1. Because we need more nonterminals than the number of letters between S and Z, we just continue using nonterminals A and B.

S → `array` T `of` Y`;`
T → `[`U`]` $\big|$ V
U → U`,`W $\big|$ W
V → V`[`W`]` $\big|$ `[`W`]`
W → X`..`X $\big|$ Y`..`Y $\big|$ Z`..`Z $\big|$ A`..`A
X → `true` $\big|$ `false`
Y → Y`a` $\big|$ Y`b` $\big|$ Y`c` $\big|$ `a` $\big|$ `b` $\big|$ `c`
Z → `'x'` $\big|$ `'y'` $\big|$ `'z'`
A → `+`B $\big|$ `-`B $\big|$ B
B → B`0` $\big|$ B`1` $\big|$ B`2` $\big|$ B`3` $\big|$ B`4` $\big|$ B`5` $\big|$ B`6` $\big|$ B`7` $\big|$ B`8` $\big|$ B`9` $\big|$ `0` $\big|$ `1` $\big|$ `2` $\big|$ `3` $\big|$ `4` $\big|$ `5` $\big|$ `6` $\big|$ `7` $\big|$ `8` $\big|$ `9`

2.   a.   S → $aaSbb \mid \epsilon$

       b.   S → TW
           T → Uc
           U → aU$cc \mid$ V
           V → V$b \mid \epsilon$
           W → W$d \mid \epsilon$

       c.   S → $aSa \mid bSb \mid aa \mid bb$

       d.   Rewrite the language as $a^m b^n a^n a^m$, which makes it easier to see how the grammar can be written.
           S → $aSa \mid$ T
           T → $bTa \mid ba$

       e.   Rewrite the language as $a^{n-m} a^m b^m a^{n-m}$, which makes it easier to see how the grammar can be written.
           S → $aSa \mid$ T
           T → $aTb \mid \epsilon$

       f.   S → $(S)S \mid \epsilon$

       g.   S → aT $\mid b$U $\mid \epsilon$
           T → bS $\mid a$TT
           U → aS $\mid b$UU

       h.   S → UT $\mid$ TV
           T → aT$b \mid \epsilon$
           U → U$a \mid a$
           V → V$b \mid b$

       i.   S → aS$d \mid a$T$c \mid b$U$d \mid b$V$c \mid \epsilon$
           T → aT$c \mid b$V$c \mid \epsilon$
           V → bV$c \mid \epsilon$
           U → bU$d \mid b$V$c \mid \epsilon$

3. S → T $\mid$ V
    T → UU
    U → aU$b \mid ab$
    V → aV$b \mid a$Wb
    W → bW$a \mid ba$

To show the grammar is ambiguous show two leftmost derivations, or two rightmost derivations, or two parse trees, for a string such as $abab$. Here are two leftmost derivations:

S $\Longrightarrow$ T $\Longrightarrow$ UU $\Longrightarrow$ abU $\Longrightarrow$ $abab$
S $\Longrightarrow$ V $\Longrightarrow$ aWb $\Longrightarrow$ $abab$

You can construct the corresponding parse trees.