

AP COMPUTER SCIENCE B – COMBINATIONS

A combination is a permutation where the order of the elements does not matter. The number of combinations of n distinct objects taken k at a time is

$${}_nC_k = \frac{n!}{(n-k)!k!} \cdot \quad (1)$$

Recall that the ${}_3P_3 = 6$ permutations of 3 objects taken 3 at a time are

$$012 \quad 021 \quad 102 \quad 120 \quad 201 \quad 210 \quad (2)$$

The ${}_3C_3 = 1$ combination of 3 things taken 3 at a time is

$$012 \quad (3)$$

Note that the combination (3) is contained in the permutation list (2).

The ${}_3P_2 = 6$ permutations of 3 objects taken 2 at a time are

$$01 \quad 02 \quad 10 \quad 12 \quad 20 \quad 21 \quad (4)$$

The ${}_3C_2 = 3$ combinations of 3 things taken 2 at a time are

$$01 \quad 02 \quad 12 \quad (5)$$

As before, the combinations (5) are contained in the permutation list (4).

Finally, the ${}_3P_1 = 3$ permutations of 3 objects taken 1 at a time are

$$0 \quad 1 \quad 2$$

which also corresponds to the ${}_3C_1 = 3$ combinations of 3 things taken 1 at a time.

The goal of this assignment is to write a program `TestCombination.java` that generates combinations using the idea described above. The class `TestCombination` will be used as-is. You will make the class `Combination` by starting with `CombinationStub.java`.

AP COMPUTER SCIENCE B – COMBINATIONS

In particular (note that the instructions are in the comments in file `CombinationStub.java`):

- 1) Add the required code to method `private static int numCombs (int ni, int ki)`.
- 2) Add the required code to method `private String [] nCk (int k)`.
- 3) Add the required code to method `public String toString ()` to generate a `String`, *e.g.*, for $n = 3$, in the format

```
3C1 = 3
0
1
2
3C2 = 3
01
02
12
3C3 = 1
012
```

Do not put a line-feed character after the last line.