CS 170 Algorithms
Spring 2014 Elchanan Mossel
HW 2

1. (20 pts.) (Quadratic Residues)

(a) **(8 points).** Since a is a non-zero quadratic residue $\mod N$, there exists $x \in \{1,\ldots,N-1\}$ such that $x^2 \equiv a \mod N$ (because $a \neq 0$ implies we cannot have $x = 0$). Notice that we also have $(N-x)^2 \equiv x^2 \mod N \equiv a \mod N$, and also that $(N-x) \in \{1,\ldots,N-1\}$. We will now show that when N is odd, $x \neq N-x$. For if not, then we have $x = N-x$, so that $N = 2x$ which contradicts the fact that N is odd. Thus, we have at least two distinct values for $x \in \{1,2,\ldots,N-1\}$ which satisfy $x^2 \equiv a \mod N$.

We will now show that these are the only possible solutions. Consider any other value $y \in \{1,2,\ldots,N-1\}$ such that $y^2 \equiv a \mod N$. We then have $x^2 \equiv y^2 \mod N$, so that N divides $x^2 - y^2 = (x-y)(x+y)$. Since N is a prime, this implies that N divides at least one of $x-y$ and $x+y$. We now consider both these cases.

N **divides** $x-y$**.** If N divides $x-y$, then we see that $x \equiv y \mod N$ which implies that $x = y$, since both x and y are in the set $\{1,2,\ldots,N-1\}$.

N **divides** $x+y$**.** Since both x and y are in $\{1,2,\ldots,N-1\}$, we have $0 < x+y < 2N$, so that N can divide $x+y$ if and only if $x+y = N$. But then we have $y = N-x$.

We therefore see that x and $N-x$ are the only possible solutions. This completes the proof.

(b) **(6 points).** Clearly 0 is a quadratic residue since $0^2 = 0 \mod N$. Now let $S = \{1,2,\ldots,(N-1)/2\}$. Notice that since N is odd, we have $|S| = (N-1)/2$. From the first part, we know that for an odd prime N, and for $x,z \in \{1,2,\ldots,N-1\}$, we have $x^2 \equiv z^2 \mod N$ if and only if either $x = z$ or $x+z = N$. Since $x+z < N$ for $x,z \in S$, this implies that for two distinct elements x and z in S, we have $x^2 \not\equiv z^2 \mod N$. We therefore conclude that the set $T = \{x^2 | x \in S\}$ is of size $|S| = (N-1)/2$. Since every element of T is a non-zero quadratic residue, this shows that there are at least $(N-1)/2$ quadratic residues.

We will now show that all the non-zero quadratic residues are contained in T. Consider any non-zero quadratic residue $a \equiv y^2 \mod N$, where $y \in \{1,2,\ldots,N-1\}$. If $y \in S$, then $a \in T$, by definition. Otherwise, we have $(N+1)/2 \leq y \leq N-1$, so that $N-y \in S$, and hence $a \equiv (N-y)^2 \mod N$ is again in T. Thus, the set T contains all the non-zero quadratic residues. We therefore get that the number of all quadratic residues is $|\{0\} \cup T| = 1 + (N-1)/2 = (N+1)/2$, as required.

(c) **(6 points).** Clearly we cannot take $N = 2$ or an odd prime. We take $N = 8$, and notice that for any odd number b, $b^2 \equiv 1 \mod 8$. Thus, the equation $x^2 = 1 \mod 8$ has *four* solutions $\{1,3,5,7\}$ in the set $\{0,1,2,\ldots,7\}$.

2. (10 pts.) (These exponents are large, they contain multitudes)

(a) **(4 points).** We first note the prime factorization of 35: $35 = 5 \times 7$. Thus, an integer M is divisible by 35 if and only if M is divisible by *both* 5 and 7. We also know from Fermat's

little theorem that for a coprime to 5, we have $a^4 \equiv 1 \mod 5$. Using this, we now calculate:

$$2014^{2015} \mod 5 \equiv 4^{2012+3} \mod 5 \equiv \left(4^{503}\right)^4 \cdot 4^3 \mod 5 \equiv 1 \cdot 4^3 \mod 5 \equiv 4 \mod 5. \quad (1)$$

Similarly,

$$2012^{2013} \mod 5 \equiv 2^{2012+1} \mod 5 \equiv \left(2^{503}\right)^4 \cdot 2^1 \mod 5 \equiv 1 \cdot 2 \mod 5 \equiv 2 \mod 5. \quad (2)$$

Denoting $2014^{2015} - 2012^{2013}$ by M, we then combine eqs. (1), (2) to get that $M \equiv 4-2 \equiv 2 \mod 5$. Thus, 5 does not divide M, and hence 35 does not either.

(b) **(6 points).** From Fermat's little theorem, we know that if a is coprime to 5, we have $a^4 \equiv 1 \mod 5$ (since 5 is a prime). To use this fact, we would like to represent $E := 170^{70}$ as $4s+t$, for some positive integer s and some $t \in \{0,1,2,3\}$. Given such a representation, we would get from Fermat's little that

$$2^E \mod 5 \equiv (2^s)^4 \cdot 2^t \mod 5 \equiv 1 \cdot 2^t \mod 5. \quad (3)$$

In order to determine $t \equiv E \mod 4$, we note that $E = 170^{70} = (2 \cdot 85)^{2 \cdot 35} = \left((2 \cdot 85)^2\right)^{35}$, so that

$$E \mod 4 \equiv \left((2 \cdot 85)^2\right)^{35} \mod 4 \equiv \left(4 \cdot 85^2\right)^{35} \mod 4 \equiv 0 \mod 4.$$

Thus, we get that $t = 0$. Substituting this in eq. (3), we get that $2^E \equiv 1 \mod 5$, so that the remainder when it is divided by 5 is 1.

3. (10 pts.) (No compromises)

Since d is the multiplicative inverse of $e \mod (p-1)(q-1)$, we know that,

$$ed - 1 = 0 \mod (p-1)(q-1),$$

which means for some positive integer k,

$$ed - 1 = k(p-1)(q-1) \quad (4)$$
$$\implies k = \frac{ed-1}{(p-1)(q-1)}$$

Since $d < (p-1)(q-1)$, and $e = 3$, we get

$$k < \frac{3(p-1)(q-1)-1}{(p-1)(q-1)} < 3 - \frac{1}{(p-1)(q-1)},$$

which implies that $k \in \{1,2\}$. Now, for each $k \in \{1,2\}$ we can use eq. (4), which now has only 2 unknowns (p and q) in conjunction with equation $N = pq$, to solve for p and q. To do this, consider one of the two fixed values of k, and substitute $N = pq, e = 3$ into eq. (4) to obtain the following quadratic equation for p:

$$kp^2 - c_k p + kN = 0, \text{ where } c_k := k(N+1) - (3d-1),$$

which has the solutions

$$p_k = \frac{c_k \pm \sqrt{c_k^2 - 4k^2N}}{2k}.$$

Using the result of Problem 2, we can efficiently determine if the solutions p_k are integral for a given value of k. Our arguments above show that at least one of the p_k's will yield the desired non-trivial factors of N.

4. (22 pts.) (Recurrences)

a) $T(n) = 2T(n/3) + 1 = \Theta(n^{\log_3 2})$ by the Master theorem.

b) $T(n) = 5T(n/4) + n = \Theta(n^{\log_4 5})$ by the Master theorem.

c) $T(n) = 7T(n/7) + n = \Theta(n \log_7 n)$ by the Master theorem.

d) $T(n) = 9T(n/3) + n^2 = \Theta(n^2 \log_3 n)$ by the Master theorem.

e) $T(n) = 8T(n/2) + n^3 = \Theta(n^3 \log_2 n)$ by the Master theorem.

f) $T(n) = 49T(n/25) + n^{3/2} \log n = \Theta(n^{3/2} \log n)$. Apply the same reasoning of the proof of the Master Theorem. The contribution of level i of the recursion is

$$\left(\frac{49}{25^{3/2}}\right)^i n^{3/2} \log\left(\frac{n}{25^{3/2}}\right) = \left(\frac{49}{125}\right)^i O(n^{3/2} \log n)$$

Because the corresponding geometric series is dominated by the contribution of the first level, we obtain $T(n) = O(n^{3/2} \log n)$. But, $T(n)$ is clearly $\Omega(n^{3/2} \log n)$. Hence, $T(n) = \Theta(n^{3/2} \log n)$.

g) $T(n) = T(n-1) + 2 = \Theta(n)$.

h) $T(n) = T(n-1) + n^c = \sum_{i=0}^{n} i^c + T(0) = \Theta(n^{c+1})$.

i) $T(n) = T(n-1) + c^n = \sum_{i=0}^{n} c^i + T(0) = \frac{c^{n+1}-1}{c-1} + T(0) = \Theta(c^n)$.

j) $T(n) = 2T(n-1) + 1 = \sum_{i=0}^{n-1} 2^i + 2^n T(0) = \Theta(2^n)$.

k) $T(n) = T(\sqrt{n}) + 1 = \sum_{i=0}^{k} 1 + T(b)$, where $k \in Z$ such that $n^{\frac{1}{2^k}}$ is a small constant b, i.e. the size of the base case. This implies $k = \Theta(\log \log n)$ and $T(n) = \Theta(\log \log n)$.

5. (12 pts.) (Squaring a Matrix)

a) **(4 points).**

$$\begin{bmatrix} a & b \\ c & d \end{bmatrix}^2 = \begin{bmatrix} a^2+bc & b(a+d) \\ c(a+d) & bc+d^2 \end{bmatrix}$$

Hence the 5 multiplications $a^2, d^2, bc, b(a+d)$ and $c(a+d)$ suffice to compute the square.

b) **(4 points).** We do get 5 subproblems but they are *not of the same type as the original problem*. Note that we started with a squaring problem for a matrix of size $n \times n$ and three of the 5 subproblems now involve *multiplying* $n/2 \times n/2$ matrices. Hence the recurrence $T(n) = 5T(n/2) + O(n^2)$ does not make sense.

c) **(4 points).** Given two $n \times n$ matrices X and Y, create the $2n \times 2n$ matrix A:

$$A = \begin{bmatrix} 0 & X \\ Y & 0 \end{bmatrix}$$

It now suffices to compute A^2, as its upper left block will contain XY:

$$A = \begin{bmatrix} XY & 0 \\ 0 & XY \end{bmatrix}$$

Hence, the product XY can be calculated in time $O(S(2n))$. If $S(n) = O(n^c)$, this is also $O(n^c)$.

6. (26 pts.) (Closest Pair)

a) **(4 points).** Suppose 5 or more points in L are found in a square of size $d \times d$. Divide the square into 4 smaller squares of size $\frac{d}{2} \times \frac{d}{2}$. At least one pair of points must fall within the same smaller square: these two points will then be at distance at most $\frac{d}{\sqrt{2}} < d$, which contradicts the assumption that every pair of points in L is at distance at least d.

b) **(10 points).** The proof is by induction on the number of points. The algorithm is trivially correct for two points, so we may turn to the inductive step. Suppose we have n points and let (p_s, p_t) be the closest pair. There are three cases.
If $p_s, p_t \in L$, then $(p_s, p_t) = (p_L, q_L)$ by the inductive hypothesis and all the other pairs tested by the algorithm are at a larger distance apart, so the algorithm will correctly output (p_s, p_t). The same reasoning holds if $p_s, p_t \in R$.
If $p_s \in L$ and $p_t \in R$, the algorithm will be correct as long as it tests the distance between p_s and p_t. Because p_s and p_t are at distance smaller than d, they will belong to the strip of points with x-coordinate in $[x-d, x+d]$. Suppose that $y_s \leq y_t$. A symmetric construction applies in the other case. Consider the rectangle S with vertices $(x-d, y_s), (x-d, y_s+d), (x+d, y_s+d), (x+d, y_s)$. Notice that both p_s and p_t must be contained in S. Moreover, the intersection of S with L is a square of size $d \times d$, which, by a), can contain at most 4 points, including p_s. Similarly, the intersection of S with R can also contain at most 4 points, including p_t. Because the algorithm checks the distance between p_s and the 7 points following p_s in the y-sorted list of points in the middle strip, it will check the distance between p_s and all the points of S. In particular, it will check the distance between p_s and p_t, as required for the correctness of the algorithm.

c) **(8 points).** When called on input of n points this algorithm first computes the median x value in $O(n)$ and then splits the list of points into those belonging to L and R, which also takes time $O(n)$. Then the algorithm can recurse on these two subproblems, each over $n/2$ points. Once these have been solved the algorithm sorts the points in the middle strip by y coordinate, which takes time $O(n \log n)$ and then computes $O(n)$ distances, each of which can be calculated in constant time. Hence the running time is given by the recursion $T(n) = 2T(\frac{n}{2}) + O(n \log n)$. This can be analyzed as in the proof of the Master theorem. The kth level of the recursion tree will contribute $t_k = 2^k \frac{n}{2^k}(\log n - k)$. Hence, the total running time will be:

$$\sum_{k=0}^{\log n} t_k = n \log^2 n - n \sum_{k=0}^{\log n} k \leq n \log^2 n - \frac{n}{2} \log^2 n = O(n \log^2 n)$$

.

d) **(4 points).** We can save some time by sorting the points by y-coordinate only once and making sure that the split routine is implemented as not to modify the order by y when splitting by x. Sorting takes time $O(n \log n)$, while the time required by the remaining of the algorithm is now described by the recurrence $T(n) = 2T(\frac{n}{2}) + O(n)$, which yields $T(n) = O(n \log n)$. Hence, the overall running time is $O(n \log n)$.