COPY CONSTRUCTOR SEC 2.11 Ezust

ClassName (const ClassName& x)

ASSIGNMENT OPERATOR

ClassName& operator = (const ClassName& x)

---

chapter 2 / example 2_16 to 2_18 

FROM 6.6

chapter 6 / copy-assign

TEMPLATES

classes | functions

DECLARING .h

```
...
template < class any_type >
class my_class{
    ...
    void my_function (...);
    ...
}
```

T

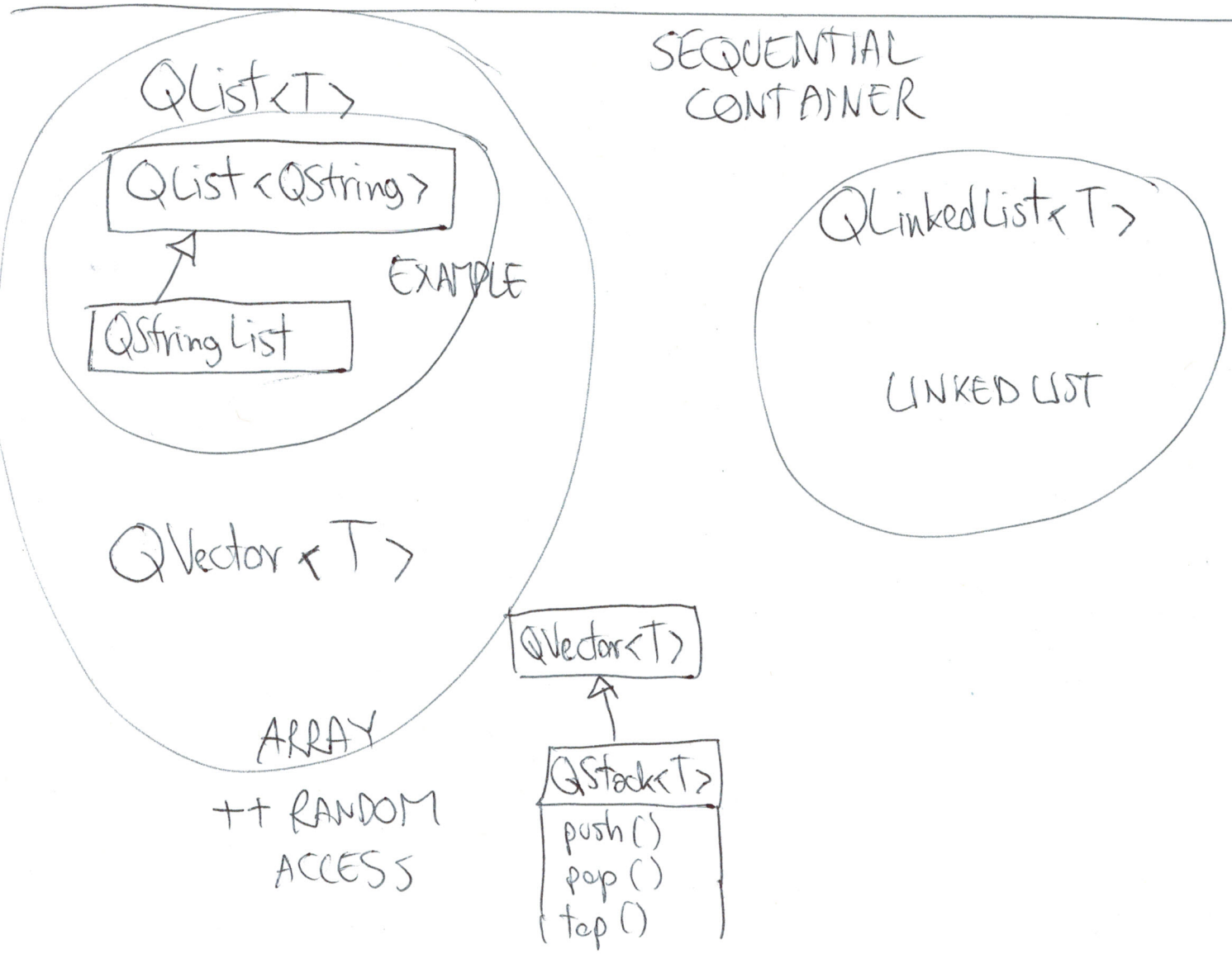
DEFINE .h / .cpp

```
...
template <class any_type>
void my_class<any_type>::my_function (...){
...
}
```

DECLARE AN INSTANCE

```
...
my_class<int> an_int_instance_of_my_class;
my_class<float> a_float_...;
```

SEC 6.8 CONTAINERS



	Lookup	Insert	Prepend	App
QLinkedList<T>	O(n)	O(1)	O(1)	O(1)
QList<T>	O(1)	O(n)	Amort. O(1)	Amort O(1)
QVector<T>	O(1)	O(n)	O(n)	Amort O(1)

ASSOCIATIVE CONTAINERS

QMap<key, T>

skip-list

SORTED ORDER

QHash<key, T>

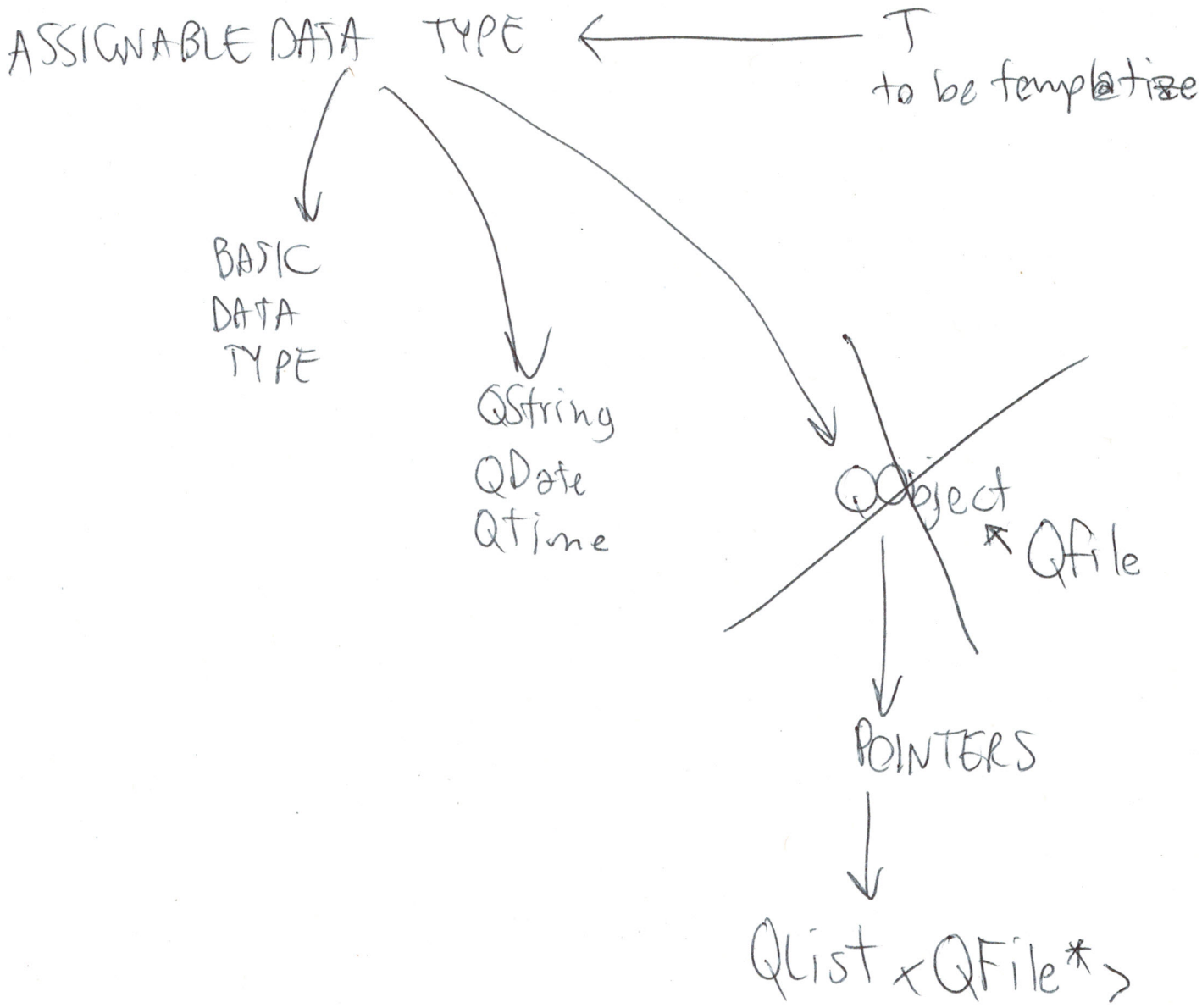
HASH TABLE

QCache<key, T>

QSet<T>

	LOOKUP		INSERTION	
	AVG	WCET	AVG	WCET
QMap<key, T>	O(log n)	O(log n)	O(log n)	O(log n)
QHash<key, T>	Amort O(1)	O(n)	Amort O(1)	O(n)

SEE http://doc.qt.io/qt-5/containers.html#algorithmic-complexity



CHECK PAGE 205 Exist

SEC 6.9

DIFFERENCE BETWEEN
MANAGED AND UNMANAGED container
(we already saw this in prev lectures)

SEC 6.10

- DESTRUCTION OF OBJECTS
- DON'T ACCESS OBJECTS THAT HAVE BEEN DESTROYED

- COPY CONSTRUCTOR
  ASSIGNMENT OPERATOR

p 210 book

p 212
chap6/example 6_30to6_40

FIG 6.8 Ezust p212
(Copyright Ezust et al.)

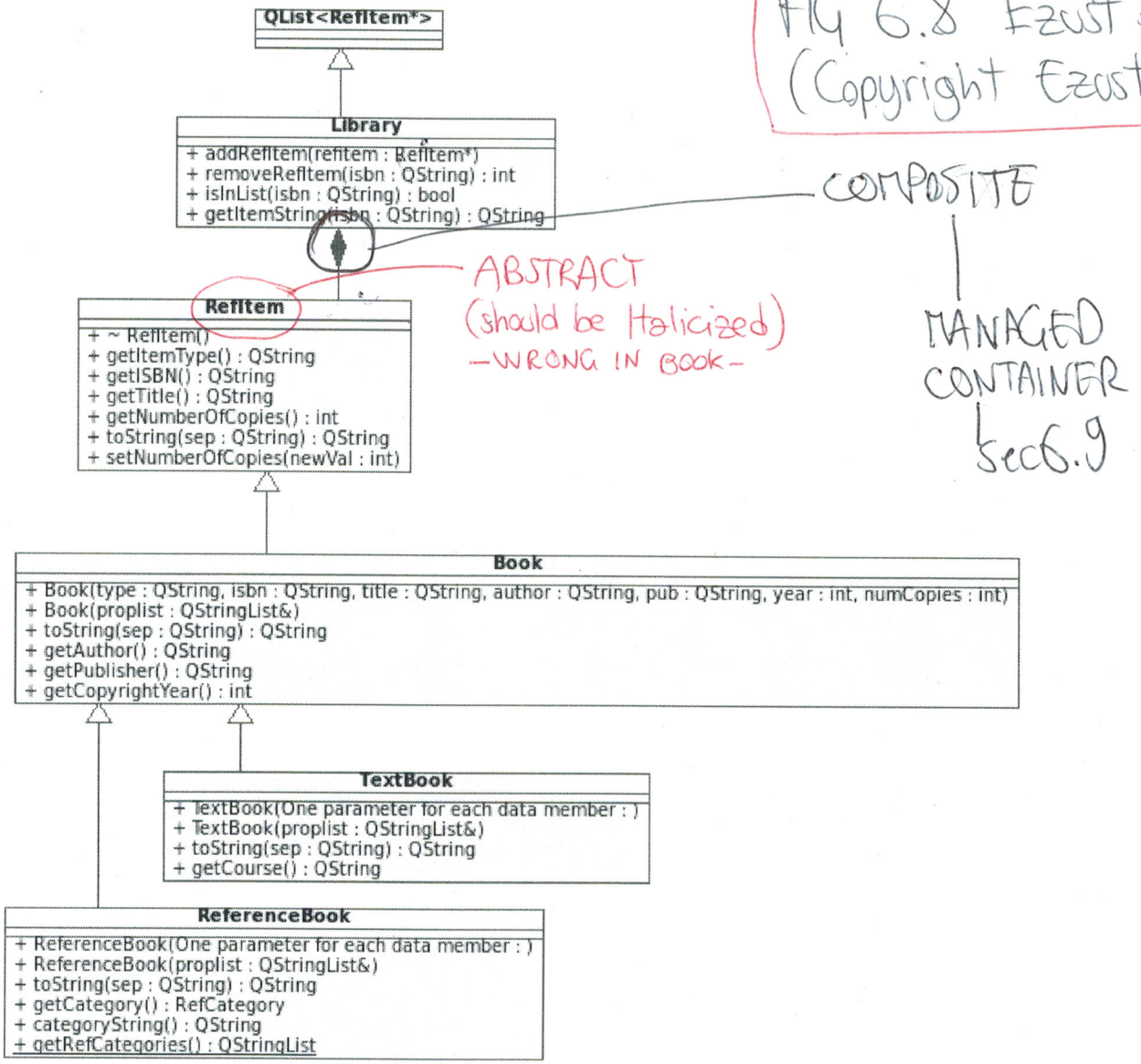
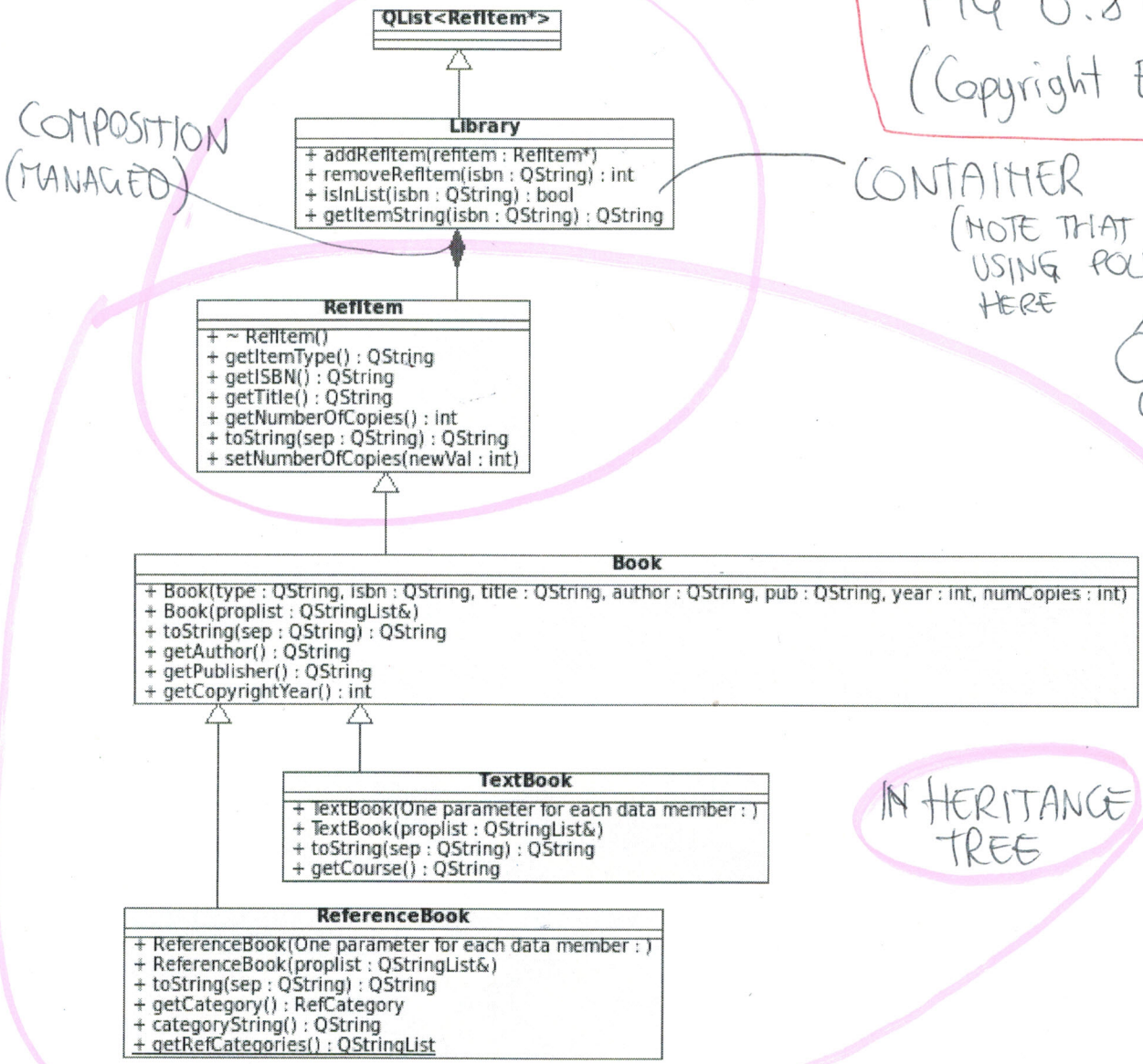


FIG 6.8 Ezust p212
(Copyright Ezust et al.)



COMPOSITION (MANAGED)

CONTAINER (NOTE THAT WE ARE USING POLYMORPHISM HERE)

OR

LIBRARY KNOWS ONLY ABOUT RefItem*

BUT IT CAN BE Book, TextBook, or ReferenceBook

INHERITANCE TREE