CS101 - Data Abstraction

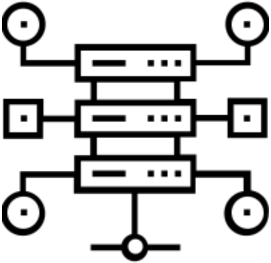
DS Basics - Module1

Aravind Mohan

Allegheny College

April 6, 2021

Data Structures



- **Definition** - A data structure is a technique that is primarily used to access, process, store, and organize data.

Data Structures

- Core operations supported by a data structure:
  - Add [store]
  - Retrieve [access/read]
  - Remove [organize]
- Other supporting operations are possible based on the data structure.
- Few examples:

  Arrays, Linked List, Stacks, Queues, Hash Maps, Trees, Graphs, etc.

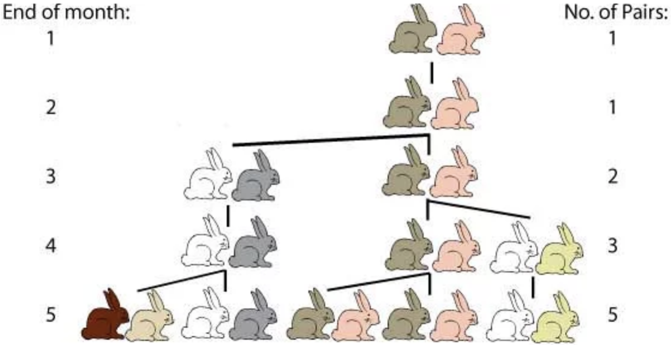
What is an Array?

- Consecutive blocks of data in memory. Elements in an array are contiguous (located next to each other).
- All elements of an array should be of same type.
- Any type of object can be stored in an array: integers, doubles, booleans, strings, . . .
- The size of the array can be found using **len(a)**

How is an Array different from a List?

- Arrays are **Homogenous** data storage. List are **Heterogenous** data storage.
- Arrays are compact and does not take up too much of space in memory. List take up additional space in memory.
- Arrays [**built-in** and **Numpy**] are more efficient in terms of time taken compared to a List.
- Mutable like List.

How do we process an Array?



- **Fibonacci Sequence:** {0,1,1,2,3,5,8, . . . }
- **Leonardo Problem:** At the end of year, how many pairs of rabbits exist?

Two-Dimensional Arrays

- Solution: Add a second dimension!

  `ls = [[1,2,3,4,5],[6,7,8,9]]`
- Format is [rows][columns]
- Each element still must be the same type
- Can still access each item individually
  - ls[6][1]

2D Example

Display 2d list (row-wise)

```
ls = [[1,2,3,4,5],[6,7,8,9],[10,11,12],[13,14],[15]]
for i in range(len(ls)):
  for j in range(len(ls[i])):
    print(ls[i][j],"\t",end='')
  print("")
```

Find the min, and max in each row and column!

2D Example

Display 2d list (column-wise)

```
ls = [[1,2,3,4,5],[6,7,8,9],[10,11,12],[13,14],[15]]
for i in range(len(ls)):
  for j in range(len(ls[i])):
    print(ls[j][i],"\t",end='')
  print("")
```

Check for duplicates in a row!

Dimensions

- ls[1][2][3]
- ls[1][2][3][4]
- ls[1][2][3][4][5]
- ls[1][2][3][4][5][6]. . .

. . . if you get this far, you may want to rethink your data storage . . .

Dynamic Arrays

- Dynamic Arrays are customized arrays using low level ctypes module.
- It is an implementation of a List.
- Ability to modify (increase or decrease) the size a list.
- Mutable.

PS darray.py in repo!

Next ...

- **Recursion**
- **Linked List**

Reading Assignment

GT Chapter 5 - 5.2, 5.3, 5.4

Questions?

Please ask if there are any Questions!