Informatics 41 • Fall 2005 • David G. Kay • UC Irvine

Your name ________________________________________

# FOURTH QUIZ

Your student ID ____________________________________

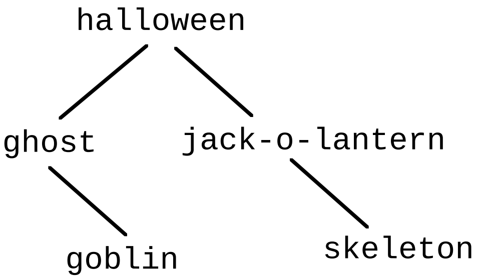
You have 15 minutes from the start of class to complete this quiz. Read the questions with care; work with deliberate speed. Don’t give us more than we ask for. The usual instructions apply. Good luck!

**Problem 1** (5 points)

Complete the definition of `keep-french-rrants` below. All the parentheses are in the correct places and each blank should be filled by exactly one symbol, function name, or constant. Restaurants are defined as usual:

```
(define-struct rrant (name cuisine phone dish price))
;; French?: rrant -> boolean
;; Return true if the input restaurant serves French cuisine
(define French?
     (lambda (R)
          (equal? ‘French (rrant-cuisine R))))
;; keep-french-rrants:  list-of-rrants -> list-of-rrants
;; Return a list containing all the French restaurants on the input list, and no others
(define keep-french-rrants
     (lambda (L)
          (cond

               ((empty? ________________) ________________)

               ((________________ (________________ L))

                (________________ (________________ L) (________________ (________________ L))))

               (else (________________ (________________ L))))))
```

**Problem 2** (5 points)



(a) (1 point) At the right is a picture of a binary search tree. Insert the value “gremlin” into the tree; draw a new branch and node to indicate where it belongs. Be careful to distinguish a left subtree from a right subtree, if necessary (by the angle of the branch).

(b) (1 point) Now insert the value “monster” into the tree.

(c) (2 points) List all seven items in the tree in the order they would be visited in an inorder traversal of the tree. In other words, if you converted this BST to a list using an inorder traversal, what would be the order of items in the list?

(d) (1/2 point) In a *preorder* traversal of the tree above, what is the value of the very *first* node visited?

(e) (1/2 point) In a *postorder* traversal of the tree above, what is the value of the very *last* node visited?

## Problem 3 (10 points)

Suppose we have a binary search tree of `rrant` structures (defined as above), with nodes defined as follows:

```
(define-struct node (key value left right))
```

where key is the `rrant`'s name (a string), value is a `rrant`, and left and right are either empty or a node and the binary search tree property holds. Complete the definition below of `add-new-rrant`, adding the necessary code in the blank spaces.

```
;; add-new-rrant:  string   rrant   BST-of-rrants   ->   BST-of-rrants
;; Insert the input rrant into the input BST according to the input string (the name
;; of the rrant to be added); return the new tree
(define add-new-rrant
  (lambda (new-key new-rrant T)
    (cond
      ((empty? T) (make-node



         ))
      ((string<? new-key (node-key T)) (make-node



         ))
      ((string>? new-key (node-key T)) (make-node



         ))
      (else T)))) ; No need to add rrant that's already in tree
```