

1 CFGs

Give CFG grammars (using BNF notation) for the following languages or say it cannot be done

1. All strings over the alphabet $\{0,1,2,3\}$ sorted in decreasing order.

Answer:

```
<S> ::= <threes> <twos> <ones> <zeros> | epsilon
<threes> ::= 3 <threes> | epsilon
<twos> ::= 2 <twos> | epsilon
<ones> ::= 1 <ones> | epsilon
<zeros> ::= 0 <zeros> | epsilon
```

2. All strings over the alphabet $\{a,b\}$, in which every b is both immediately preceded by and followed by at least one a.

Answer:

```
<S> ::= a <BA> <S> | epsilon
<BA> ::= b a <BA> | epsilon
```

3. $a^n b^m a^n b^m$ $n, m \geq 0$

cannot be expressed by CFG

2 english descriptions

Give English descriptions of the languages described by each of the following grammars.

a) $\langle S \rangle ::= \langle S \rangle a \langle S \rangle b \langle S \rangle \mid \langle S \rangle b \langle S \rangle a \langle S \rangle \mid \text{epsilon}$

Answer:

All strings with an equal number of a's and b's.

Example:

```
 $\langle S \rangle \Rightarrow \langle S \rangle a \langle S \rangle b \langle S \rangle$   
 $\Rightarrow \langle S \rangle b \langle S \rangle a \langle S \rangle a \langle S \rangle b \langle S \rangle$   
 $\Rightarrow \langle S \rangle b \langle S \rangle aa \langle S \rangle b \langle S \rangle$   
 $\Rightarrow \langle S \rangle b \langle S \rangle aa \langle S \rangle b \langle S \rangle a \langle S \rangle b \langle S \rangle$   
 $\Rightarrow \text{baabab}$ 
```

b) $\langle S \rangle ::= a \langle S \rangle a \mid b \langle S \rangle b \mid c \langle S \rangle c \mid \text{epsilon}$

Answer: All even length strings over {a,b,c}, where the second half of the string is the reverse of the first half.

Example:

```
 $\langle S \rangle \Rightarrow a \langle S \rangle a$   
 $\Rightarrow a b \langle S \rangle b a$   
 $\Rightarrow a b c \langle S \rangle c b a$   
 $\Rightarrow a b c b \langle S \rangle b c b a$   
 $\Rightarrow \text{abcb bcba}$ 
```

c) Adjust the grammar in (b) to make it generate all palindromes.

What do we need? We need to also generate odd length strings.

Answer: $\langle S \rangle ::= a \langle S \rangle a \mid b \langle S \rangle b \mid c \langle S \rangle c \mid a \mid b \mid c \mid \text{epsilon}$

3 ambiguity

a) $\langle S \rangle ::= \langle S \rangle \text{ and } \langle S \rangle \mid \langle S \rangle \text{ or } \langle S \rangle \mid a \mid b \mid c$

Is this grammar ambiguous?

Yes. Two distinct parse trees for string ‘‘a and b or c’’.

Develop a grammar that generates the same strings,
but is not ambiguous.

```
 $\langle S \rangle ::= \langle OR \rangle$   
 $\langle OR \rangle ::= \langle AND \rangle \mid \langle OR \rangle \text{ or } \langle AND \rangle$   
 $\langle AND \rangle ::= \langle VAR \rangle \mid \langle AND \rangle \text{ and } \langle VAR \rangle$   
 $\langle VAR \rangle ::= a \mid b \mid c$ 
```

b) $\langle S \rangle ::= a \langle S \rangle a \mid b \langle S \rangle b \mid \text{epsilon}$

Not ambiguous.