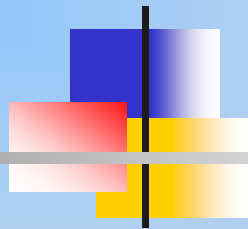


# CSE302: Compiler Design



Instructor: Dr. Liang Cheng  
Department of Computer Science and Engineering  
P.C. Rossin College of Engineering & Applied Science  
Lehigh University

April 26, 2007



# Outline

---

- Recap of 3-address code generation
  - Array references
- Review and feedback



# SDT of Array Type Declaration

- $D \rightarrow T \text{ id}; \{\text{top.put}(\text{id.lexeme}, T.\text{type}, T.\text{width});\}$
- $T \rightarrow B \{t = B.\text{type}; w = B.\text{width};\} C \{$   
     $T.\text{type} = C.\text{type}; T.\text{width} = C.\text{width};$   
     $\}$
- $B \rightarrow \mathbf{int} \{B.\text{type} = \text{integer}; B.\text{width} = 4;\}$
- $B \rightarrow \mathbf{float} \{B.\text{type} = \text{float}; B.\text{width} = 8;\}$
- $C \rightarrow [\mathbf{num}] C1 \{$   
     $C.\text{type} = \text{array}(\mathbf{num}.\text{value}, C1.\text{type});$   
     $C.\text{width} = \mathbf{num}.\text{value} \times C1.\text{width};$   
     $\}$
- $C \rightarrow \varepsilon \{C.\text{type} = t; C.\text{width} = w;\}$



# Translation of Array References

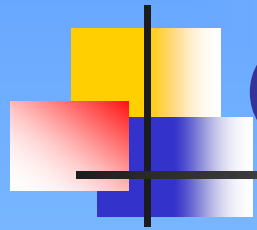
- $L \rightarrow \mathbf{id} [ E ]$ 
  - $L.array = top.get(id.lexeme)$
  - $L.type = L.array.type.elem$
  - $L.addr = \mathbf{new} Temp()$
  - $gen(L.addr '=' E.addr '*' L.type.width)$
- $L \rightarrow L1 [ E ]$ 
  - $L.array = L1.array$
  - $L.type = L1.type.elem$
  - $t = \mathbf{new} Temp()$
  - $L.addr = \mathbf{new} Temp()$
  - $gen(t '=' E.addr '*' L.type.width)$
  - $gen(L.addr '=' L1.addr '+' t)$
- $S \rightarrow \mathbf{id} = E ;$ 
  - $S.code = E.code || gen(top.get(id.lexeme) '=' E.addr)$
- $E \rightarrow L$ 
  - $E.addr = \mathbf{new} Temp()$
  - $gen(E.addr '=' L.array.base '[' L.addr ']')$
- $S \rightarrow L = E ;$ 
  - $gen(L.array.base '[' L.addr '] '=' E.addr$



# Outline

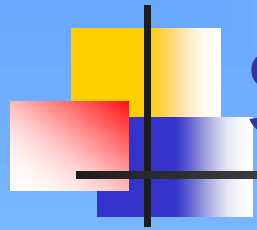
---

- Recap of 3-address code generation
- **Review and feedback**



# Course Evaluation

---



# Suggestion and Comments?

---