Mid-term Exam for Programming Methodologies (Fall of 2007)
Instructor: Kyuseok Shim
You are not going to win at everything in life. Go out there and do your best.
When it is over, congratulate the winner – if it's not you!
– By Gail Denvers's father (Gail won a gold medal at the 1992 Olympic)

1. Consider the following program written in Scheme syntax.

```
(define (tnu m n)
    (begin (define (g m) (- m n))
           (define (h k) (* m (g k)))
           (+ (h m) n) ))
```

   If this program is a pure Lisp program, it will use *dynamic* scoping. If this program is a Common Lisp program, it will use *static* scoping. Then write the output of the function call (tnu 7 3) assuming this program is

   (a) pure Lisp code: [5 points]

   (b) Common Lisp code: [5 points]

   (In order to get full credit, you have to describe how each output is calculated by showing the environment of the code at run time.)

2. Consider the following two Scheme functions:

```
(define (foo bar baz)
    (cond ( (null? bar) baz )
          ( (cat (car bar) baz) (foo (cdr bar) baz) )
          ( else (cons (car bar) (foo (cdr bar) baz)) )
    )
)

(define (cat foo bar)
    (cond ( (null? bar) #f )
          ( (equal? foo (car bar)) #t)
          ( else (cat foo (cdr bar)) )
    )
)
```

   What is the value of `(foo '(a b c d e) '(b a e))`? [10 points]

   `SOLUTION: (c d b a e)`

3. Draw a tree showing Scheme interpreter's internal representation for `abc` after the following expression is evaluated: [5 pts.]

   `(define abc '((1.2) () (3 4)))`

4. Consider the following mystery function:

```
(define (foo M L)
   (cond ((null? L) '())
         ((number? L) (M L))
         (else (cons (foo M (car L)) (foo M (cdr L))))

   )
)
```

   (a) For each of the following expressions, indicate if the expression is legal and if so, what is going to return. Note that `zero?` is a built-in predicate that tests if its argument is 0.

      i. (foo (lamda (x y) (+ x y)) '(1 2 3 4)) [5 points]

      ii. (foo (lamda (x) (+ 1 x)) '(1 (2 (3 4) 5) 6)) [5 points]
      `SOLUTION:(Illegal:  the second argument should be a function which takes one argument.`
      `(2 (3 (4 5) 6) 7)`

   (b) What does foo do in general? [5 points]

   (c) Write foo using map which is a built-in function in Scheme. [5 points]

5. `len` is a function that returns the length of the input list `L`. Fill the blanks below to implement `len`.

   (a) Give the definition of a *recursive* function for `len`. [5 points]

```
(define len (lamda (L)
   (if (null? L)

   (                        ))))
```

   (b) Complete the definition of a *tail-recursive* function `len0` and its caller `len`. [5 points]

```
(define len0 (lamda (L t)
   (if (null? L)

   (                      ))))
(define len (lambda (L)
   (len0             )))
```

`SOLUTION:`

```
(a)(define len (lambda (L)
       (if (null? L)
```

```
        0
        (+ 1 (len (cdr L))))))
```

```
(b)(define len0 (lambda (L t)
       (if (null? L)
        t
        (len0 (cdr L) (+ 1 t)))))
   (define len (lambda (L)
        (len0 L 0)))
```

6. Given the following C++ code, answer the questions.

```
line number               program
 1          class polar {
 2             double radius, theta;
 3
 4             public:
 5             polar(double r, double t) { // two-parameter constructor
 6                radius = r; theta = t;
 7             }
 8             void foo1(int x, int y) {
 9                x = x + y;
 10            }
 11            void foo2(int *x, int *y) {
 12               *x = *x + *y;
 13            }
 14            void foo3(int &x, int &y) {
 15               x = x + y;
 16            }
 17            friend istream& operator>>(istream&, polar&);
 18         };
 19
 20         // input format : 2r30t
 21         istream& operator>>(istream& s, polar& p) {
 22            // fill the code here!
 23         }
 24
 25         int main () {
 26            int x=3, y=5;
 27            polar p(5,60);
 28            cin >> p;
 29            p.foo1(x, y); cout << x << ", " << y << endl;
 30            p.foo2(&x, &y); cout << x << ", " << y << endl;
 31            p.foo3(x, y); cout << x << ", " << y << endl;
 32            return 1;
 33         }
```

(a) Show the printed results of lines 29, 30 and 31, respectively. [10 points]

(b) Complete the overload function `operator>>` at line 21 so that `cin` at line 28 can be valid. We assume that the input is "2r30t" if the radius is 2 and theta is 30 in a polar coordinate. [10 points]

SOLUTION:

(a) 3, 5 8, 5 13, 5

(b)
```
// input format : 2r30t
istream& operator>>(istream& s, polar& p) {
double r, t;
char c;
s >> r >> c;
s >> t >> c;
p = polar(r, t);
return s;
}
```

7. Given a C++ program below, find all the illegal statements from the program, and mark X on them. Explain why the statements are illegal. Also, mark O on the remaining legal statements. [15 points]

```
            int* bar(int *a) {
            int *b;
               .
               .
(   )          a[2] = *b%3;
(   )          return a+1;
            }

            main () {
               int i, j, k, m[10];
               char d;
                  .
                  .
               k = 3;
(   )          i = bar(m)+k;
(   )          *(m+2) = d + 10;
(   )          j = &m + k;
            }
```

SOLUTION: O O X O X X O O X O O O X O X

8. Given the C++ program below, answer the following questions.

`line number` `program`

```
1   #include <stream.h>
2   inline void foo(char &d) {
3      d = (d == ’x’ ? ’2’ : ’3’);
4   }
5   int sum(int i, int j, int k = 0, char c = ’x’) {
6      foo(c);
7      if (c == ’3’) return i + j + k;
8      return i + j;
9   }
10  main() {
11     int a[5] = { 10, 20, 30, 0, 0};
12     a[3] = sum(a[0], a[1], a[2]);
13     a[4] = sum(a[1], a[2], a[3], ’y’);
14     cout << a[3] << ’ ’ << a[4];
15  }
```

(a) What is the output of the line 14? [5 points]

(b) `foo` is an inline subroutine. Replace the original code on line 6 with the code after `foo` is inlined. [5 points]

`SOLUTION:` (a) 30 80 (b) c = (c == ’x’ ? ’2’ : ’3’);