一、Mark each statement *true* or *false* (2 points each)

1. A grammar is ambiguous if it has two different derivations or two different parse trees for a sentence..
2. If a grammar is LR(I), but not LALR(I). There are not shift-reduce conflicts in its parsing table of LALR(1).

二、Single Choice (2 points each)

1、In the Top-Down Parsing, the action ( ) will never be used.

[A] Shift [B] Match [C] Generate [D] Accept

2、In the Top-Down Parsing, the action ( ) will never be used.

[A] Shift [B] Match [C] Generate [D] Accept

3、IF one CFG grammar contains two non-terminals 'A','B' and two terminal 'a','b', where 'A' is the start symbol, then the Follow set of 'A' may be ( )

[A] {a, b} [B] {a, b, $} [C] {a, b, ε} [D] {a, b, B}

三、Questions (40 cents)

1、Convert the following regular expression (a|b)*abb(a|b)* to the minimum deterministic finite automata. **(10 cents)**

2、Consider the following grammar of simplified C declarations:

declaration → *type* *var-list*

type → **int** | **float**

var-list → **identifier**, *var-list* | **identifier**

(a) Left factor this grammar. (**5 cents**)

(b) Construct First and Follow sets for the nonterminals of the resulting grammar.(**10 cents**)

(c) Construct the LL(1) parsing table for the resulting grammar. **(10 cents)**

(d) Show the actions of the corresponding LL(1) parser, give the input string **int x,y,z**. **(5 cents)**