Appendix C

Recursive computation of the difference between dates

; Return the number of days spanned by earlier-date and later-date.
; Earlier-date and later-date both represent dates in 2002,
; with earlier-date being the earlier of the two.
(define (day-span earlier-date later-date)
  (cond
   ((same-month? earlier-date later-date)
    (same-month-span earlier-date later-date) )
   ((consecutive-months? earlier-date later-date)
    (consec-months-span earlier-date later-date) )
   (else
    (general-day-span earlier-date later-date) ) ) )

; Access functions for the components of a date.
(define (month-name date) (first date))
(define (date-in-month date) (first (butfirst date)))

; Return true if dat1 and date2 are dates in the same month, and
; false otherwise. Dat1 and date2 both represent dates in 2002.
(define (same-month? date1 date2)
  (equal? (month-name date1) (month-name date2)))

; Return the number of the month with the given name.
(define (month-number month-name)
  (cond
   ((equal? month-name 'january) 1)
   ((equal? month-name 'february) 2)
   ((equal? month-name 'march) 3)
   ((equal? month-name 'april) 4)
   ((equal? month-name 'may) 5)
   ((equal? month-name 'june) 6)
   ((equal? month-name 'july) 7)
   ((equal? month-name 'august) 8)
   ((equal? month-name 'september) 9)
   ((equal? month-name 'october) 10)
   ((equal? month-name 'november) 11)
   ((equal? month-name 'december) 12) ) )

; Return true if dat1 is in the month that immediately precedes the
; month date2 is in, and false otherwise.
; Date1 and date2 both represent dates in 2002.
(define (consecutive-months? date1 date2)
  (= (month-number (month-name date2))
     (+ 1 (month-number (month-name date1))) ) )

; Return the difference in days between earlier-date and later-date,
; which both represent dates in the same month of 2002.
(define (same-month-span earlier-date later-date)
  (+ 1
     (- (date-in-month later-date) (date-in-month earlier-date)) ) )
; Return the number of days in the month named month-name.
(define (days-in-month month-name)
  (cond
    ((equal? month-name 'january) 31)
    ((equal? month-name 'february) 28)
    ((equal? month-name 'march) 31)
    ((equal? month-name 'april) 30)
    ((equal? month-name 'may) 31)
    ((equal? month-name 'june) 30)
    ((equal? month-name 'july) 31)
    ((equal? month-name 'august) 31)
    ((equal? month-name 'september) 30)
    ((equal? month-name 'october) 31)
    ((equal? month-name 'november) 30)
    ((equal? month-name 'december) 31) ) )

; Return the number of days remaining in the month of the given date,
; including the current day. Date represents a date in 2002.
(define (days-remaining date)
  (+ 1 (- (days-in-month (month-name date)) (date-in-month date)))
)

; Return the difference in days between earlier-date and later-date,
; which represent dates in consecutive months of 2002.
(define (consec-months-span earlier-date later-date)
  (+ (days-remaining earlier-date) (date-in-month later-date))
)

; Return the name of the month with the given number.
; 1 means January, 2 means February, and so on.
(define (name-of month-number)
  (item
    month-number
    '(january february march april may june
     july august september october november december) )
)

; Return the sum of days in the months represented by the range
;   first-month ... last-month.
;   first-month and last-month are integers; 1 represents January, 2 February,
;   and so on. This procedure uses recursion.
(define (day-sum first-month last-month)
  (if (> first-month last-month) 0
    (+ (days-in-month (name-of first-month))
      (day-sum (+ first-month 1) last-month)) )
)

; Return the number of the month that immediately precedes the month
; of the given date. 1 represents January, 2 February, and so on.
(define (prev-month-number date)
  (- (month-number (month-name date)) 1) )

; Return the number of the month that immediately follows the month
; of the given date. 1 represents January, 2 February, and so on.
(define (next-month-number date)
  (+ (month-number (month-name date)) 1) )

; Return the difference in days between earlier-date and later-date,
; which represent dates neither in the same month nor in consecutive months.
(define (general-day-span earlier-date later-date)
  (+ (days-remaining earlier-date)
    (day-sum
      (next-month-number earlier-date)
      (prev-month-number later-date) )
    (date-in-month later-date) ) )