Q.1

The production rules are:

S → aS

S → ab

The rule S → aS can be applied any number of times (including zero times) that will generate the form an where n ≥ 0.

Then we can apply the rule S → ba as a suffix to any number of strings generated in the previous step.

Therefore, any string generated by this CFG will be of the form a\*ba.

Q.2

The CFGs for the language with all words that do not have the substring ‘ab’:

S → bS | aA

A → aA | ε

Q.3

In CFG 1, it cannot generate ‘abba’ because the base word close to ‘abba’ that this CFG can generate is ‘aabbab’. In CFG 3, it cannot generate ‘abba’ because the base word close to ‘abba’ that this CFG can generate is ‘aabba’. In CFG 4, it cannot generate ‘abba’ because the base word close to ‘abba’ that this CFG can generate is ‘aabba’.

CFG 2 can generate the word ‘abba’ and the syntax tree is:

S

/ \

a S

/\

b S

/\

b S

|

a

CFG 5 can generate the word ‘abba’ and the syntax tree is:

S

/ \

a B

/\

b S

/\

b A

|

a

Q.4

Converted to CNF:

S → AX | YB

X → MN | a

Y → b | BY | B

Z → a

A → a

B → b

M → ZX

N → ZY

Q.5

