**Practice Exercise #36: North-East Paths**

<http://www.comp.nus.edu.sg/~cs1020/4_misc/practice.html>

**Objective:**

Programming with recursion

**Task statement:**

In a special town where pedestrians are allowed to walk only northwards or eastwards, each of the following examples shows the total number of unique north-east paths **ne(x, y)** to get from point *A* to point *B*, where *B* is **x** rows north and **y** columns east of *A*. Assume that **x** and **y** are non-negative integers. By convention, ne(0, 0) is defined to be 1.

#### A

#### B

**ne(0,2) = 1**

#### A

#### B

#### A

#### B

**ne(1,3) = 4**

**ne(2,2) = 6**

*A*

*B*

*A*

*B*

*A*

*B*

Write a recursive method to compute the number of north-east paths, as well as to display all the north-east paths. Sample runs are shown below. You may observe that you are to explore northwards before exploring eastwards in the path if there is a choice.

Note that each letter ‘N’ or ‘E’ in the output is always followed by a space character. For example, in the first example below, the output path is "**E E** ", i.e. there is a space after the second ‘E’.

**Enter rows apart: 0**

**Enter columns apart: 2**

**E E**

**Number of paths = 1**

**Enter rows apart: 2**

**Enter columns apart: 2**

**N N E E**

**N E N E**

**N E E N**

**E N N E**

**E N E N**

**E E N N**

**Number of paths = 6**

**Enter rows apart: 1**

**Enter columns apart: 3**

**N E E E**

**E N E E**

**E E N E**

**E E E N**

**Number of paths = 4**