CSE443
Compilers

Dr. Carl Alphonce
alphonce@buffalo.edu
343 Davis Hall
On-line delivery

Guiding principles

- maintain integrity of course learning outcomes
- use UB sanctioned tools/services for content delivery/interactions
- be judicious in technology use to maintain reliability
On-line delivery

- Project plan
  - distributed software development
  - really try to use git, trello/zenhub to manage your project
  - e-mail, Webex, Piazza, etc for communication

- Team meeting plan
  - Current plan: Webex at regularly scheduled time
  - Webex need not involve concurrent video stream for all participants
On-line delivery

- Lectures
  - Slides/video/transcript posted ahead of time
  - During class (MWF @ 10): live Q&A session (technology TBA - e.g. could be Webex, could be Piazza)

- Exam
  - Either on-line or take-home, during regular final exam slot.
On-line delivery

- My concerns
  - Internet service @ all students
  - Technology problems (webcam issues, etc)

- Your concerns?
  - Office hours - handling load?
Phases of a compiler

Intermediate Representation (IR): specification and generation

Figure 1.6, page 5 of text
Project notes
Helpful Links

**MIDRULE ACTIONS**
- Bison manual: Using mid-rule actions
- Bison manual: How mid-rule actions are translated

**ERROR HANDLING**
- Bison manual: error reporting
- Bison manual: error recovery
- Article and sample code from IBM showing error handing
Helpful Links

- Type look-up for primitive types?
- Marker non-terminal rules
- %union for type checking
Intermediate Representations
Our language
(use name equivalence)

- pre-defined types:
  - primitive types: integer, real, Boolean, character
  - composite type: string
- user-defined types:
  - record types have names
    - type rec : [ real : x , y ]
  - array types have names
    - type arr : 2 -> string
  - function types have names
    - type fun : ( real : x ) -> rec
Recursive records

Recursive functions

A record type must allow a component to be of the same type as the type itself:

\[ \text{type Node: [ integer datum:=} 0 ; \text{ Node rest:=} \text{null } \text{ ] } \]
type information

- type indicates size
- type indicates storage location
  - primitives: either stack or heap
  - records: on heap (via pointer)
  - arrays: on heap (via pointer)
  - functions: code in static, locals on stack
- need to determine how to lay out records, arrays, invocation records in memory
Sizes of types

- int: 32 bits (2's complement)
- real: 64 bits (IEEE 754)
- Boolean: 8 bits (TBD)
- character: 8 bit (ASCII)
Sizes of types

- type string: 1 -> character
  - 4 bytes + length of string * size of character (= 1 byte)
- # of dimensions is part of type

<table>
<thead>
<tr>
<th>size of dimension 1 (integer)</th>
<th>(0)</th>
<th>(1)</th>
<th>(2)</th>
<th>(3)</th>
<th>(4)</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>S</td>
<td>V</td>
<td>AX</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

https://en.wikipedia.org/wiki/VAX