CMPT-225 Jan Manuch

Recommended Labs – Monday, June 5, 2006

1) Implement the ADT Character String by using a linked list of characters (you can choose any variant of linked list for doing so: simple linked list, circular linked list, doubly-linked list, with or without dummy node). Your implementation should be a class called CharacterString. In the following specification we will use the following notations: s,t,u,v represent arbitrary string, s+t is a concatenation of string s and t, and |s| is the length of string s. For example, if s="zip" and t="code" then s+t="zipcode" and |s|=3, |t|=4. The class should contain the following public methods:

```
// 3 constructors:
CharacterString();
// create an empty string s=""
CharacterString(char c);
// create a string of length containing c
CharasterString(CharacterString S);
// create a copy of string represented by S
// Note: it's not enough to just copy the head reference,
// as the copy should be independent from original

void append(CharacterString str);
// Pre: current object represents string s and CharacterString str represents string t
// Post: current string represents s+t and str represents the empty string ""

CharacterString split(int index);
// Pre: current object represents string s; index>=0 and index<=|s|
// Post: current object represents u and returned object represents v such that
//       u+v=s and |u|=index

int length();
// Pre: current object represents string s
// Post: return |s|

char get(int index);
// Pre: current object represents string s; index>=0 and index<=|s|-1
// Post: return the (index-1)-th character of s

int find(CharacterString str);
// Pre: current object represents string s and CharacterString str represents t
// Post: If t is not a substring of s, return -1,
//       otherwise return the smallest index at which t can be found.
//       More formally, return the smallest i such that there exists
```

```
//          string u and v such that s=u+t+v and |u|=i.

void print();
// Pre: current object represents string s
// Post: prints the string s to the screen
```

Test your implementation with the following sequence of instructions:

```
CharacterString S=new CharacterString();
S.append(new CharacterString('a'));
S.append(new CharacterString('b'));
S.append(new CharacterString('a'));
S.append(new CharacterString('a'));
S.append(new CharacterString('b'));
S.append(new CharacterString('b'));
S.append(new CharacterString('a'));
S.print();
System.out.println(S.get(2));

CharacterString T=new CharacterString(S);
T.split(3);
T=T.split(2);
T.print();

int index=S.find(T);
System.out.println(index);
```

The output of the sequence should be:

```
abaabba
a
ab
0
```

2) For those who find the above problem too easy, implement array-based implementation of linked list: see Exercise 10 on page 287 of the textbook.