CSE443
Compilers

Dr. Carl Alphonce
alphonce@buffalo.edu
343 Davis Hall
Announcements

- Office hours will start at 12:45 today.
- HW3 will be released in a day or two.
- Weekly team meetings.
Announcements

- Office hours will start at 12:45 today.
- HW3 will be released in a day or two.
- Weekly team meetings with me.
Announcements

- Office hours will start at 12:45 today.
- HW3 will be released in a day or two.
- Weekly team meetings with me:
  - Have one (1) team member sign up on behalf of team: the person whose UBIT comes first alphabetically
  - Meetings will be no longer than 15 minutes
  - All team members must attend and be on time
  - Link to Doodle poll will be e-mailed
  - First meetings early next week
Project checkpoint

SECTION 1: Lexical structure
project 1: done

SECTION 2: Syntactic structure
project 2: done

SECTION 3: Type checking and semantics
project 3: checkpoint Wednesday 3/28

SECTION 4: Intermediate code generation
project 3: due Monday 4/9 @ 5:00 PM
Intermediate Representation (IR): specification and generation

Figure 1.6, page 5 of text
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:

type Node: [ integer datum:=0 ; Node rest:=null ]
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
- 8 bytes + length of string * size of character (= 1 byte)

<table>
<thead>
<tr>
<th># of dimensions</th>
<th>size of dimension 1</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>0</th>
<th>5</th>
<th>V</th>
<th>A</th>
<th>X</th>
<th>E</th>
<th>S</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>5</td>
<td>V</td>
<td>A</td>
</tr>
</tbody>
</table>

https://en.wikipedia.org/wiki/VAX
Sizes of types

Q: Since the number of dimensions is part of the type, do we need to store it at runtime?
A: No.

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