ML PATTERN MATCHING

1. What do each of the following ML expressions evaluate to?

(a) case (3, 1, "Hello") of
   (\_, x, "Y") => "Y"
   | (x,\_, \_) => Int.toString(x)

(b) let
    val x = [1, 3, 5];
    in
    case x of
    nil => 21
    | x::y::\_ => x+y
    | x::\_ => x
    end;

(c) case ("a", "b", ["c","d"]) of
    (x, y, z::\_) => z ^ y ^ x
    | (x, y, \_) => y ^ x

(d) let
    fun f(x::xs) = x * f(xs)
    | f(nil) = 1;
    in
    f([1,2,3,4,5,6])
    end;

SCALA PATTERN MATCHING

2. Translate the following ML code into Scala using case classes and pattern matching:

```scala
datatype tree = LEAF of int | NODE of (int * tree * tree);

val myTree = NODE(1, NODE(2, LEAF 3, LEAF 4), LEAF 5);

fun inTree(x, LEAF(y)) = x = y
| inTree(x, NODE(y, a, b)) =
    x = y orelse
    inTree(x, a) orelse
    inTree(x, b);

inTree(7, myTree);
inTree(3, myTree);
inTree(5, myTree);
```

(over)
3. (a) What is a type?
(b) List three reasons types are used in programming languages.

4. (a) Define type error.
(b) Give an example of an operation and operands that would produce a hardware exception.
(c) Give an example of an operation and operands that would not produce a hardware exception but would not behave as expected because of the type of the operands.
(d) Which of these two examples are type errors? (answer ‘neither,’ ‘first,’ ‘second,’ or ‘both’)

5. (a) Define type safety.
(b) List three language features found in C, C++, and/or Pascal that cause those languages to not be type safe.

6. (a) List one advantage to using run-time type instead of compile-time type checking?
(b) List two advantages of compile-time type checking over run-time type checking?
(c) Why does compile-time type-checking need to be conservative if it is to be safe?

7. (a) What is polymorphism?
(b) What is parametric polymorphism? Give an example of an ML program exhibiting parametric polymorphism and explain how the example demonstrates parametric polymorphism.
(c) What is subtype polymorphism? Give an example of a Scala program exhibiting subtype polymorphism and explain how the example demonstrates subtype polymorphism.
(d) What is ad-hoc polymorphism? Give an example of a Scala program exhibiting ad-hoc polymorphism and explain how the example demonstrate ad-hoc polymorphism.

8. What is type inference? How does it differ from simple type checking?

Note: If you collaborated with your classmates or used their notes, please note which classmates you collaborated with. If you use an external source, besides the text book, lectures, notes provided by the instructor, and your own intellect, please cite that source. Use quote marks if you are quoting material word-for-word from any source (including the text book).