CSC 301 – Practice Midterm

NAME_________________________

Part I - Short Answers (20 points, 5 points each)

1. Which programming language class do languages such as FORTAN, Pascal, and C belong to?

2. Name the stages of a typical programming language translator?

3. A data type can be considered a ______________.

4. What is a widening type conversion?
Part II - Problems (80 points)

1. Analyze the following short SML programs.

(a) Given the following SML function f and the input [1,2,3,4]; what is the value of it? Give both value and type of the result. Justify your answer. (5 points)

- fun f ([]) = 0
  | f(H::T) = H + f(T);
- f([1,2,3,4]);
  Val it = ?????

(b) Given the following SML function f and the input [1,2,3,4]; what is the value of it? Give both value and type of the result. Justify your answer. (5 points)

- fun g ([]) = []
  | g(H::T) = g(T) @ [H];
- g([1,2,3,4]);
  Val it = ?????

(c) Given the following SML code snippet; what is the value of h? Give both value and type of the result. Justify your answer. (5 points)

- val h = fn (a:int,b:int) => a + b;
  val h = ?????

(d) Given the following SML code snippet; what is the value of it? Give both value and type of the result. Justify your answer. (5 points)

- map (fn a => a - 1) [1,2,3,4];
  val it = ?????
2. Consider the following grammar for a small programming language:

```
Grammar
*<Expr> ::= <Expr> + <Term> | <Term>
<Term> ::= <Term> * <Prim> | <Prim>
<Prim> ::= α | β | γ
```

(a) What are the terminal symbols in this grammar? (5 points)

(b) What are the non-terminal symbols in this grammar? (5 points)

(c) Circle the valid sentences that are part of the language that this grammar generates (10 points):

\[
\alpha \beta + \gamma \quad \beta + \gamma + \alpha \quad \alpha * \beta + \eta \quad \alpha - \beta * \gamma
\]

(d) Show the parse tree for \(\alpha \ast \beta + \gamma\) (10 points)

(e) Is the grammar ambiguous? Why? Why not? (10 points)
3. Consider the following class hierarchy in Java:

```java
class Fruit {
    ...
};

class Orange extends Fruit {
    ...
};

class Apple extends Fruit {
    ...
};
```

Does Java allow the following statements or does it generate an error? Explain your answer. (5 points each)

(a) `Fruit f = new Orange();`

(b) `Apple a = new Fruit();`

(c) `Apple a = new Orange();`

(d) `void funFunction(Fruit f) {
      ...
    }
    ...
    funFunction(new Apple());`