

F28PL1 Programming Languages Laboratory 11

Frequency counts are held in a *count list* of pairs of characters and integers:
e.g. `[[a,1],[b,2],[c,1]]`

A)

write a rule which given a character C and an old count list, finds a new count list with the count for C incremented:

- if the old list is empty, the new list has a pair for C & 1;
- if the old list has a head which matches C, then the new list has a head with the count for C incremented and the old tail;
- if the old list has a head which doesn't match C, then the new list has the old head and a new tail with the count for C incremented in the old tail.

e.g.

```
| ?- inc(c, [[a,1],[b,2],[c,3]], L) .  
L = [[a,1],[b,2],[c,4]]
```

B)

i) write a rule which given a character and an old count list, finds a final count list with the counts for every character from the current input stream incremented in the old count list:

- if the character is end of file, the final count list is the old count list;
- otherwise:
 - the old count list is incremented for the character to give an intermediate count list;
 - another character is input from the current input stream;
 - the rule is applied to that character and the intermediate count list to form the final count list.

e.g.

```
| ?- incAll('\n', [], L) .  
|: abcaba  
|: ^D  
L = [[ '\n', 1], [a, 3], [b, 2], [c, 1], [ '\n', 1]]
```

ii) write a rule which given a file F returns a count list L by:

- opening the file F for input;
- setting the current input stream to the stream for F;
- reading the first character from the current input stream into X;
- finding the final count list L for F, starting with character X and an empty old count list;
- closing the current input stream.