Homework 2: Written Solutions

Written Part

5. Let's begin by proving correcntess of the algorithms from the implementation part.

   (a) Prove the correctness of `getBinary` from Problem 2. Explicitly, prove that the list $[A_0, \cdots, A_r]$ it returns satisfies conditions (i)-(iii).

   Proof. Condition (i) says that each A_i is either 0 or 1. As each i is the remainder of a division with remainder by 2, this is easily seen to hold.

   For the other two parts we highlight the following important statement, which is clear from the definition of the algorithm.

   The algorithm precisely in the iteration where division with remainder has quotient 0.

   To see that $A_r = 1$ it suffices to show it is not zero. To do this suppose that we set $A_i = 0$ in step (3). This means that in the division with remainder step (2) we had remainder zero so that $x = 2q$. For the algorithm to terminate q must be 0, but then $x = 0$, in which case the algorithm would have terminated in the previous iteration. In particular, we see that if the algorithm terminates at step r then $A_r = 1$, proving condition (iii) holds.

   For the last step we will prove the following claim by induction.

   Claim 1. *If the algorithm terminates after N steps, then $A = A_0 + A_1 * 2 + \cdots + A_N * 2^N$.*

   Proof. For the base case we let $N = 0$. Then in the division with remainder step we have quotient 0, so that $x = 2q + r$ becomes $A = A_0$ and we are done.

   For the general step, we assume that the algorithm works if it terminates in $N-1$ steps. We run the first iteration of the loop to get $A = 2q + A_0$. Then when we repeat the loop a second time we are running the `getBinary` on q. Furthermore, it will take $N-1$ steps to terminate (as they are the same steps as A but starting at the second iteration). In particular, by the inductive hypothesis we may assume that computes the binary expansion of q, that is:

   $$q = A_1 + A_2 * 2 + \cdots + A_N * 2^{N-1}$$

   Plugging into the division from the first step we have

   $$A = A_0 + 2q = A_0 + A_1 * 2 + \cdots + A_n * 2^N,$$

   as desired. □

   The claim precisely proves that condition (ii) holds and we have now proved the correctness of our algorithm. □

   (b) Analyze the time of complexity `getBinary`. What is the maximum amount of times it performs division with remainder? Justify your answer.

Proof. It performs division with remainder precisely $r+1$ times. Furthermore, r is the largest number such that $2^r \leq A$. In particular, $r = \lfloor \log_2 A \rfloor$. So the algorithm takes precisely $\lfloor \log_2 A \rfloor + 1$ steps. $\square$

(c) Prove that the correctness of `fastPowerSmall` from 3(b). Explicitly, prove that it correctly returns an element $b \in \{0, \cdots, N-1\}$ such that $g^A \equiv b \mod N$.

Proof. Let the binary expansion of A be $[A_0, A_1, \cdots A_r]$ as in part (a). We will by induction show that on the i'th step of the algorithm, the following 3 things hold.

i. At the beginning $A \equiv A_i \mod 2$
ii. At the beginning $a = g^{2^i}$
iii. At the end $b \equiv g^{A_0 * 2^0 + A_1 * 2^1 + \cdots + A_i * 2^i} \mod N$

The base case is clear: $A \mod 2$ is the first binary digit, and a was set to 1, so that b is multiplied by g if $A_0 = 1$ and by 1 if $A_0 = 0$, so that $b = g^{A_0}$. For the inductive step, the fct that $A \mod 2$ is the i'th binary digit follows from the proof of part (a), as in each step we are replacing A with $\lfloor A/2 \rfloor$, which is precisely how we computed its binary expansion. For (ii) we observe that by induction, we finished the $i-1$st step doing $a^2 = (g^{2^{i-1}})^2 = g^{2^i}$. For (iii) then observe that multiplying by a if and only if $A_i = 1$, is exactly multiplying by $a^{A_0} = g^{A_i * 2^i}$. Therefore by induction we have:

$$b = g^{A_0 * 2^0 + A_1 * 2^1 + \cdots + A_{i-1} * 2^{i-1}} g^{A_i * 2^i} = g^{A_0 * 2^0 + A_1 * 2^1 + \cdots + A_i * 2^i}.$$

This completes the inductive proof of i,ii,iii. To conclude, we observe that exactly as in part (a), the algorithm terminates after r steps, so that we return:

$$b = g^{A_0 * 2^0 + A_1 * 2^1 + \cdots + A_r * 2^r} = g^A \mod N.$$

$\square$

(d) Analyze the time complexity of `fastPowerSmall`. What is the maximum amount of times it runs steps (2)-(4)? Justify your answer.

Proof. The algorithm essentially adds the other two parts of fast powering into the binary expansion algorithm, so as before we run in $r + 1 = \lfloor \log_2 A \rfloor + 1$ steps. $\square$

(e) Write a couple of sentences informally discussing the space advanges of `fastPowerSmall` over `fastPower`.

In `fastPower` we needed to save the entire binary expansion of A (say of r bits), as well as all the powers g^{2^i} for $0 \leq i \leq r$. In particular, we need two lists of length $r + 1 = \lfloor \log_2 A \rfloor + 1$. Furthermore, the list of squares will consist of elements of $\mathbb{Z}/N\mathbb{Z}$ where N can be quite large. If N, A are on the order of 1000 bits, the list of squares has approximately 1000 elements of 1000 bits, so a million bits of data that needs to be remembered. As N and A increase the memory use grows on the order of the product of the number of bits of each, which is not ideal for scaling. As we can see in `fastPowerSmall`, we only need to remember one thing at a time in order to compute as we go, which is much better and completely avoids any memory issues.

6. Let $\{p_1, \cdots, p_r\}$ be a set of prime numbers and let $a = p_1 p_2 \cdots p_r + 1$. Show that a has a prime divisor which is not equal to any of the p_i. Use this fact to conclude that there are infinitely many prime numbers. (We should point out that this proof appears in Euclid's *Elements*, and is quite literally thousands of years old!)

   Proof. Let q be any prime divisor of a (which we know exists by the fundamental theorem of arithmetic). Then $a \equiv 0 \mod q$. Nevertheless, for each i it is clear that $a \equiv 1 \mod p_i$. Therefore $p_i \neq q$.

   If there were finitely many primes, we could put them in a all in a finite list as above. But we just showed that all such finite lists are incomplete. □

7. Recall that in the $\mathbb{Z}$, you couldn't always divide by 2, but you could *sometimes* divide by 2. We can explain this phenomenon by saying $2x = b$ has a solution in $\mathbb{Z}$ precisely when b is even. Let's study this problem in the modular setting. Consider the congruence

$$ax = c \mod m.$$

   (a) Prove that there is a solution if and only if $\gcd(a, m)$ divides c.

   Proof. Suppose there were some x solving the congruence, so that there is an integer k with $ax = c + km$. We can rewrite this as $ax - km = c$. Letting $g = \gcd(a, m)$, we notice $g|a$ so it divides ax, and similarly $g|km$, so it divides the difference, which is c.

   Conversely, let $g = \gcd(a, m)$ divides c, so that there is some l such that $gl = c$. By the extended Euclidean algorithm there are u, v such that $au + mv = g$. Multiplying through by l gives $aul + mvl = gl = c$, so that $aul \equiv c \mod m$. Thus ul solves the congruence. □

   (b) In the case of $\mathbb{Z}$, the solution is unique. Here this need not be the case. Prove that if there is one solution to the congruence, there are precisely $\gcd(a, m)$ many solutions in $\mathbb{Z}/m\mathbb{Z}$.

   Proof. Let $g = \gcd(a, m)$. We will show that if x_0 is a solution to the congruence, than the other are precisely elements of the form $x = x_0 + km/g$ for some $k \in \mathbb{Z}$. Notice that saying x_0 is a solution to the congruence implies that $ax_0 + l_0 m = c$ for some $l_0 \in \mathbb{Z}$.

   First suppose x is of the given form. Then we have:

$$\begin{aligned} ax - c &= a(x_0 + km/g) - c \\ &= ax_0 - c + \frac{ka}{g}m \\ &\equiv 0 \mod m \end{aligned}$$

   Therefore all such x solve the congrunce. Conversely suppose that $ax \equiv c \mod m$. Then $ax \equiv ax_0 \mod m$ since they are both congruent to c. Therefore there is some $l \in \mathbb{Z}$ such that $ax = ax_0 + ml$, which we rearrange to to $a(x - x_0) = ml$. Dividing through by g gives:

$$\frac{a}{g}(x - x_0) = \frac{m}{g}l. \tag{1}$$

In particular $\frac{a}{g}|\frac{m}{g}l$. Since $\gcd\left(\frac{a}{g}, \frac{m}{g}\right) = 1$, we must have $\frac{a}{g}|l$ (as in HW1 Problem 7(c)), so that there is some k with $l = k\frac{a}{g}$. Plugging this back into equation 1 gives

$$\frac{a}{g}(x - x_0) = k\frac{a}{g}\frac{m}{g},$$

and solving for x gives $x = x_0 + km/g$ as desired.
Now notice that the solutions are precisely the distinct elements

$$\{x_0, x_0 + m/g, x_0 + 2m/g, ..., x_0 + (g-1)m/g\},$$

which consists of exactly g elements. □

(c) Check that this works by finding all the solutions to the following congruences:

i. $2x \equiv 4 \mod 6$
We list the multiples of 2 in $\mathbb{Z}/6\mathbb{Z}$.

x	0	1	2	3	4	5
$2x$	0	2	4	0	2	4

We see the solutions are $x = 2, 5$, and as expected $\gcd(2, 6) = 2$.

ii. $3x \equiv 4 \mod 9$
We list the multiples of 3 in $\mathbb{Z}/9\mathbb{Z}$.

x	0	1	2	3	4	5	6	7	8
$2x$	0	3	6	0	3	6	0	3	6

We see that there is no solution to the equation, which was expected as $\gcd(3, 9) = 3$ does not divide 4.

iii. $12x \equiv 15 \mod 21$
We will find one solution and use the proof of part (b) to enumerate the rest. Notice that $g = \gcd(12, 21) = 3$ which divides 15 so we should expect 3 solutions. We also notice $12 \cdot 3 = 36 = 21 + 15 \equiv 15 \mod 21$. So 3 is a solution. By the proof of the theorem, the rest of the solutions look like $3 + km/g = 3 + 21k/3 = 3 + 7k$ for various k. So we have $x = 3, 10, 17$ (the next one is 24 which repeats back to 3). And indeed we check that $12 \cdot 10 = 120 = 105 + 15 = 21 * 5 + 15 \equiv 15$ and similarly for 17.

8. This homework assignment concludes with a study of square roots modulo p for some prime p (adapted from Exercises 1.36 and 1.39 in [HPS]). We will see that the behaviour is similar to that of $\mathbb{Z}$, where every number has either 2 square roots, or none at all.

(a) Let p be prime, and suppose $p \neq 2$, and $b \in \mathbb{Z}/p\mathbb{Z}$ a nonzero element. Show that b either has 2 square roots modulo p, or none. That is, show that the congruence

$$x^2 \equiv b \mod p$$

has either 2 or 0 solutions in $\mathbb{Z}/p\mathbb{Z}$. What happens if $p = 2$?

Proof. Suppose b has one square root mod p (say x). Then $p - x \not\equiv x \mod p$ since p is odd, and

$$(p - x)^2 \equiv (-x)^2 \equiv x^2 \equiv b \mod p,$$

so if there is one there are at least 2 solutions, namely $\pm x$ or equivalently x and $p-x$. We suppose y were another square root of b. Then $x^2 - y^2$ is divisible by p. But $x^2 - y^2 = (x-y)(x+y)$ so that either $p|x-y$ (whence $x \equiv y \mod p$) or else p divides $x+y$ (whence $y \equiv -x \mod p$). But these are precesely the two solutions previously enumerated.

Notice that if $p = 2$ then $-x \equiv -x + 2x \equiv x \mod p$ so the two solutions $\pm x$ are in fact the same. $\square$

(b) Test out this works by finding all solutions to the following congruences:

i. $x^2 = 2 \mod 7$

ii. $x^2 = 3 \mod 11$

We enumerate all the squares $\mod 7$.

$$\begin{aligned} 0^2 &= 0 \\ 1^2 &= 1 \\ 2^3 &= 4 \\ 3^2 &= 9 \equiv 2 \\ 4^2 &= 14 \equiv 2 \\ 5^2 &= 25 \equiv 4 \\ 6^2 &= 36 \equiv 1 \end{aligned}$$

By inspection we see that the square roots of 2 are 3 and 4. Since $4 = 7 - 3 \equiv -3$ this agrees. We similarly can list the squares mod eleven:

a	0	1	2	3	4	5	6	7	8	9	10
a^2	0	1	4	9	5	3	3	5	9	4	1

We see both 5 and 6 are square roots of 3, and $6 = 11 - 5 \equiv -5 \mod 11$.

(c) Much like the integers, if a number has a square root modulo p, we call it a *perfect square* modulo p. Find all the perfect squares in $\mathbb{Z}/5\mathbb{Z}$ and $\mathbb{Z}/7\mathbb{Z}$.

We did $\mathbb{Z}/7\mathbb{Z}$ in the previous problem, getting $0, 1, 2$ and 4. For $\mathbb{Z}/5\mathbb{Z}$ we see

$$\{0^2, 1^2, 2^2, 3^2, 4^2\} = \{0, 1, 4, 4, 1\} = \{0, 1, 4\}.$$

Notice that in each case exactly half of the nonzero elements were perfect squares (suggesting that perfect squares are more common mod p than in $\mathbb{Z}$). Let's prove this in general.

(d) Let p be an odd prime and $g \in \mathbb{Z}/p\mathbb{Z}$ be a primitive root. Then any $a \in (\mathbb{Z}/p\mathbb{Z})^*$ is congruent to a power of g modulo p, say $a \equiv g^k \mod p$. Show that a is a perfect square if and only if k is even. Conclude that exactly half of the nonzero elements of $\mathbb{Z}/p\mathbb{Z}$ are perfect squares.

Proof. We may assume $0 \le k \le p-2$. Suppose $k = 2l$ is even. Then

$$a \equiv g^k = g^{2l} = (g^l)^2 \mod p,$$

so a is a square. Conversely, suppose a is a square, say $a \equiv b^2 \mod p$. Since g is a primitive root, $b = g^t$ for some $0 \le t \le p-2$. Then $a \equiv g^{2t}$. Notice that $0 \le$

$2t \leq 2p-4$. If $2t \leq p-2$ then $2t = k$ (since $\{g^0, g^1, \cdots, g^{p-2}\}$ are all distinct), and so k is even. Otherwise $2t-(p-1) \in \{0, 1, \cdots, p-2\}$. By Fermat's little theorem $g^{-(p-1)} \equiv (g^{p-1})^{-1} \equiv 1$. Therefore we compute

$$g^k \equiv g^{2t} \equiv g^{2t}g^{-(p-1)} \equiv g^{2t-(p-1)} \mod p.$$

Therefore, as above $k = 2t-(p-1)$. As p is an odd prime, $p-1$ is even so k is the difference of even numbers and therefore even.

Since exactly half of the elements of $(\mathbb{Z}/p\mathbb{Z})^* = \{g^0, g^1, \cdots, g^{p-1}\}$ are even powers of g, we have shown that exactly half are squares. □

With these tools in hand we can prove the following variant of Fermat's little theorem.

(e) For each prime p such that $3 \leq p < 20$, compute $b^{(p-1)/2} \mod p$ for various nonzero values of $b \in \mathbb{Z}/p\mathbb{Z}$ (for example, $b = 2$). Feel free to use the algorithms you wrote in the implementation part. You should notice a pattern, make a conjecture as to the possible values of $b^{(p-1)/2} \mod p$ for $p \not| b$, and prove it is correct. (**Hint:** Use Fermat's little theorem and part (a) above.)

Using the fast powering algorithm (or even slow powering since numbers are small) we compute:

p	3	5	7	11	13	17	19
$2^{\frac{p-1}{2}}$	2	4	1	10	12	1	18
$3^{\frac{p-1}{2}}$	n/a	4	6	1	1	16	18
$4^{\frac{p-1}{2}}$	1	1	1	1	1	1	1
$5^{\frac{p-1}{2}}$	2	n/a	6	1	12	16	1
$6^{\frac{p-1}{2}}$	n/a	1	6	10	12	16	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Perhaps pattern is becoming clear. It seems as though $b^{(p-1)/2}$ is either 1, or else it is $p-1 \equiv -1$, so this will be our conjecture. You may have notice that in the case of 4, when we have $4^{(p-1)/2} = 1$ in each of the primes. In fact, trying to explain this suggests the proof of our conjecture. Indeed:

$$4^{(p-1)/2} = (2^2)^{(p-1)/2} = 2^{p-1} \equiv 1 \mod p,$$

where the last step is Fermat's little theorem. Notice that once we square something to the $(p-1)/2$ power we get 1 by Fermat's little theorem. We can now state and proof our conjecture.

Theorem 1. *Let p be an odd prime and $b \in (\mathbb{Z}/p\mathbb{Z})^*$. Then $b^{(p-1)/2} \equiv \pm 1 \mod p$.*

Proof. Notice that by Fermat's little theorem

$$(b^{(p-1)/2})^2 = b^{p-1} \equiv 1 \mod p,$$

so that $b^{(p-1)/2}$ must be a square root of 1. By part (a), there are only 2 of these, and as ± 1 both square to 1, these these are the only possibilities.. □