Tutorial session 8 - Recursion
CSC 113
King Saud University
College of Computer and Information Sciences

- Do **not** use loops in any method except `main`.
- Do **not** use global variables
- Do **not** give the class `recursor` any attributes
- Do **not** use **static** variables in any method.

1 Integer manipulation

Implement the following `static` recursive functions.

1.1 Exercise 1

Write the `static`, recursive function which takes an integer x and returns the fibonacci(x). The fibonacci function is defined as follows:

$$f(x) = \begin{cases} 0 & x = 0 \\ 1 & x = 1 \\ f(x-1) + f(x-2) & \text{otherwise} \end{cases}$$

1.2 Exercise 2

Write the `static`, recursive function `gcd` which receives two integers and returns the greatest common divisor of the two integers.
Write a main function to test it by reading two integers from the user and printing their GCD.

```
Enter number 1: 24
Enter number 2: 18
GCD is 6
```

$$gcd(x, y) = \begin{cases} y & y \text{ divides } x = 0 \\ gcd(y, x\%y) & \text{otherwise} \end{cases}$$

1.3 Exercise 3

Implement the `static` function `product` which receives two integers and recursively calculates their product.

```
Enter an integer: 5
Enter an integer: 4
4 * 5 = 20
```

- Do **not** use the * operator.
- Can you modify your function to make it handle negative input?

1.4 Exercise 4

Implement the `static` function `power` which receives two integers x, y and recursively calculates x^y.

```
Enter an integer: 5
Enter an integer: 4
power(4,5) = 1024
```

- Do **not** use `java.math.pow`
- Can you modify your function to make it handle negative input?

1.5 Exercise 5

Write a `main` function to test all the implemented recursive methods.