Northern Michigan University (Marquette Co, MI)

Time: 15 minutes

Consider the following series:

$f(0) = 1$
$f(1) = 2$
$f(2) = 3$
$f(3) = 8$
$f(4) = 16$
$f(5) = 35$
$f(6) = 75$, etc.

Similar to the Fibonacci series, each entry is found by adding the number before it, the number three before it, and double the number two before it. For example $f(7) = 75 + 16 + 2*35 = 161$.

Using recursion and hashmaps, write an efficient method `int f (int n)` to compute this function. (Do not use loops.) Don't worry about BigInteger; just use ints.

```java
private HashMap <String,String> HT = new HashMap ();

public int qu8 (int n) {

 int ans = 0;

 String s = HT.get(""+n);
 if (s==null) {
  if (n <=2) ans = n+1;
  else ans = qu8(n-1)+2*qu8(n-2)+qu8(n-3);
  HT.put (""+n,""+ans);
 } else ans = Integer.parseInt (s);
 return ans;
}
```