Golang & Python tasks

Basic

- Define a function, that takes string as argument and prints `"Hello, %arg%!"`
- Define a function `sum()` and a function `multiply()` that sums and multiplies (respectively) all the numbers in an array (list) of numbers. For example, `sum([1, 2, 3, 4])` should return 10, and `multiply([1, 2, 3, 4])` should return 24.
- Define a function `reverse()` that computes the reversal of a string. For example, `reverse("I am testing")` should return the string “gnitset ma I”.
- Define a function `isPalindrome()` that recognizes palindromes (i.e. words that look the same written backwards). For example, `isPalindrome("radar")` should return `True`.
- Define a procedure `histogram()` that takes an array(list) of integers and prints a histogram to the screen. For example, `histogram([4, 9, 7])` should print the following:

```
****
*********
******
```

(usage some kind of sleep for better visualization)

- Define a function `caesarCipher` that takes string and `key(number)`, whuch returns encrypted string using the Ceasar Cipher
- define a function `diagonalReverse()` that takes matrix (two dimensional array (list)) and returns diagonal-reversed one:

```
1 2 3         1 4 7
4 5 6 returns 2 5 8
7 8 9         3 6 9
```

- Write a function `game()` number-guessing game, that takes 2 int parameters defining the range. Use some kind of `random` function to generate random int from the range. While user input isn’t equal that number, print “Try again!”. If user guess the number, congratulate him and exit.

- Define a function, which takes a string with N opening brackets `("[") and N closing brackets ("]")`, in some arbitrary order.
  Determine whether the generated string is balanced; that is, whether it consists entirely of pairs of opening/closing brackets (in that order), none of which mis-nest.
  Examples:

```
  []        OK   ][        NOT OK
  [][]      OK   ][][      NOT OK
  [[][]]    OK   []][[]    NOT OK
```

- Write a function `charFreq()` that takes a string and builds a frequency listing of the characters contained in it. Represent the frequency listing as Map(Dictionary). Try it with something like `charFreq("abbabcbdbabdbdbababababcbcbab")`.

- Write a function `decToBin()` that taces decimal integer and outputs its binary representation.

- Optional: Write a battleship game. It should generate field with random ships and accept user input (X, Y coordinates) and report hit or miss.

- hard task: Visualize the battleship game in console (fmt.Print, etc.).

Advanced (OOP)

- Create structure for department:
  - There are 3 types of employee: developer, designer and manager
  - Each employee has: first name, second name, salary, experiance (years) and manager
  - Each designer has effectivness coefficient(0-1)
  - Each manager has team of developers and designers.
  - Department should have list of managers(which have their own teams)
  - Department should be able to give salary (for each employee message: `"@firstName@ @secondName@: got salary: @salaryValue@"`)
  - Each employee gets the salary, defined in field Salary. If experiance of employee is > 2 years, he gets bonus + 200$, if experiance is > 5 years, he gets salary*1.2 + bonus 500
  - Each designer gets the salary = salary*effCoeff
  - Each manager gets salary +
    1. 200$ if his team has >5 members
    2. 300$ if his team has >10 members
    3. salary*1.1 if more than half of team members are developers.
  - Redefine string representation for employee as follows: `"@firstName@ @secondName@, manager:@manager secondName@, experiance:@experiance@"`