```
; HW6 reference solution
; CMPSC 461, Spring 2022, Dr. Gary Tan
; Q1

(define (ackermann m n)
  (cond ((= m 0) (+ n 1))
        ((and (> m 0) (= n 0)) (ackermann (- m 1) 1))
        (else (ackermann (- m 1) (ackermann m (- n 1))))))

(define (bind k v al)
  (cons (list k v) al))

(define (lookup k al)
  (cond ((null? al) #f)
        ((equal? k (caar al)) (cadar al))
        (else (lookup k (cdr al)))))

(define al '())

(define (ackermann_mem m n)
  (let ((v (lookup (list m n) al)))
    (if (equal? v #f)
        (let ((res (ackermann m n)))
          (begin
            (set! al (bind (list m n) res al))
            res))
        (begin
            (display "memoization hit\n")
            v))))

; Q2

(define (construct_mem f)
  (let ((al '()))
    (lambda (x y)
      (let ((v (lookup (list x y) al)))
        (if (equal? v #f)
            (let ((res (f x y)))
              (begin
                (set! al (bind (list x y) res al))
                res))
            (begin
              (display "memoization hit\n")
              v))))))
```