Lecture 5: Collections

Exercise: Program for testing whether a word is palindrome

a) Implement generic function `reverseList` which reverse a list:

```
public static <T> List<T> reverseList(List<T> list)
```

The function accepts any kind of list and should return a new ArrayList<T> which has elements in reverse order.

For example, input list (1,2,3,4,5), output list is (5, 4,3,2,1)

b) Implement function `isPalindrome` which test if a word is a palindrome (i.e. reads the same backwards as forwards, e.g. madam) **:**

```
public static boolean isPalindrome(String word) throws NotWordException
```

For example:

input = ”madam”, output = true;

input = ”palindrom”, output = false

The function should be implemented in the following way:

1. Convert string to a list of characters using this code:

```
List<String> wordList = Arrays.asList(word.split(""));
```

2. Using `reverseList` function and iterators over lists, compare whether input word as a list of characters is equal to its reversed list of characters.

The function throws `NotWordException`, a new type of exception you should define, if input string is not a valid word. You can test for valid words using method `matches` of String class in the following way:

```
word.matches("[A-Za-z]+") // returns true if match, otherwise false
```

c) Write main program which read a word from keyboard and test whether the word is a palindrome. The program should catch exceptions and try to recover from `NotWordException` by offering the user to reenter the word again **.**