

MCS 441 Theory of Computation

Two Context Free Grammars

A Fragment of English

| | | |
|------------------------|---|---|
| ⟨Sentence⟩ | → | ⟨Noun Phrase⟩⟨Verb Phrase⟩. |
| ⟨Noun Phrase⟩ | → | ⟨Name⟩ ⟨Complex Noun⟩ ⟨Complex Noun⟩⟨Prepositional Phrase⟩ |
| ⟨Prepositional Phrase⟩ | → | ⟨Preposition⟩⟨Complex Noun⟩ |
| ⟨Complex Noun⟩ | → | ⟨ Article⟩⟨Noun⟩ ⟨ Article⟩⟨Adjective⟩⟨Noun⟩ |
| ⟨Verb Phrase⟩ | → | ⟨Verb⟩ ⟨Adverb⟩⟨Verb Phrase⟩ ⟨Verb⟩⟨Noun Phrase⟩ |
| ⟨Article⟩ | → | a the |
| ⟨Noun⟩ | → | boy girl flower cake book library |
| ⟨Name⟩ | → | Alice Bob Carol |
| ⟨Adjective⟩ | → | tall shy green |
| ⟨Verb⟩ | → | likes eats touches sees |
| ⟨Adverb⟩ | → | quickly softly |
| ⟨Preposition⟩ | → | with under at |

Sample Sentences

the tall girl softly touches the green flower

Bob eats the shy cake

the shy boy with the green book likes the tall girl with the green flower

A Simple Programming Language

| | | |
|--|---------------|--|
| $\langle \text{Statement} \rangle$ | \rightarrow | $\langle \text{Begin-Statement} \rangle \mid \langle \text{If-Statement} \rangle \mid$ $\langle \text{While-Statement} \rangle \mid \langle \text{Assignment} \rangle$ |
| $\langle \text{Begin-Statement} \rangle$ | \rightarrow | begin $\langle \text{Statement-list} \rangle$ end |
| $\langle \text{Statement-List} \rangle$ | \rightarrow | $\langle \text{Statement} \rangle \mid \langle \text{Statement} \rangle \langle \text{Statement-list} \rangle$ |
| $\langle \text{While-Statement} \rangle$ | \rightarrow | while $\langle \text{Boolean} \rangle$ do $\langle \text{Statement} \rangle$ |
| $\langle \text{If-Statement} \rangle$ | \rightarrow | if $\langle \text{Boolean} \rangle$ then $\langle \text{Statement} \rangle$ else $\langle \text{Statement} \rangle$ |
| $\langle \text{Assignment} \rangle$ | \rightarrow | $\langle \text{Variable} \rangle := \langle \text{Expression} \rangle$ |
| $\langle \text{Boolean} \rangle$ | \rightarrow | $\langle \text{Expression} \rangle \langle \text{Compare-op} \rangle \langle \text{Expression} \rangle$ |
| $\langle \text{Expression} \rangle$ | \rightarrow | $\langle \text{Variable} \rangle \mid \langle \text{Number} \rangle \mid$ $\langle \text{Expression} \rangle \langle \text{Arith-op} \rangle \langle \text{Expression} \rangle$ |
| $\langle \text{Arith-op} \rangle$ | \rightarrow | $+$ $ $ $-$ $ $ $*$ $ $ $/$ |
| $\langle \text{Compare-op} \rangle$ | \rightarrow | $<$ $ $ $>$ $ $ \leq $ $ \geq $ $ $=$ $ $ \neq |
| $\langle \text{Variable} \rangle$ | \rightarrow | a $ $ b $ $ \dots $ $ z |
| $\langle \text{Number} \rangle$ | \rightarrow | $\langle \text{Digit} \rangle \mid \langle \text{Digit} \rangle \langle \text{Number} \rangle$ |
| $\langle \text{Digit} \rangle$ | \rightarrow | 0 $ $ 1 $ $ \dots $ $ 9 |

Sample Statements

We will use spacing and indentations that make them easier to read

```
1)
begin
x : = 0
y : = 0
while x < 10 do
  begin
  x:= x+1
  y:= y*x
  end
end

2)
if 2*y+1=z then x:=0
else
begin
x:=1
z:=z-y
end
```