2.9)

S → TC | AR
T → aTb | ε
R → bRc | ε
C → cC | ε
A → aA | ε

The grammar is ambiguous because any string $a^ib^jc^k$ with $i = j = k$ can be derived ambiguously.

2.10)

2.10) The top branch is $i = j$; the bottom branch is $j = k$.

Explanation for 2.10:
Top branch:
Push a’s, then pop a’s while reading b’s. If $i = j$, then the stack will have $\$ \$ on top so we can go to the final state where we skip c’s and accept.
Bottom branch:
Skip a’s, then push b’s, then pop b’s while reading c’s. If $j = k$, then the stack will have $\$ \$ on top so we can go to the final state and accept.
2.11)
2.12)