Real biquadratic fields of narrow class number 1

```
#This function computes a table of class numbers and narrow
class numbers for biquadratic fields L=Q(sqrt(p), sqrt(q)) for
primes 2 <= p < q <=n

def list_of_narrow_class_numbers_biquadratic_fields(n):
    number_of_fields = 0
    field_narrow_one = 0
    rows = []
    for p in range(2,n+1):
        for q in range(p+1,n+1):
            if ((Integer(p).is_prime())) and
((Integer(q).is_prime())):
                F.<a> = QuadraticField(p)
                R.<x> = F['x']
                if (x^2-q).is_irreducible():
                    number_of_fields = number_of_fields +1
                    K.<b> = F.extension(x^2-q)
                    L.<c> = K.absolute_field()
                    hFnarr = F.narrow_class_group().order()
                    hL = L.class_number()
                    hLnarr = L.narrow_class_group().order()
                    if hLnarr == 1:
                        field_narrow_one = field_narrow_one + 1
                    rows.append([p, q, hL, hLnarr])
    proportion =
RDF(field_narrow_one/number_of_fields)
    print "The number of biquadratic fields L = Q(sqrt(p),
sqrt(q)) with 2 <= p < q <= " + str(n) + " is " +
str(number_of_fields) + " and the number with hLnarrow = 1 is "
+ str(field_narrow_one)
    return table(rows, header_row = ["p", "q", "hLwide",
"hLnarrow"], frame = True, align = 'center')
```

```
list_of_narrow_class_numbers_biquadratic_fields(30)
```

```
The number of biquadratic fields L = Q(sqrt(p), sqrt(q)) with 2
<= p < q <= 30 is 45 and the number with hLnarrow = 1 is 6
+----+----+--------+----------+
| p  | q  | hLwide | hLnarrow |
+====+====+========+==========+
| 2  | 3  |   1    |    2     |
+----+----+--------+----------+
| 2  | 5  |   1    |    1     |
+----+----+--------+----------+
| 2  | 7  |   1    |    4     |
+----+----+--------+----------+
| 2  | 11 |   1    |    2     |
+----+----+--------+----------+
| 2  | 13 |   1    |    1     |
+----+----+--------+----------+
| 2  | 17 |   1    |    2     |
+----+----+--------+----------+
| 2  | 19 |   1    |    2     |
+----+----+--------+----------+
| 2  | 23 |   1    |    4     |
+----+----+--------+----------+
| 2  | 29 |   1    |    1     |
+----+----+--------+----------+
| 3  | 5  |   1    |    2     |
+----+----+--------+----------+
| 3  | 7  |   1    |    2     |
+----+----+--------+----------+
| 3  | 11 |   1    |    2     |
+----+----+--------+----------+
| 3  | 13 |   1    |    4     |
+----+----+--------+----------+
| 3  | 17 |   1    |    4     |
+----+----+--------+----------+
| 3  | 19 |   1    |    2     |
+----+----+--------+----------+
| 3  | 23 |   1    |    4     |
+----+----+--------+----------+
| 3  | 29 |   1    |    2     |
+----+----+--------+----------+
| 5  | 7  |   1    |    2     |
+----+----+--------+----------+
| 5  | 11 |   1    |    4     |
```

5	13	1	1
5	17	1	1
5	19	1	4
5	23	1	2
5	29	2	2
7	11	1	4
7	13	1	2
7	17	1	4
7	19	1	2
7	23	1	4
7	29	1	4
11	13	1	2
11	17	1	4
11	19	1	2
11	23	1	2
11	29	1	2
13	17	1	2
13	19	1	2
13	23	1	4
13	29	1	2
17	19	2	8
17	23	1	4

17	29	1	1
19	23	1	2
19	29	1	2
23	29	1	4

```
#This function determines the biquadratic fields L=Q(sqrt(p),
sqrt(q)) with narrow class number 1 for primes 2 <= p < q <=n

def biquadratic_fields_narrow_class_number_one(n):
    number_of_fields = 0
    field_narrow_one = 0
    rows = []
    for p in range(2,n+1):
        for q in range(p+1,n+1):
            if ((Integer(p).is_prime())) and
((Integer(q).is_prime())):
                F.<a> = QuadraticField(p)
                R.<x> = F['x']
                if (x^2-q).is_irreducible():
                    number_of_fields = number_of_fields +1
                    K.<b> = F.extension(x^2-q)
                    L.<c> = K.absolute_field()
                    hFnarr = F.narrow_class_group().order()
                    hL = L.class_number()
                    hLnarr = L.narrow_class_group().order()
                    if hLnarr == 1:
                        field_narrow_one = field_narrow_one + 1
                        rows.append([p, q, hL, hLnarr])
                        #print "(p = " + str(p) + ", q = "
+str(q) + ", hLwide = " +str(hL) + ", hLnarrow = " +
str(hLnarr)+")"
    proportion =
RDF(field_narrow_one/number_of_fields)
    print "The number of biquadratic fields L = Q(sqrt(p),
sqrt(q)) with 2 <= p < q <= " + str(n) + " is " +
str(number_of_fields) +", the number of L with hLnarrow = 1 is "
+ str(field_narrow_one)
    return table(rows, header_row = ["p", "q", "hLwide",
"hLnarrow"], frame = True, align = 'center')
```

```
biquadratic_fields_narrow_class_number_one(30)
```

```
The number of biquadratic fields L = Q(sqrt(p), sqrt(q)) with 2
<= p < q <= 30 is 45, the number of L with hLnarrow = 1 is
6
```

p	q	hLwide	hLnarrow
2	5	1	1
2	13	1	1
2	29	1	1
5	13	1	1
5	17	1	1
17	29	1	1