
    } else if (integerMessage == 12) {
        decodedChar = 'F';
    } else if (integerMessage == 13) {
        decodedChar = 'G';
    } else if (integerMessage == 14) {
        decodedChar = 'H';
    } else if (integerMessage == 15) {
        decodedChar = 'I';
    } else if (integerMessage == 16) {
        decodedChar = 'J';
    } else if (integerMessage == 17) {
        decodedChar = 'K';
    } else if (integerMessage == 18) {
        decodedChar = 'L';
    } else if (integerMessage == 19) {
        decodedChar = 'M';
    } else if (integerMessage == 20) {
        decodedChar = 'N';
    } else if (integerMessage == 21) {
        decodedChar = 'O';
    } else if (integerMessage == 22) {
        decodedChar = 'P';
    } else if (integerMessage == 23) {
        decodedChar = 'Q';
    } else if (integerMessage == 24) {
        decodedChar = 'R';
    } else if (integerMessage == 25) {
        decodedChar = 'S';
    } else if (integerMessage == 26) {
        decodedChar = 'T';
    } else if (integerMessage == 27) {
        decodedChar = 'U';
    } else if (integerMessage == 28) {
        decodedChar = 'V';
```

```
    } else if (integerMessage == 29) {
        decodedChar = 'W';
    } else if (integerMessage == 30) {
        decodedChar = 'X';
    } else if (integerMessage == 31) {
        decodedChar = 'Y';
    } else if (integerMessage == 32) {
        decodedChar = 'Z';
    } else if (integerMessage == 33) {
        decodedChar = ' ';
    } else if (integerMessage == 34) {
        decodedChar = '"';
    } else if (integerMessage == 35) {
        decodedChar = ',';
    } else if (integerMessage == 36) {
        decodedChar = '.';
    } else if (integerMessage == 37) {
        decodedChar = '\'';
    }

    cout << decodedChar;
}
```

(d)

```
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 26 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) ”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.”

I think this means that he can’t figure out how to use his telephone so he can only do this in RSA form, and the one who said it is Bob.

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, $0 \le x \le 40$, that satisfies the following congruence: $31x + 54 \equiv 16 \pmod{41}$. Show your work. You should not use a brute force approach.

Solution 3:
(a) $5^{1627} \pmod{12}$
$5^{1627} \equiv 5^{2*813+1} \pmod{12}$

$\equiv (5^2)^{813}$
$\equiv (25)^{813} * 5$
$\equiv (25 \pmod{12})^{813} * 5$
$\equiv 1^{813} * 5$
$\equiv 5 \pmod{12}$

(b)
Multiples of 8:
$8, 16, 24, 32, 40, 48, 56, 64, 72, 80, 88, 96, 104, 112, 120$
Multiples of 17:
$17, 34, 51, 68, 85, 102, 119$
Since gcd(8,11)=1, the theorem implies that $7^-1(\pmod{11}$ exists. We then need to find α and β such that
$\alpha * 8 + \beta * 17 = 1$
So $8 * 15 + (-7) * 17 = 1$
And this gives us that $8^{-1} \pmod{17} = 15$

(c)
Using $8^{16} \equiv 1(mod 17)$
$8^{16} * 8^{-1} \equiv 1 * 8^{-1}(\pmod{17})$
$8^{-1} \equiv 8^{15}$
$8^-1 \equiv 8^{15}$
$8 * (8^2)^7$
$\equiv 8 * (64 \pmod{17})^7$
$\equiv 8 * (13)^7$
$\equiv 8 * 13 * (13)^6$
$\equiv 8 * 13 * (913^2)^3$
$8 * 13 * (169 \pmod{17})^3$
$\equiv 8 * 13 * (16)^3$
$\equiv 8 * 13 * 16 * (16^2)$
$\equiv 8 * 13 * 16 * 256$
$\equiv 104 * 16 * 256$
$\equiv 104 \pmod{17} * 16 * 256$
$\equiv 2 * 16 * 256$
$\equiv 32 * 256$
$\equiv 15 * 256 \pmod{17} \equiv 15 * 1$
$\equiv 15 \pmod{17}$

(d) $8^{-11} \equiv 1 * 8^{-11} \pmod{17}$
$\equiv 8^{16} * 8^{-11} \pmod{17}$
$\equiv 8^5 \pmod{17}$
$\equiv (8^2)^2 * 8 \pmod{17}$
$\equiv (64 \pmod{17})^2 * 8$
$\equiv (13^2) * 8$
$\equiv (169 \pmod{17}) * 8$
$\equiv 16 * 8$
$\equiv 128 \pmod{17}$
$\equiv 9 \pmod{17}$

(e)
$31x + 54 = 16 \pmod{41}$

$31x + 54 \pmod{41} = 16 \pmod{41}$
$31x + 13 = 16 \pmod{41}$
$31x = 16 - 13 \pmod{41}$
$31x = 3 \pmod{41}$
$31^{-1} * 31x = 3 \pmod{41}$
$\Rightarrow x = 3 * 31^{-1} \pmod{41}$
31^{-1} exists because gcd(31,41) is 1.
We need to find α and β such that $\alpha * 31 + \beta * 41 = 1$
Multiples of 31:
$31, 62, 93, 124$
Multiples of 41:
$41, 82, 123$
So $\alpha = 4, \beta = -3 =: 4 * 31 + (-3) * 41 = 1$
And this gives us $31^{-1} \pmod{41} = 4$
$x = 3 * 31^{-1} \pmod{41}$
$x = 3 * 4$
$x = 12$

Academic integrity declaration. The homework papers must include at the end an academic integrity declaration. This should be a brief paragraph where you state *in your own words* (1) whether you did the homework individually or in collaboration with a partner student (if so, provide the name), and (2) whether you used any external help or resources.

We helped each other, this was a partner collaboration as seen with the names and SID, partners names are Ben Pham and Gokul Nookula. We had help from Alice Thai, who was the TA, and from YouTube to explain more about the concept. We also looked at the concepts from the lecture notes and the discussion slides to help us with our problem. About coding, we used VSCode and the autograder to check the code.

Submission. To submit the homework, you need to upload the pdf and cpp files to Gradescope. If you submit with a partner, you need to put two names on the assignment and submit it as a group assignment.

Reminders. Remember that only LaTeX papers are accepted.