Arithmetic with binary numbers

- **Adding** binary numbers:
  - Addition rules:

```
  0     1     0     1
+ 0   + 0   + 1   + 1
---   ---   ---   ---
  0     1     1    10
                   ^
                   |
                   +--- carry
```

  - Small unsigned example:

```
In decimal     In binary

                     ***      * = indicate that previous
     5           00000101         bit addition produced a carry
   + 7         + 00000111
  ----         ----------
    12           00001100
```

  - Large unsigned example:

```
In decimal     In binary

                  **   *       * = indicate that previous bit addition
   145           10010001          produced a carry
 +   61        + 00111101
-------        ----------
   206           11001110
```

- **Subtracting** binary numbers:
  - Subtraction rules:

```
  0     1     0     1
- 0   - 0   - 1   - 1
---   ---   ---   ---
  0     1    *1     0
             ^
             |
             +--- BORROW !
```

  - When you borrow, you receive 2 more units (because digit values increases by 2 each time you move to the left one position).
  - Smale example:

```
In decimal     In binary

                   *         * = indicate that previous
     9         00001001          bit subtraction produced
   - 5       - 00000101          a borrow
  ----       ----------
     4         00000100
```

- Large example:

```
In decimal     In binary

                ** *             * = indicate that previous bit subtraction
     149        10010101             produced a borrow
   -  41      - 00101001
   -----      ----------
     108        01101100
```

- **Multiplying** binary numbers:
  - Multiplication rules:

```
  0     1     0     1
x 0   x 0   x 1   x 1
---   ---   ---   ---
  0     0     0     1
```

  - Small **unsigned** example:

```
      5            00000101
    x 3          x 00000011
   ----         -----------
     15            00000101
                  00000101*
                -----------
                   00001111  = 15
```

  - Large **unsigned** example:

```
          10010101    (= 149 dec)
        x 00101001    (= 41  dec)
        ----------
          10010101
         00000000*
        00000000**
       10010101***
      00000000****
     10010101*****
    00000000******
   00000000*******
  ----------------
   001011111011101     (= 6109 = 149*41)
```

  - **NOTE:** when you multiply 2 binary numbers of ***n*** bits, the result may be as large as ***2n*** bits !

- **Dividing** binary numbers:
  - Division rule:
    - Find number that is larger than divider
    - It will always devide **once** (much easier than division in decimal number system)
  - Small example:

```
In decimal:                    In binary:

     03 (quotient)                  00000011 (quotient)
   -------                         ------------------
3 /  10                        11 / 00001010
      9                                  11
    ----                                ---
      1 (remainder)                      100
                                          11
                                         ---
                                           1 (remainder)
```

  - Large example: 237 divide by 7

```
         100001 (quotient = 33 dec)
     -------------------
111 /  11101101
       111
      ----
         00
          0
        ---
          01
          00
         ---
           11
           00
          ---
           110
           000
          ----
           1101
            111
          -----
            110   (remainder = 6 dec)
```