23-01-18 – COMP15 Notes

Reading

Shaffer 1.1-1.2, Next: Shaffer 3.1

Outline

1. Abstract Datatypes
2. ArrayList as client
3. Pre-lab #1

Abstract Data Types

- ADT : separates properties from the implementation
- Implementer: writes ADT implementation
- Client: call ADT functions
- Data Structures organize info into memory
- can solve any problem with an array
- Ex: delete something from an array

- Array: {5,8,13,21,34}; **delete int at position i**

  - Approach #1
    - copy over original
    - skip position i
  - Approach #2
    - set value at i to 0
    - shift everything else to the left
    - make size smaller

```
//array ray
//with size = 5
ray[i] = 0;
for (int j = i; j < size - 1; j++)
    {
        ray[j] = ray[j+1];
    }
size--;
```

ArrayList as Client

- absraction of an array
- ex: ArrayList of Pittsburgh Pirates

```
class Player
{
    string name; //input
    int number;
    double BA;
    string position;
    int lineup;
    int salary;
}
//Driver
//1. Add all players to an ArrayList (roster)
//2. Prompt user to look up a player
//3. Call find function, yes/no if in arraylist, return the
//found object
//Find function requirements
//name: find
//parameters: Player object (looking for)
//bool reference (modified in function)
//return: Player object (found)
//Drier...
bool found;
Player p1("Starling Mart");
Player p2 = ray.find(p1, found);
```