



Study

Read Session 20, chapter 20.0, 20.1, 20.2:

<http://www.ict.ru.ac.za/Resources/cspw/thinkcs3/thinkcs3.pdf>

And then do exercises: 20.8.1



Serious exercises

Exercise 1

Given the following dictionary:

```
inventory = {  
    'gold' : 500,  
    'pouch' : ['flint', 'twine', 'gemstone'],  
    'backpack' : ['xylophone', 'dagger', 'bedroll', 'bread loaf']  
}
```

Try to do the followings:

- Add a key to inventory called 'pocket'.
- Set the value of 'pocket' to be a **list** consisting of the strings 'seashell', 'strange berry', and 'lint'.
- Then `.remove('dagger')` from the list of items stored under the '**backpack**' key.
- Add 50 to the number stored under the '**gold**' key.

Exercise 2

Write a program to count number occurrences in a list, with AND without using `count()` function

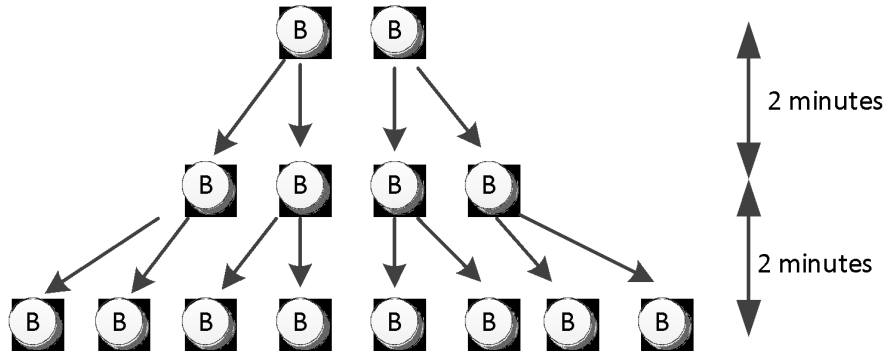
Example:

```
numbers = [1, 6, 8, 1, 2, 1, 5, 6]
```

```
Enter a number? 1  
1 appears 3 times in my list
```

Exercise 3

Bacteria B **replicates** itself each 2 minutes, write a program that asks users to enter two numbers: the **initial B bacteria number** and a period of **time (in minutes)**. Calculate and print out the **total number of B bacteria** after this period.



Expected screen output:

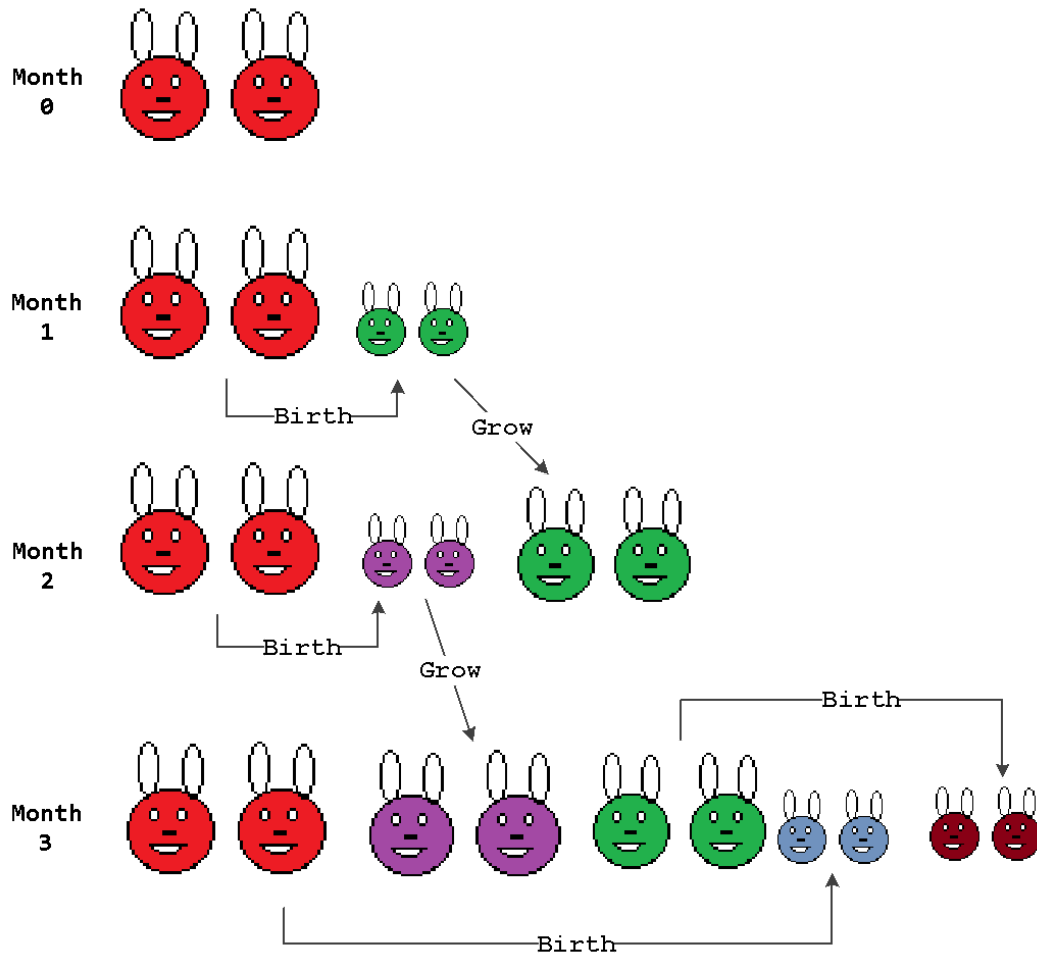
```
How many B bacterias are there? 3
How much time in minutes will we wait? 2
After 2 minutes, we would have 6 bacterias
```

Or:

```
How many B bacterias are there? 2
How much time in minutes will we wait? 4
After 4 minutes, we would have 8 bacterias
```

Exercise 4

1. (Optional) In Happy Farm, there are initially a couple of rabbits (female and male). This couple of the rabbits reproduces a new couple of rabbits each month. Each newborn rabbit couple becomes mature in one month and then gives a life to a new rabbit couple each month after. Write a program that calculates the number of pair of rabbit after 4 months.



Expected screen output:

(Note that no user's information needed)

```
Month 0: 1 pair(s) of rabbit
Month 1: 2 pair(s) of rabbit
Month 2: 3 pair(s) of rabbit
Month 3: 5 pair(s) of rabbit
Month 4: 8 pair(s) of rabbit
```

If you need hint, scroll to the last page

Exercise 5 (optional):

Create a new dictionary called `prices` using `{}` format like the example above.

Put these values in your `prices` dictionary:

- `"banana": 4,`
- `"apple": 2,`
- `"orange": 1.5,`
- `"pear": 3`

Create another dictionary called `stock` using `{}`:

Put these values in your `stock` dictionary

- `"banana": 6,`
 - `"apple": 0,`
 - `"orange": 32,`
 - `"pear": 15`
-
- Loop through each key in `prices`. For each key, print out the key along with its price and stock information. Print the answer in the following format:
 - `apple`
 - `price: 2`
 - `stock: 0`
 - `pear`
 - `price: 3`
 - `stock: 15`
 - Let's determine how much money you would make if you sold all of your food.
 - Create a variable called `total` and set it to zero.

- o Loop through the prices dictionaries. For each key in `prices`, multiply the number in `prices` by the number in `stock`. Print that value into the console and then add it to `total`.
- o Finally, outside your loop, print `total`.

Exercise 6 (Optional):

Finish `Sokoban` game with the following requirements:

1. Player can't push box out of map
2. Player can't push multiple boxes
3. Add Obstacle or Wall into map, player can't move across Obstacle and can't push box over Obstacle

```
- - - - -
- P - - -
B B B B -
D O - - -
- D - - -
D - - D O -
```

What is your next move? W/A/S/D Or (Undo - Z) s
Oops! Obstacle :'(

```
- - - - -
- P - - -
B B B B -
D O - - -
- D - - -
D - - D O -
```

4. Player can undo their last move (**one each time**)

```

- - - - -
- - - - -
P B B B - -
B O - - - -
- D - - - -
D - - D O -

```

What is your next move? W/A/S/D Or (Undo - Z) s

```

- - - - -
- - - - -
- B B B - -
P O - - - -
B D - - - -
D - - D O -

```

What is your next move? W/A/S/D Or (Undo - Z) z
Undo Action!!

```

- - - - -
- - - - -
P B B B - -
B O - - - -
- D - - - -
D - - D O -

```


Google "Fibonacci sequence"