CSE332, Spring 2012

Optional (!) Reading

Justification of Quadratic Probing with Load Factor < 1/2

[This proof is also in the textbook, but it omits a couple details that were not obvious to your instructor.]

Theorem: If an open-addressing hash table has a size TableSize that is a prime number, then quadratic probing will always find an empty hash-table position provided the table has fewer than TableSize/2 elements (i.e., the load factor is < 1/2).

Proof:

Recall that our quadratic probing function will look at index:

```
   (h(key) + i*i) % TableSize
```

for the i^th probe. Also note the theorem assumes TableSize is prime.

Because the table is less than half full, it suffices to show that the first TableSize/2 probes will never try the same table index more than once. Assume for purpose of contradiction this is not true. Then there exist j and k such that

1. j != k
2. j and k are greater than 0 and less than TableSize/2
3. (h(key) + j*j) % TableSize == (h(key) + k*k) % TableSize

Given (3), we can derive:

```
A.   (j*j) % TableSize == (k*k) % TableSize
      by subtracting h(key) % TableSize from both sides

B.   (j*j - k*k) % TableSize == 0
      by subtracting (k*k) % TableSize from both sides

C.  ((j+k)*(j-k)) % TableSize == 0
      by factoring
```

From (C), we can claim that either (j+k) % TableSize == 0 or (j-k) % TableSize == 0 because for all x, y, and p, if p is prime and x*y % p == 0, then x % p == 0 or y % p == 0. We prove this lemma below. (In this case, let x = j+k, y = j-k, and p = TableSize.) But it cannot be that (j-k) % TableSize == 0 because (1) ensures j!=k and (2) ensures j and k cannot differ by a factor of TableSize (since they are both less than TableSize / 2). So the only remaining possibility is (j+k) % TableSize == 0. But (2) also ensures j+k must be less than TableSize and greater than 0, so we have a contradiction. Therefore, the first TableSize/2 probes never repeat a table index.

Lemma: For all non-negative integers x, y, and p, if p is prime and x*y % p == 0, then x % p == 0 or y % p == 0.

Proof:

Since x*y % p == 0, there exists a non-negative integer c such that x*y == c*p. If c==0, then either x==0 or y==0, so either x % p == 0 or y % p == 0. If c > 0, then consider the prime factorizations of x, y, and c -- write them as x1*x2*...*xi, y1*y2*...*yj, and c1*c2...*ck respectively. Since p is prime, its prime factorization is p. Then x1*x2*...*xi*y1*y2*...*yj == c1*c2...*ck*p. But the prime

```
factorization of any number is unique.  So at least one of the prime
factors of x or y must be p, which means x % p == 0 or y % p == 0 as
any multiple of p is 0 modulo p.

-------

Note the lemma's proof relies on p being prime.  As an example of a
non-prime, suppose x = 4, y = 15, and p = 6.  Neither 4 % 6 == 0 nor
15 % 6 == 0, but 4*15 % 6 == 0.  Notice how the prime factorization
works out with c=10:  2*2 * 3*5 = 2*5 * 2*3.  The two sides have the
same terms, but neither x nor y contain all the factors of p -- both x
and y contain some factors of p and some of c.
```