Northern Michigan University (Marquette Co, MI)

CS422-01-25W: Algorithms (Andrew A. Poe) Name: _____
Quiz 7 Friday 28 March 2025 9:00 A.M. EDT

Time: 15 minutes

The combination function ${}_nC_r$ is the numbers of ways to select r objects from a total set of n items. For example, if I have four differently colored balls, and I grab two of them at random, there are six ways to do this, so ${}_4C_2 = 6$.

We can compute the value of ${}_nC_r$ from Pascal's Triangle:

```
1
1       1
1       2       1
1       3       3       1
1       4       6       4       1
```

Notice that every number is the sum of the number directly above it and the one diagonally above it and to the left. Remembering that rows and columns begin their numbering with zero, we let n be the row number and r be the column number. The element in the fourth row and second column is 6, so we read from the table that ${}_4C_2 = 6$.

Generally speaking, ${}_nC_r = {}_{n-1}C_{r-1} + {}_{n-1}C_r$.

Write a recursive method:

```
int comb (int n, int r) { ... }
```

that uses recursion (no loops) to compute ${}_nC_r$. It should use a hash table so that the method will run efficiently. Be sure to check for all valid base cases. Your method does not have to work for out-of-bound values of n and r.

Using a class variable:

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

int comb (int n, int r) {

 String c = HT.get (n+","+r);
 if (c == null) {
  if (r==0 || n==r) c = "1";
  else c = ""+comb(n-1,r-1)+comb(n-1,r);
  HT.put (n+","+r,c);
 }
 return c;
```