

In this tutorial you will start to build a matcher. Do not look forward in the lecture series – this defeats the purpose of the tutorial.

1. write a function called *cmp* (short for compare) which compares 2 lists for equality. You must use Lisp's **eq** function for this – not the **equal** function.

```
(cmp '(cat mat bat) '(cat mat bat)) => t
(cmp '(cat mat bat) '(cat mat bat sat)) => nil
```

2. enhance *cmp* so its first argument can contain a single wild card symbol (use either "=" or "?" for this), so...

```
(cmp '(cat = bat) '(cat mat bat)) => t
```

3. enhance *cmp* further so that its first arg allows the symbols ?x. These should match like match variables – match like wild cards on their first use but then retain their value until the *cmp* function completes, so...

```
(cmp '(cat ?x mat ?x) '(cat bat mat bat)) => t
(cmp '(cat ?x mat ?x) '(cat bat mat sat)) => nil
```

4. enhance *cmp* even further so that it can deal with 3 matching variables ?x, ?y and ?z.