

## COEN 70: Formal Specification and Advanced Data Structures

Winter 2015

### Lab 6: Template class – Map

Using the classes `list<T>` and `Pair<T1, T2>` (given), create a map abstraction. A map is a collection of key/value pairs. Keys are unique, and can be used to locate the associated values. Your map should implement the following interface:

```
template<typename K, typename V>
class map
{
public:
    /**
     * Inserts a value into the map.
     * @param key the key to insert
     * @param value the value to insert
     */
    void insert(K key, V value);
    /**
     * Determines whether the map contains a given key.
     * @param key the key to check for
     */
    bool contains_key(K key);
    /**
     * Retrieves the value of a given key.
     * @param key the key to find
     * @return the value
     */
    V value_of(K key);
    /**
     * Removes the key and value.
     * @param key the key to find
     */
    void remove_key(K key);
private:
    list< pair<K, V> > data;
};
```

Include appropriate tests and submit the implementation and the application file.