UMB CS 622

Non-CFLs

Wednesday, March 27, 2024

Announcements

- HW 6
  - Due Monday 4/1 12pm noon

Last Time

Application of this class: Compilers

A program string (chars) (e.g., `a := (5 + 3);` ...)

DFAs (recognizing regular languages) in here!

Program "words"
(e.g., `ID(a) ASSIGN LPAREN NUM(5) PLUS NUM(3) RPAREN SEMI` ...)

Last Time

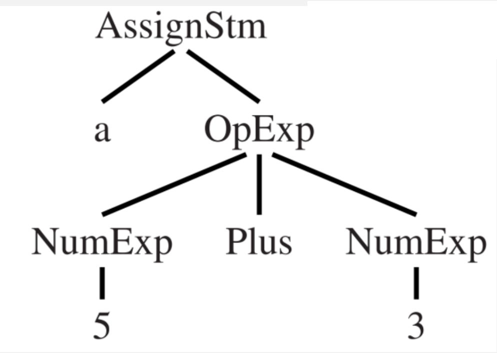
Application of this class: Compilers

A program string (chars) (e.g., `a := ( 5 + 3 ) ;` ...)

DFAs (recognizing regular languages) **in here!**

Program "words"
(e.g., `ID(a) ASSIGN LPAREN NUM(5) PLUS NUM(3) RPAREN SEMI` ...)

DPDAs (recognizing DCFLs) **in here!**



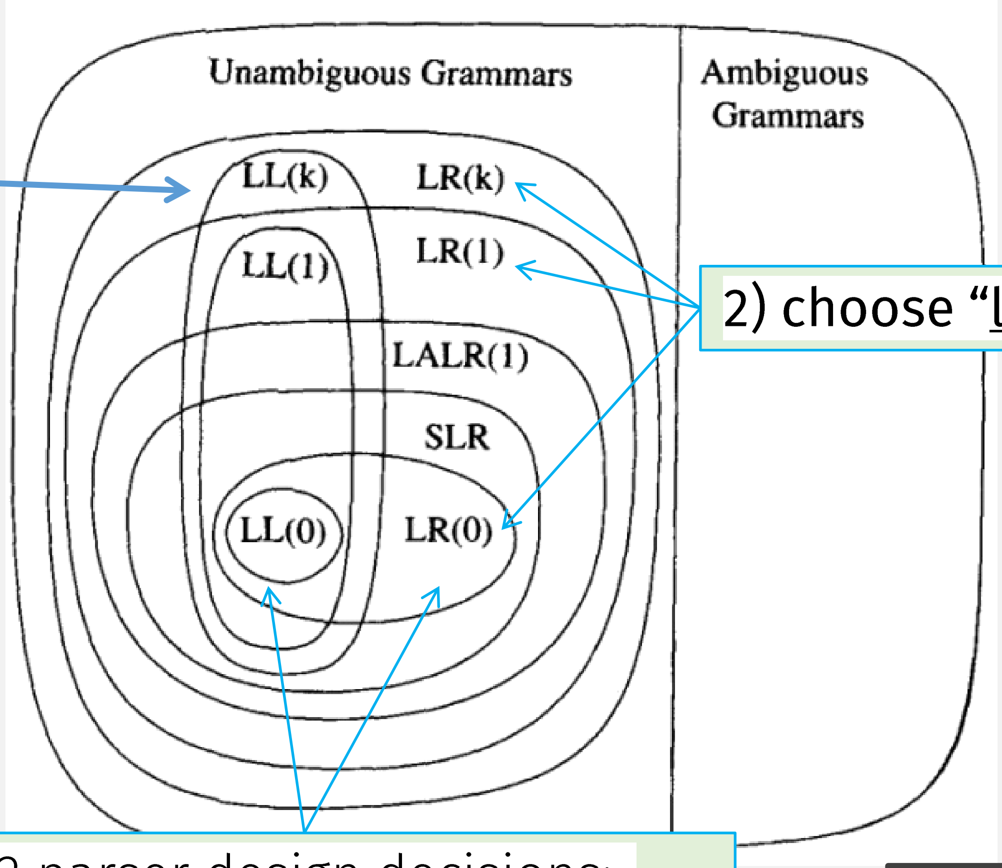
Syntax tree (AST), i.e., a **parse tree!**

Last Time

Subclasses of CFLs

DCFLs

Programming language parsers / compilers are ideally in here

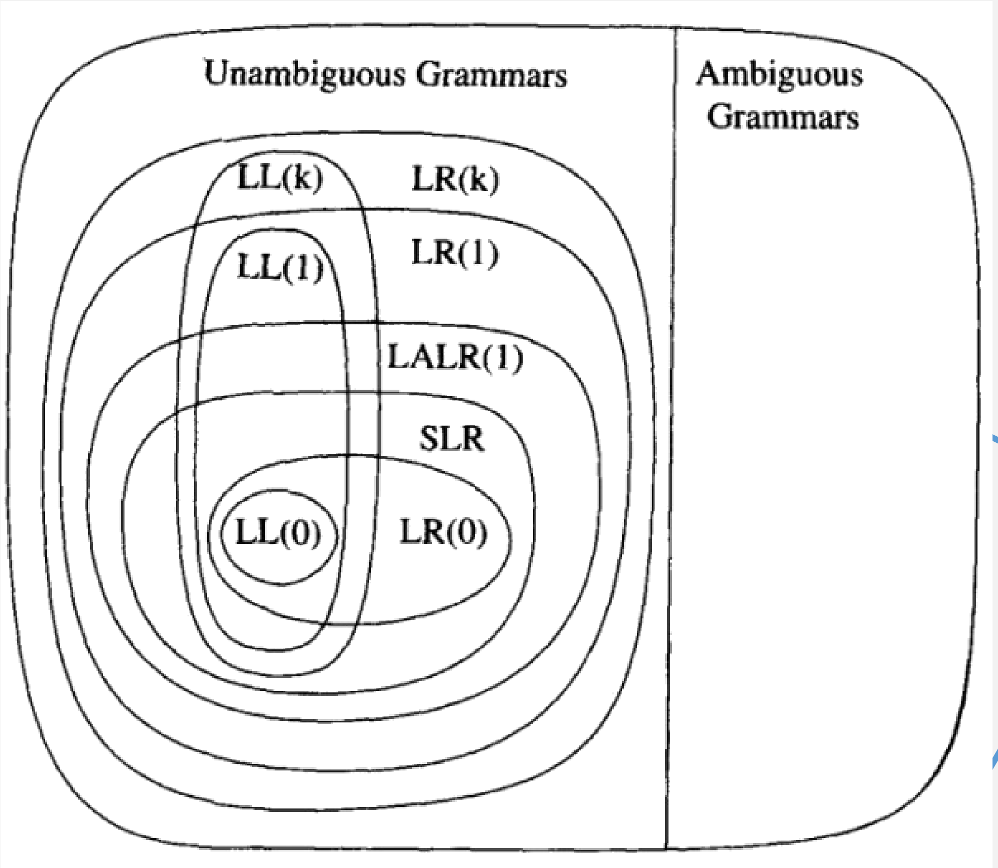


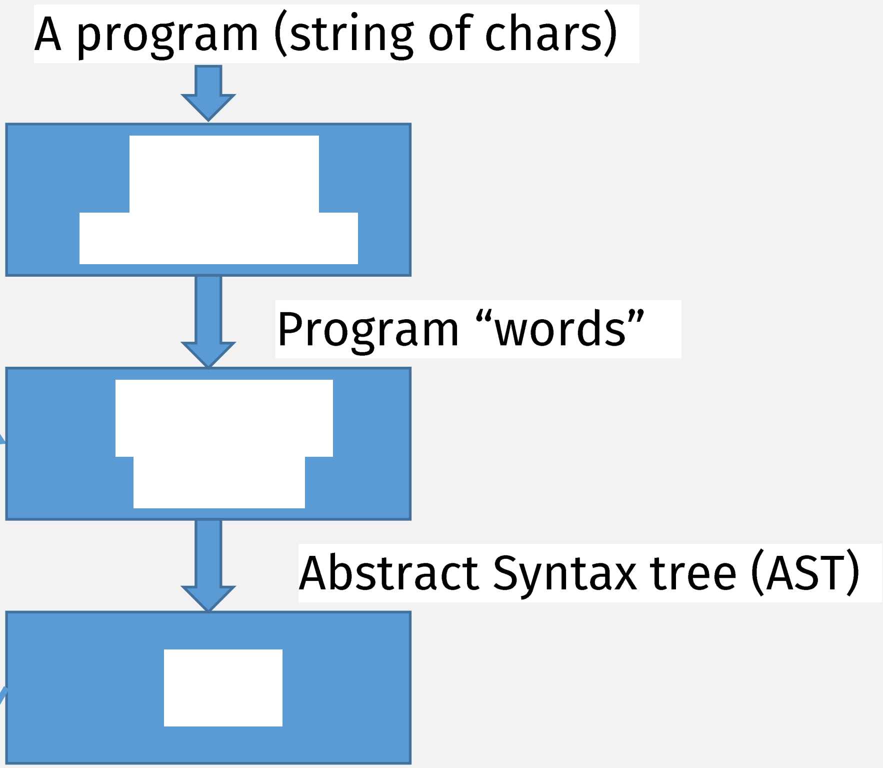
2) choose "<u>look ahead</u>" amount

2 parser design decisions:
1) Parse from <u>left</u>, or from <u>right</u>

All CFLS

To learn more, take a Compilers Class!





This phase needs computation that goes beyond CFLs

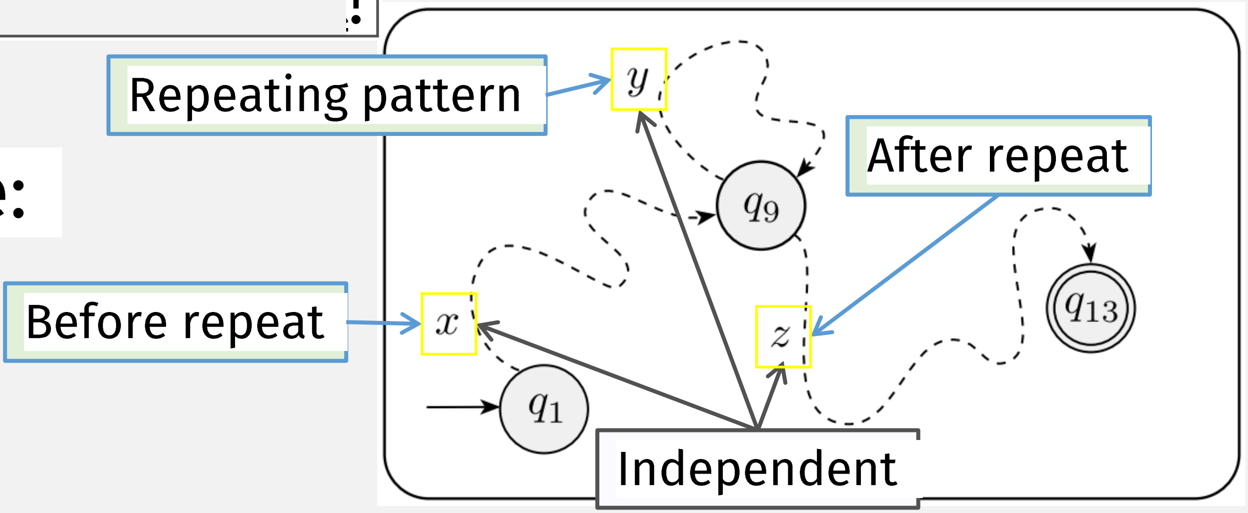
Flashback: Pumping Lemma for Regular Langs

- **Pumping Lemma** describes how strings repeat

- A non-regular language:

$$\{0^n 1^n \mid n \geq 0\}$$

Kleene star can't express this pattern:
2nd part depends on (length of) 1st part



- Q: How do CFLs repeat?

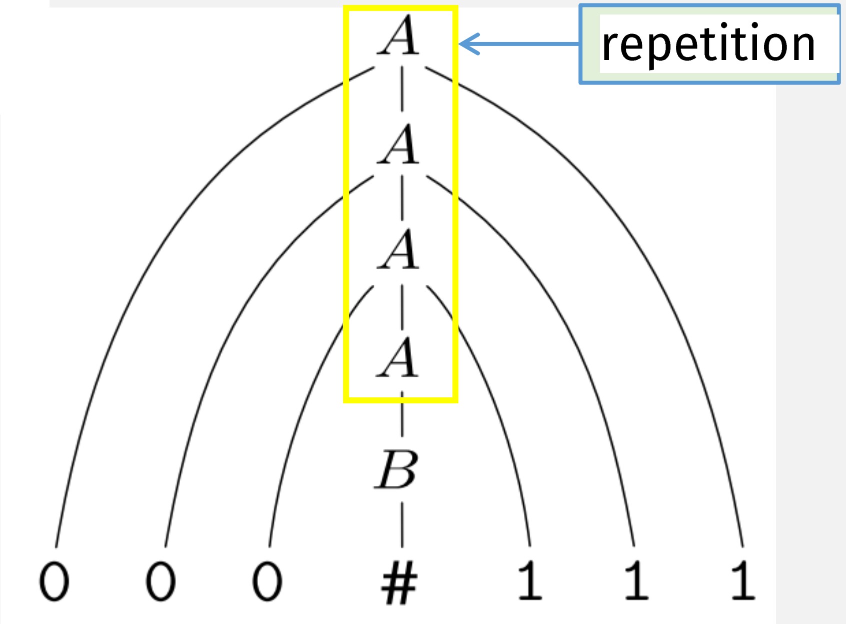
Repetition and Dependency in CFLs

Parts before/after repetition point <u>linked</u> (not independent)

Repetition

$$A \to 0A1$$
$$A \to B$$
$$B \to \#$$

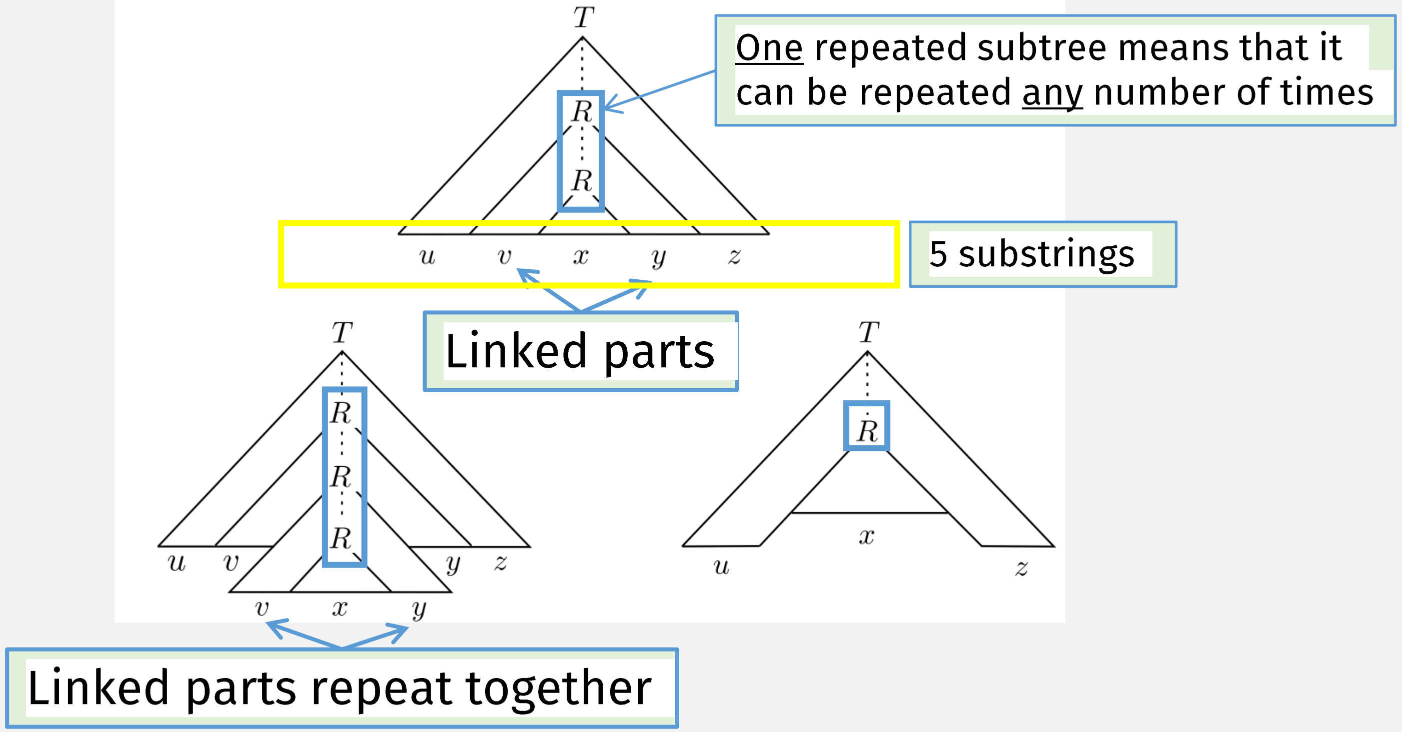
$$\{0^n\#1^n \mid n \geq 0\}$$



$$A \Rightarrow 0A1 \Rightarrow 00A11 \Rightarrow 000A111 \Rightarrow 000B111 \Rightarrow 000\#111$$

How Do Strings in CFLs Repeat?

- Strings in CFLs repeat subtrees in the parse tree

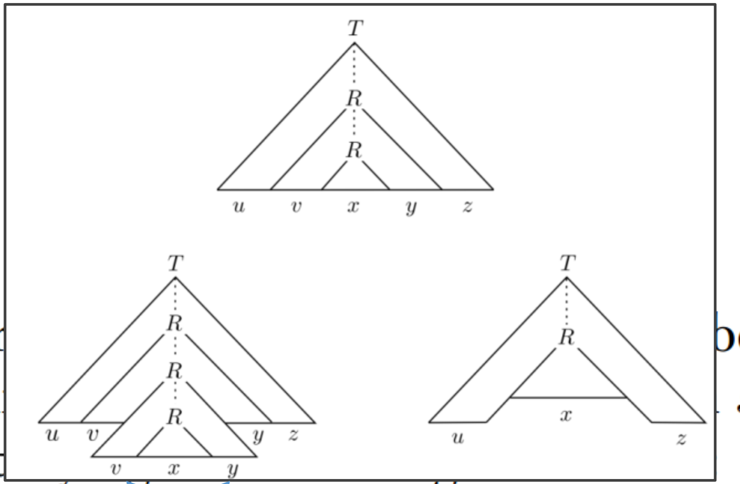


Pumping Lemma for CFLS

Pumping lemma for context-free languages If A is a context-free language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided into five pieces $s = uvxyz$ satisfying the conditions

Two pumpable parts.
But they must be pumped together!

1. for each $i \geq 0$, $uv^ixy^iz \in A$,
2. $|vy| > 0$, and
3. $|vxy| \leq p$.



Pumping lemma If A is a regular ... ber p (the pumping length) where if s is any stri... s may be divided into three pieces, $s = xyz$, sat...

1. for each $i \geq 0$, $xy^iz \in A$,
2. $|y| > 0$, and
3. $|xy| \leq p$.

One pumpable part

Two pumpable parts, pumped together

Previously

A Non CFL example

language $B = \{\mathtt{a}^n\mathtt{b}^n\mathtt{c}^n \mid n \geq 0\}$ is not context free

Intuition

- **Strings in CFLs** can have **two parts** that are **“pumped” together**
- **Language** B **requires three parts** to be **“pumped” together**
- So **it’s not a CFL!**

Proof?

Want to prove: $a^n b^n c^n$ **is not a CFL**

Pumping lemma for context-free languages If A is a context-free language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided into five pieces $s = uvxyz$ satisfying the conditions

1. for each $i \geq 0$, $uv^ixy^iz \in A$,
2. $|vy| > 0$, and
3. $|vxy| \leq p$.

Reminder: CFL Pumping lemma says:
all strings $a^n b^n c^n \geq$ length p are splittable into $uvxyz$ where v and y are pumpable

Proof (by contradiction):

Now we must find a contradiction …

- Assume: $a^n b^n c^n$ **is a CFL**
  - So it must satisfy the pumping lemma for CFLs
  - I.e., all strings $\geq$ length p are pumpable
- Counterexample = $\boxed{a^p b^p c^p}$

Contradiction if:
- A string in the language ☑
- $\geq$ length p ☑
- Is **not splittable** into $uvxyz$ where v and y are pumpable

???

$$\overbrace{a \ldots}^{p\ \text{as}}\ \overbrace{b \ldots}^{p\ \text{bs}}\ \overbrace{c \ldots}^{p\ \text{bs}}$$

Choose x split so y contains:
all 0s

Pumping y: produces a string with more 0s than 1s
Which is not in the language 0
This means that 0 does not satisfy the pumping lemma
Which means that that 0 is a not regular language
This is a **contradiction** of the assumption!

Want to prove: $a^n b^n c^n$ **is not a CFL**

Pumping lemma for context-free languages If A is a context-free language, then there is a number p (the pumping length) where, if s is any string in A of length at least p, then s may be divided into five pieces $s = uvxyz$ satisfying the conditions

1. for each $i \geq 0$, $uv^i xy^i z \in A$,
2. $|vy| > 0$, and
3. $|vxy| \leq p$.

Possible Splits

Proof (by contradiction):

contradiction

- Assume: $a^n b^n c^n$ **is a CFL**
  - So it must satisfy the pumping lemma for CFLs
  - I.e., all strings $\geq$ length p are pumpable
- Counterexample = $a^p b^p c^p$

Contradiction if:
- A string in the language
- $\geq$ length p
- **Is not splittable** into $uvxyz$ where v and y are pumpable

- Possible Splits (using condition # 3: $|vxy| \leq p$)

Not pumpable

  - ☒ vxy is all as
  - ☒ vxy is all bs
  - ☒ vxy is all cs
  - ☒ vxy has as and bs
  - ☒ vxy has bs and cs
  - (vxy cannot have as, bs, and cs)

$$\underbrace{a \ldots}_{p \text{ as}}\ \underbrace{b \ldots}_{p \text{ bs}}\ \underbrace{c \ldots}_{p \text{ bs}}$$

vxy **???**

$a^p b^p c^p$ **cannot be split into** $uvxyz$ where v and y are pumpable

So $a^n b^n c^n$ **is not a CFL**
(justification:
contrapositive of CFL pumping lemma)

ot satisfy the pumping lemma
is a not regular language
he assumption!

Another Non-CFL $D = \{ww \mid w \in \{0,1\}^*\}$

Be careful when choosing **counterexample** s: $\boxed{0^p10^p1}$

This s **can be pumped according to CFL pumping lemma:**

$$\overbrace{\underbrace{000\cdots000}_{u}\;\underbrace{0}_{v}\;\underbrace{1}_{x}}^{0^p1}\;\overbrace{\underbrace{0}_{y}\;\underbrace{000\cdots0001}_{z}}^{0^p1}$$

Pumping v and y (together) produces string still in D!

- **CFL Pumping Lemma conditions:**
  - ☑ **1.** for each $i \geq 0$, $uv^ixy^iz \in A$,
  - ☑ **2.** $|vy| > 0$, and
  - ☑ **3.** $|vxy| \leq p$.

So **this attempt** to **prove** that the **language** is **not a CFL failed**.
(It doesn't prove that the language is a CFL!)

Another Non-CFL $D = \{ww \mid w \in \{0,1\}^*\}$

- Need <u>another counterexample</u> string s:

If vyx is contained in first or second half, then any pumping will break the match

$0^p 1^p 0^p 1^p$

So vyx must straddle the middle

But any pumping still breaks the match because order is wrong

- CFL Pumping Lemma conditions:

1. for each $i \geq 0$, $uv^ixy^iz \in A$,

2. $|vy| > 0$, and

3. $|vxy| \leq p$.

Now we have proven that this language is **not a CFL!**

A Practical Non-CFL

- **XML**
  - ELEMENT → <TAG>CONTENT</TAG>
  - Where TAG is any string

- XML also looks like this non-CFL: $D = \{ww|\ w \in \{0,1\}^*\}$

- This means XML is not context-free!
  - Note: HTML *is* context-free because …
  - … there are only a finite number of tags,
  - so they can be embedded into a finite number of rules.

In practice:

- XML is parsed as a CFL, with a CFG
- Then matching tags checked in a 2^{nd} pass with a more powerful machine …

Next: A More Powerful Machine …

M_1 accepts its input if it is in language: $B = \{w\#w \mid w \in \{0,1\}^*\}$

M_1 = "On input string w:

1. Zig-zag across the tape to corresponding positions on either side of the # symbol to check whether these positions contain the same symbol. If they do not, or if no # is found, *reject*. Cross off symbols as they are checked to keep track of which symbols correspond.

Infinite memory (initial contents are the **input string**)

Can move to, and **read/write from** <u>**arbitrary**</u> **memory locations!**