

Intro to programming I

Week 6 - Program flow and functions

- A program is a recipe, a sequence of commands that do something. Like following a recipe, Python executes them one at a time, in order, from top to bottom.

- The programming strategy that we will use is called **top-down**: we solve a problem by breaking it in parts; when we are done, we consider each part as a new problem and solve it, again, using the top-down approach; this process continues until we can solve the problem using Karel's built-in commands. For example, we want Karel to pick up a token, put it in a basket, and then go home, so we break the problem in parts:

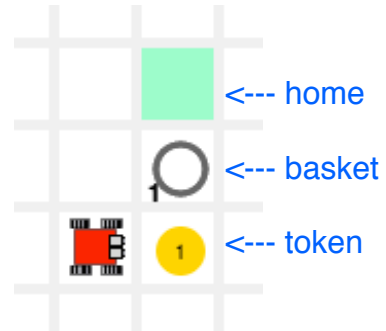
```
#main program
get_token()
put_in_basket()
go_home()
```

- Our main program solves our problem but Karel does not know what our commands are; we need to teach him what they mean using **functions**, which are mini-programs that are ran only when we 'call' them from the main program or from another function. In Python, functions look like this:

```
def name():          # header of the function
    command_1       # body of the function
    command_2
    ...
    command_n
```

- The main program can use functions that appear before the main program; the order in which we write the functions is not important, though.

Our Problem:



Our solution:

```
sound(True)
think(900)
```

```
def get_token():
    say("taking the token")
    move()
    take()
```

```
def put_in_basket():
    say("putting the token")
    turn_left()
    move()
    put()
```

```
def go_home():
    say("going home")
    move()
```

```
# main program
get_token()
put_in_basket()
go_home()
say("done")
```

Our problem solved:

