

Office Hours

CS-151

HWS

Problem 1

(b)

back-one

back-one - input is a list of activities and output is a list of activities.

We are suppose to do the same operation to every activity in the list.

Step 1: Write function which takes an activity and moves it back by 1 day.

(define-type Day-of-Week (U 'Monday 'Tuesday 'Wednesday ...))

(: back-one-activity : Activity
→ Activity)

(define (back-one-activity ac)

Activity (Activity-desc ac)

(back-a-day (Activity-day ac))

(Activity-location ac)))

(: back-a-day : Day-of-Week → Day-of-Week)

(define (back-a-day d)

(match d

['Sunday 'Saturday]
['Monday 'Sunday]
.
.
.
))

(: back-one : Calendar → Calendar)
(define (back-one cal)

(map back-one-activity cal))

c)

(: in-ry251? : Activity \rightarrow Boolean)

(define (in-ry251? act)

(string=? "Ryerson 251" (Activity-location
act)))

(: ry251 : Calendar \rightarrow Integer)

(define (ry251 cal)

(length (filter in-ry251? cal))))

2:

(a). apply-one-or-the-other

First input: $(A \rightarrow \text{Real}) \Rightarrow (\rightarrow A \text{ Real})$
Second input $(B \rightarrow \text{Real})$

Third input $(\vee A B)$

Output : Real

$(\text{All } (A B) (A \rightarrow \text{Real})$
 $(B \rightarrow \text{Real}) (\vee A B)$
 $\rightarrow \text{Real})$

(b) find-maximizes

Input: (List of A)

Input: $(A \rightarrow \text{Real})$

Output: A

$(\text{All}(A) (\text{Listof } A) (A \rightarrow \text{Real}))$
 $\rightarrow A)$