

# Principles of Software Programming

## Lecture 9 (Wahlfach): Exceptions and Dynamic data structures



Anton Yeshchenko  
SS 2018

APRIL 2018

Some slides and/or Ideas were borrowed from:  
MIT Introduction to Computer Science and Programming in Python  
and Svitlana Vakulenko WS 2017 lecture slides



- Exceptions (when things go wrong)
- Dynamic data structures (is there only lists out there?)

# 1. Exceptions. Syntax errors (Program won't run)

```
while True print('Hello world')
```

```
File "<ipython-input-1-614901b0e5ee>", line 1
```

```
while True print('Hello world')
```

^

```
SyntaxError: invalid syntax
```

# 1. Exceptions

GET THE CAR TO RAV

00:19.99

1,434 YARDS





# 1. Exceptions



Your PC ran into a problem and needs to restart. We're just collecting some error info, and then we'll restart for you. (0% complete)

If you'd like to know more, you can search online later for this error: MEMORY\_MANAGEMENT

# **Uncaught** exception! Just crashes everything

# 1. Exceptions

GET THE CAR TO RAV

00:19.99

1,434 YARDS



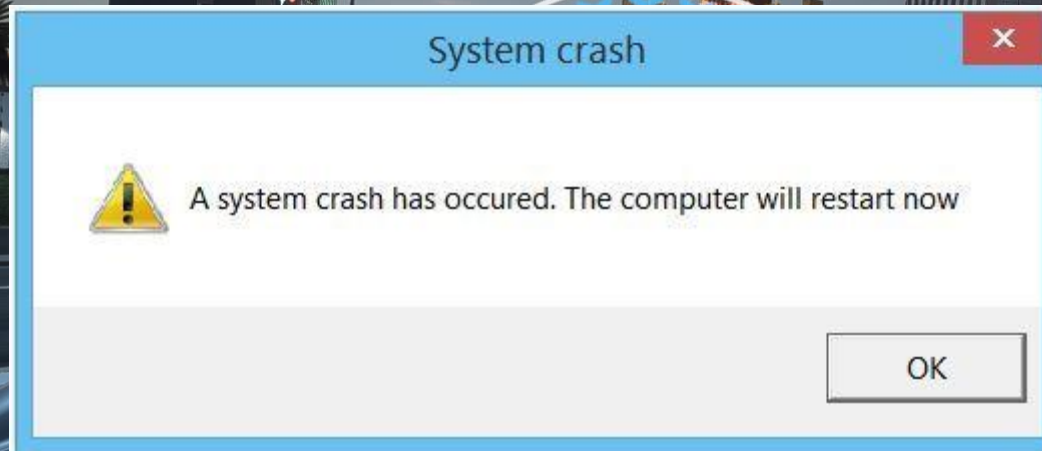


# 1. Exceptions

GET THE CAR TO RAV

00:19.99

1,434 YARDS





# 1. Exceptions



# Handling exceptional situations

```
>>> 10 * (1/0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
>>> 4 + spam*3
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'spam' is not defined
>>> '2' + 2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Can't convert 'int' object to str implicitly
```

# Handling exceptional situations

```
>>> 10 * (1/0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
>>> 4 + spam*3
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'spam' is not defined
>>> '2' + 2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Can't convert 'int' object to str implicitly
```

# Handling exceptional situations

```
>>> 10 * (1/0)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ZeroDivisionError: division by zero
>>> 4 + spam*3
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'spam' is not defined
>>> '2' + 2
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: Can't convert 'int' object to str implicitly
```



# Implicit conversion miracle!

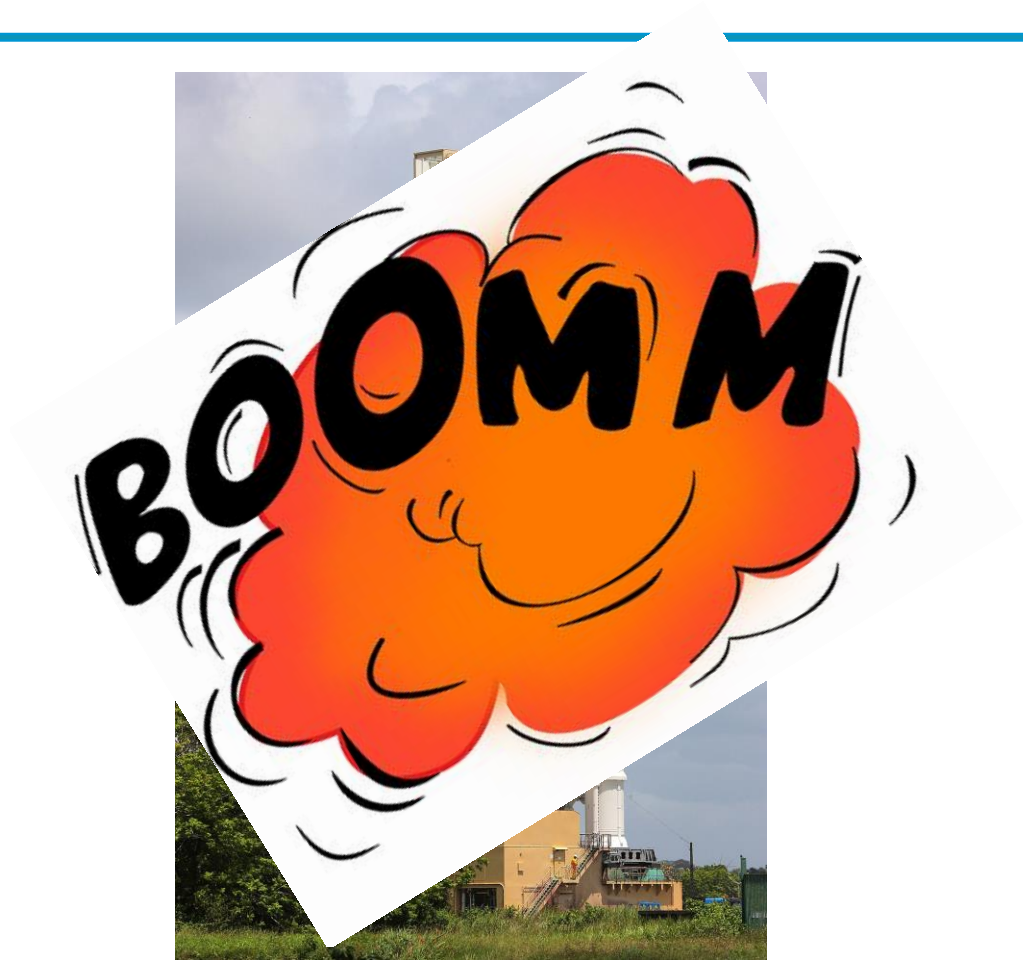
## Ariane 5

- 4 June 1996
- European Space Agency 10 years and \$7 billion
- takeoff in French Guiana



[HTTP://WWW.MILITARY.COM/VIDEO/SPACE-TECHNOLOGY/LAUNCH-VEHICLES/ARIANE-5-ROCKET-LAUNCH-FAILURE/2096157730001](http://www.military.com/video/space-technology/launch-vehicles/ariane-5-rocket-launch-failure/2096157730001)  
[HTTPS://WWW.LINKEDIN.COM/PULSE/ARIANE-5-ROCKET-LAUNCH-FAILURE-DO-ONE-LINE-COMPUTER-CODE-DAHLBERG/](https://www.linkedin.com/pulse/ariane-5-rocket-launch-failure-do-one-line-computer-code-dahlberg/)  
[HTTPS://UPLOAD.WIKIMEDIA.ORG/WIKIPEDIA/COMMONS/THUMB/3/3C/ARIANE\\_5ES\\_WITH\\_ATV\\_4\\_ON\\_ITS\\_WAY\\_TO\\_EL](https://upload.wikimedia.org/wikipedia/commons/thumb/3/3C/Ariane_5ES_with_ATV_4_on_its_way_to_ELA-3.JPG/1200px-Ariane_5ES_with_ATV_4_on_its_way_to_ELA-3.JPG)  
[A-3.JPG/1200PX-ARIANE\\_5ES\\_WITH\\_ATV\\_4\\_ON\\_ITS\\_WAY\\_TO\\_ELA-3.JPG](https://upload.wikimedia.org/wikipedia/commons/thumb/3/3C/Ariane_5ES_with_ATV_4_on_its_way_to_ELA-3.JPG/1200px-Ariane_5ES_with_ATV_4_on_its_way_to_ELA-3.JPG)

# Implicit conversion miracle!



[HTTP://WWW.MILITARY.COM/VIDEO/SPACE-TECHNOLOGY/LAUNCH-VEHICLES/ARIANE-5-ROCKET-LAUNCH-FAILURE/2096157730001](http://www.military.com/video/space-technology/launch-vehicles/ariane-5-rocket-launch-failure/2096157730001)

[HTTPS://WWW.LINKEDIN.COM/PULSE/ARIANE-5-ROCKET-LAUNCH-FAILURE-DO-ONE-LINE-COMPUTER-CODE-DAHLBERG/](https://www.linkedin.com/pulse/ariane-5-rocket-launch-failure-do-one-line-computer-code-dahlberg/)

# Easy solution to save **poor rocket!**

```
while True:
    try:
        x = int(input("Please enter a number: "))
        break
    except ValueError:
        print("Oops! That was no valid number. Try again...")
```

Please enter a number: 23sdf

Oops! That was no valid number. Try again...

Please enter a number:

```
while True:
    try:
        x = int(input("Please enter a number: "))
        break
    except ValueError:
        print("Oops! That was not a valid number.")
```



# Exception is a class (with it's own hierarchy)

- Multiple **except** from the most specific to the most general
- Exception inheritance
- **Except** without the exception name (wild card)
- **Else** after the **except**
- **Exception with value**
  - **except** Exception **as** inst:

# Raising exceptions

```
>>> raise NameError('HiThere')
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: HiThere
```

```
raise ValueError # shorthand for 'raise ValueError()'
```

```
>>> try:
...     raise NameError('HiThere')
... except NameError:
...     print('An exception flew by!')
...     raise
...
An exception flew by!
Traceback (most recent call last):
  File "<stdin>", line 2, in <module>
NameError: HiThere
```

# Custom exceptions (usually for messages, or even handling)

- Exceptions should be derived from the class **Exception**

```
class Error(Exception):  
    """Base class for exceptions in this module."""  
    pass  
  
class InputError(Error):  
    """Exception raised for errors in the input.  
  
    Attributes:  
        expression -- input expression in which the error occurred  
        message -- explanation of the error  
    """  
  
    def __init__(self, expression, message):  
        self.expression = expression  
        self.message = message
```

# Happy **uninterrupted** playing!





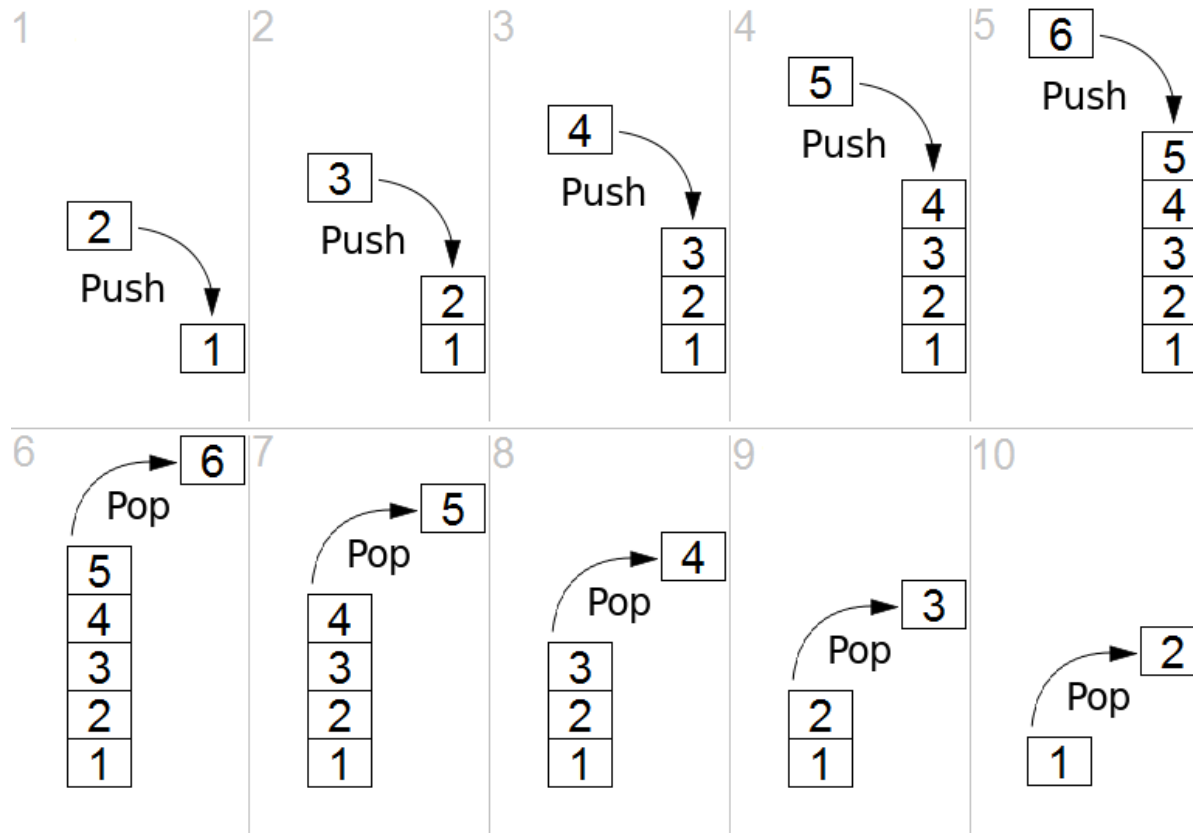
# Exercise 1

1. Create a function that will take two numbers
2. Multiply them
3. if the result is bigger than 1000 raise error
4. Is not return the result
5. Catch the error



# STACK

# Stack



# QUEUE



# FIFO

from collections import deque



```
from collections import deque
queue = deque(["Eric", "John", "Michael"])
queue.append("Terry")           # Terry arrives
queue.append("Graham")         # Graham arrives
queue.popleft()                 # The first to arrive now Leaves
queue.popleft()                 # The second to arrive now Leaves
```

# Dictionary (Map)

## Hater

People\_I\_hate["Andrew"] = "because he broke my arm when I was 10"

People\_I\_hate["Celina"] = ["because she owns me money", "She is too tall, and I am too small"]

Etc..

# Binary TREE





# Tree with data

# How to make tree

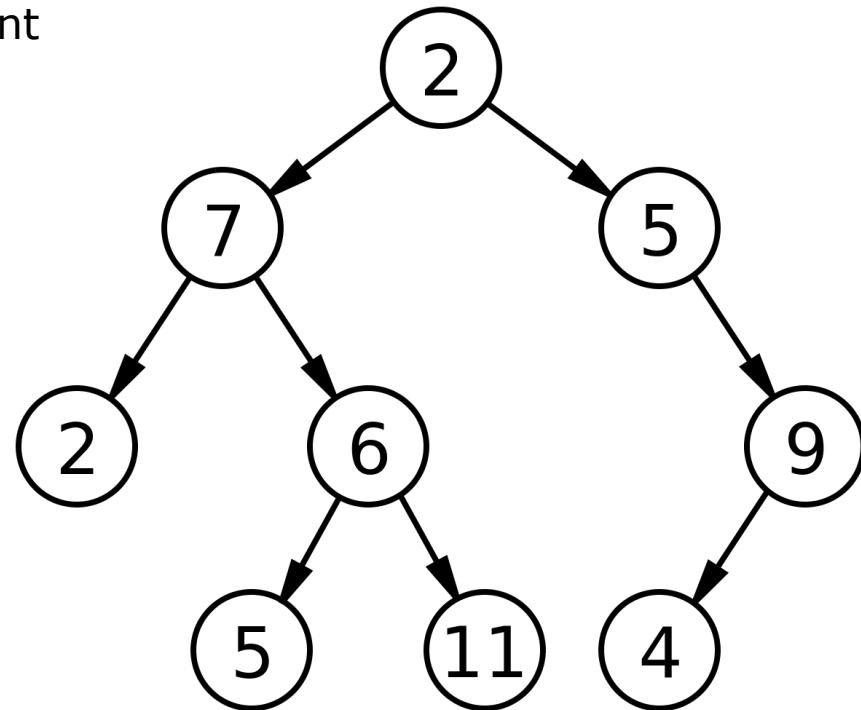
```
class Tree(object):  
    def __init__(self):  
        self.left = None  
        self.right = None  
        self.data = None
```

You can use it like this:

```
root = Tree()  
root.data = "root"  
root.left = Tree()  
root.left.data = "left"  
root.right = Tree()  
root.right.data = "right"
```

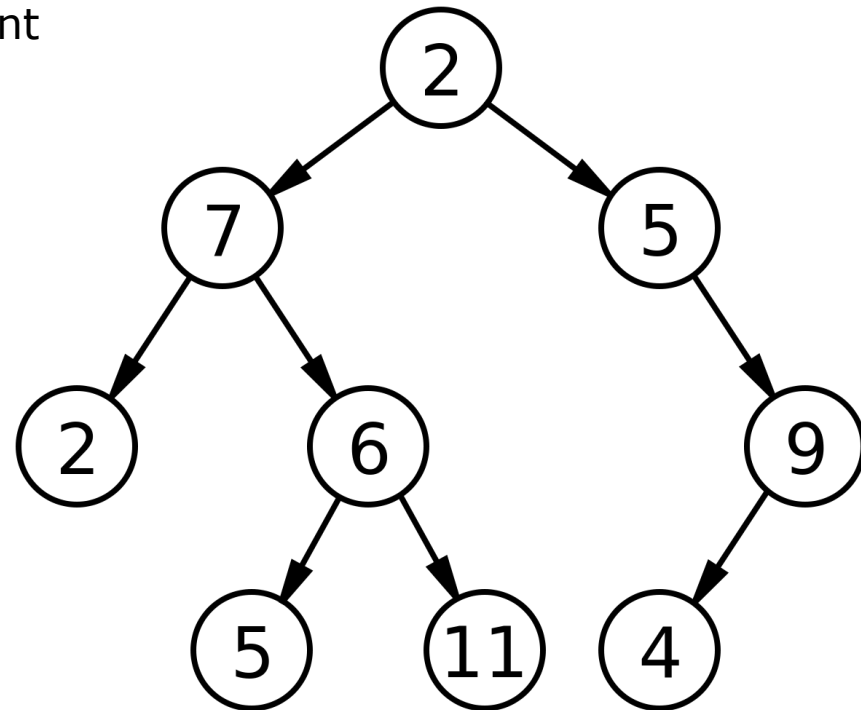
# Binary search tree

- They allow fast lookup, addition and removal of items
- Lookup -> is element there?
- Addition -> input the new element
- Removal



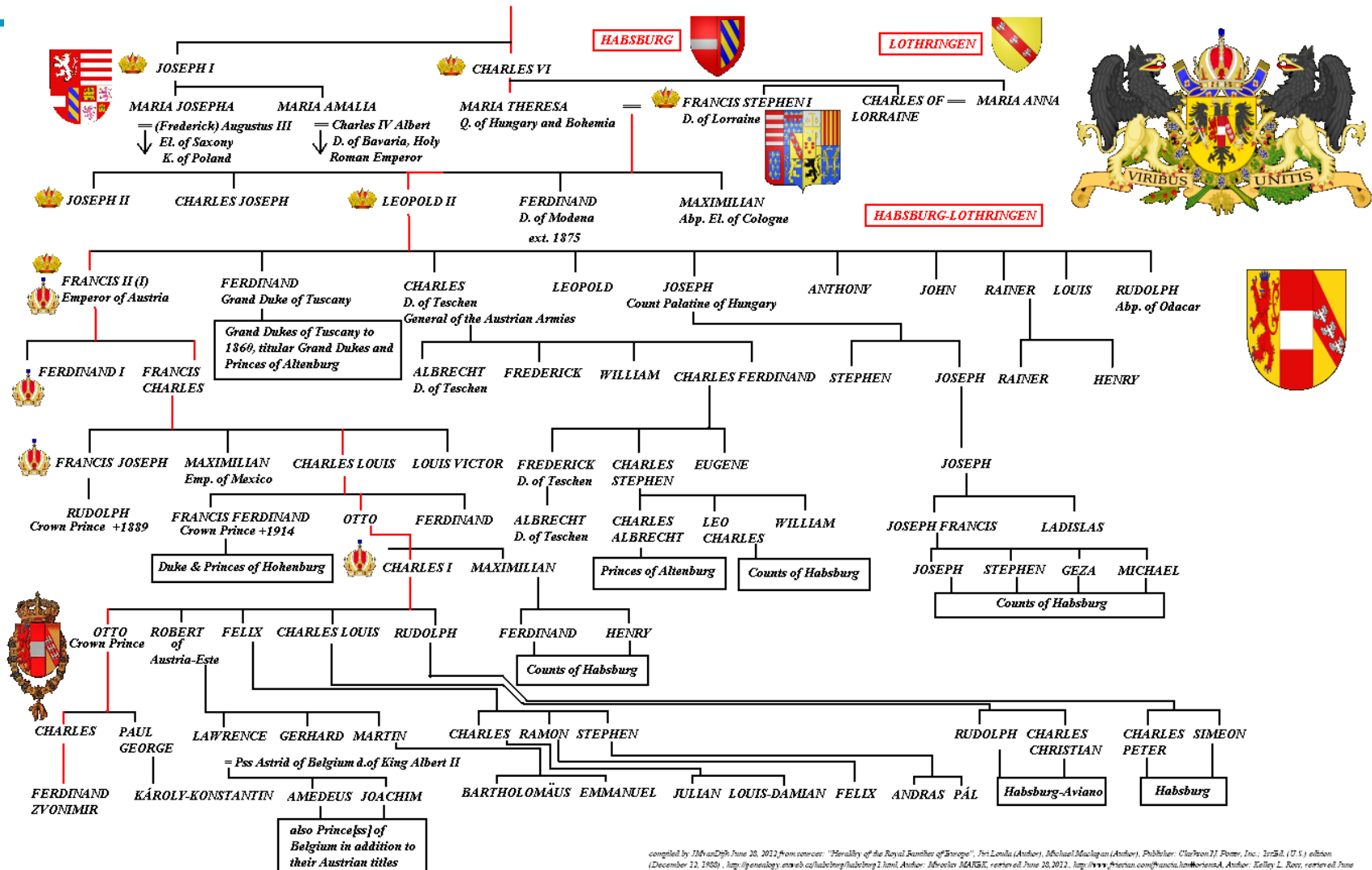
# Binary search tree

- They allow fast lookup, addition and removal of items
- Lookup -> is element there?
- Addition -> input the new element
- Removal





# Habsburg Family Tree



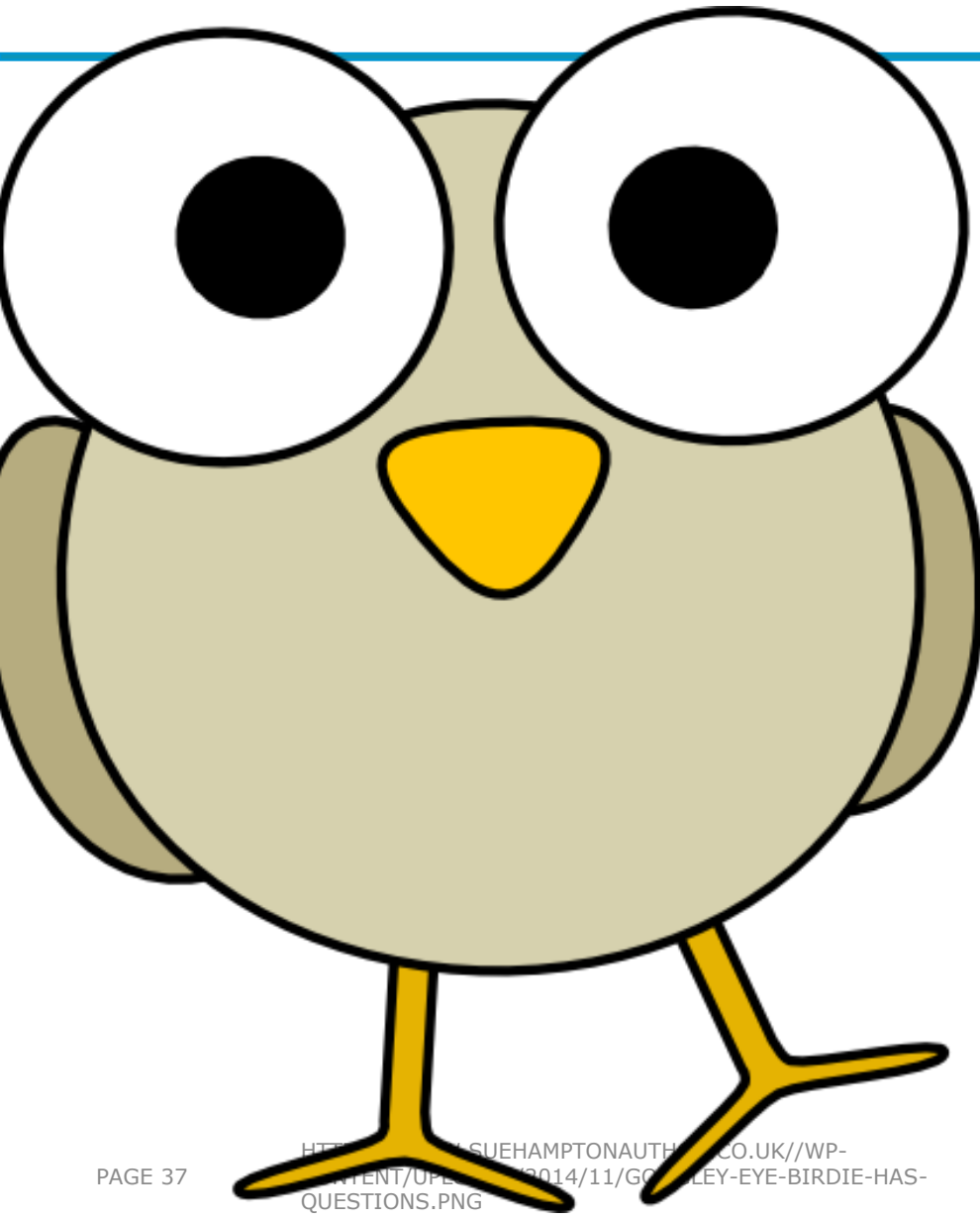
compiled by JMW and JPH June 20, 2012 from sources: "Hereditary of the Royal Family of Europe", J. L. L. (Author), Michael Maclean (Author), Publisher: Charles J. Potter, Inc., 1st Ed. (17 x 11) edition (December 12, 1958), <http://www.royal.gov.uk/habsburg/habsburg1.html>, Author: Michael MARKE, revised June 20, 2012, <http://www.friar.com/franciscan/monarchsA.html>, Author: Kelley L. Ross, revised June 20, 2012, <http://www3.tel.nl/ac/stijgen/encyclo/CEDOOM.html>, Author: Brian Tongue's Royal and Noble Genealogy, revised June 20, 2012, <http://royal.gov.uk/habsburg/habsburg1.html>, Bruce R. Gordon's Royal Chronology, revised June 20, 2012.

## Exercise 2

1. Create **stack** shopping basket, and **stack** backpack
2. Create **queue** conveyor belt
3. Put some items in the shopping basket like: "apples", "milk", etc.
4. Put it all in the conveyor ( **directly from the shopping basket** )
5. Then from conveyor put it into backpack



# TIME FOR! **KAHOOT** quiz!



- Exceptions (when things go wrong)
- Dynamic data structures
  - Stack
  - Queue
  - Dictionary
  - Trees