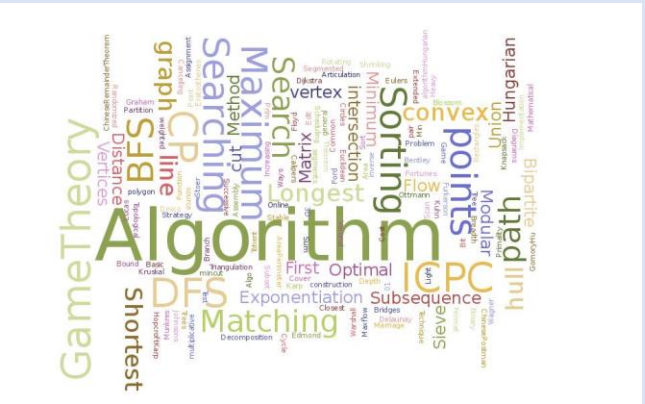
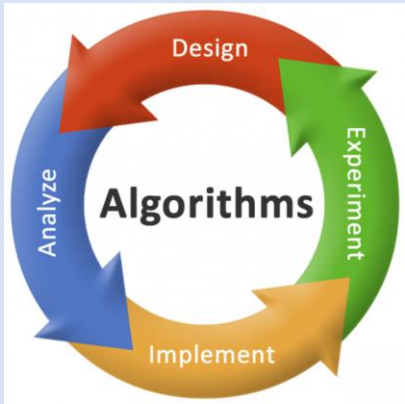
Java Generics

COP 3503
Fall 2021
Department of Computer Science
University of Central Florida
Dr. Steinberg





Prerequisites going into the lesson assumed

- Basic understanding of Java Language
- Basic understanding of Object-Oriented Concepts

What are Generics why it’s important?

- Generic programming means writing code that can be reused for objects of many different types.
- Generic classes are classes with one or more type variables.
- This allows Java programmers to create on class that handle a variety of types instead of creating separate class file for each type.
- Before generics has evolved, the concept of inheritance was used to create classes that would inherit certain behavior from the parent class.

Generic Class Example

```
public class Pair<T>
{
   private T first;
   private T second;

   public Pair() { first = null; second = null; }
   public Pair(T first, T second) { this.first = first;  this.second = second; }

   public T getFirst() { return first; }
   public T getSecond() { return second; }

   public void setFirst(T newValue) { first = newValue; }
   public void setSecond(T newValue) { second = newValue; }
}
```

Variable Type T

Lets Look at an Example

Generic Methods

- The example we saw showed a generic class.
- Besides generic classes we can also create generic methods.

```
class ArrayAlg
{
    public static <T> T getMiddle(T... a)
    {
        return a[a.length / 2];
    }
}
```

Calling a Generic Method

```
String middle = ArrayAlg.<String>getMiddle(words2);
```

OR

```
String middle = ArrayAlg.getMiddle(words2);
```

Bounding Type Variables

- While generics allows us to use arbitrary types, we may need to sometimes restrict on the type.
- Example. What if we wanted to find the smallest element?
- We need to make sure the class has the compareTo method from the comparable interface.

```
public static <T extends Comparable> T min(T[] a)
```

Lets look at an example!

Elementary Data Structures Quick Review

- In CS1, you learned 6 elementary data structures
  - Array
  - LinkedList
  - Stack
  - Queue
  - Hash Tables
  - Binary Trees
    - Binary Search Trees (BST)
- Each data structure has a unique properties.
- Remember!!! Each data structure has its own limits! There is no superior data structure!

ADT (Abstract Data Type)

- While languages such as Java have built in data structures, programmers can create their own version from scratch to improve qualities (like running time!).

Now Let's Build a ADT Generic LinkedList