CSEN 1003: Compilers

Tutorial 11 - Intermediate Code Generation

Today's Plan

1. Intermediate Representations
2. Translating to Intermediate Code
3. Recap

Intermediate Code

- We will look into two types of intermediate representations: Expression DAGs and 3-address code.

Intermediate Code

- We will look into two types of intermediate representations: Expression DAGs and 3-address code.
- The DAG can be more useful to a compiler than a parse tree because it can guide the representation of efficient code.

Intermediate Code

- We will look into two types of intermediate representations: Expression DAGs and 3-address code.
- The DAG can be more useful to a compiler than a parse tree because it can guide the representation of efficient code.
- The nodes of the DAG are operators and the leaves are atoms.

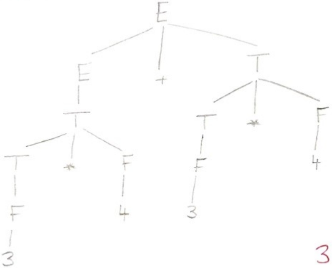
Intermediate Code

- We will look into two types of intermediate representations: Expression DAGs and 3-address code.
- The DAG can be more useful to a compiler than a parse tree because it can guide the representation of efficient code.
- The nodes of the DAG are operators and the leaves are atoms.
- 3-address code is a linearized version of the expression DAG where:
  - Each instruction has at most three addresses.
  - An address is an ID, a constant, or a compiler generated temporary variable.
  - Each instruction is a jump, has one operator, or two operators one of them is an assignment operator (=).

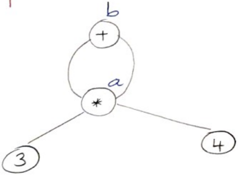
Parse Trees, Expression DAGs, and 3-address Code

(3 * 4) + (3 * 4)

Parse Tree



Expression DAG:



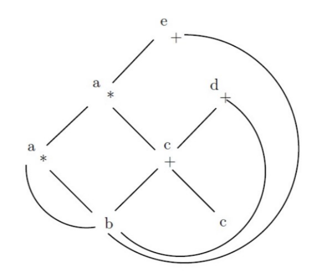
3-address code

a = 3 * 4

b = a + a

Exercise 10-1

Example



$$c = b + c$$
$$a = b * b$$
$$d = c + b$$
$$a = a * c$$
$$e = a + b$$

Today's Plan

1. Intermediate Representations
2. Translating to Intermediate Code
3. Recap

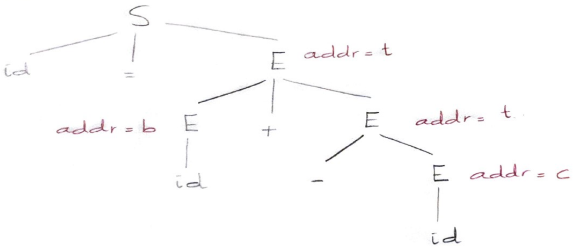
Translating Simple Arithmetic Expressions

$$
\begin{array}{lll}
S \rightarrow \textbf{id} = E\ ; & \{\ gen(\ top.get(\textbf{id}.lexeme)\ '='\ E.addr);\ \} \\
E \rightarrow E_1 + E_2 & \{\ E.addr = \textbf{new}\ Temp\,(); \\
 & \quad gen(E.addr\ '='\ E_1.addr\ '+'\ E_2.addr);\ \} \\
\quad \mid\ -E_1 & \{\ E.addr = \textbf{new}\ Temp\,(); \\
 & \quad gen(E.addr\ '='\ '\textbf{minus}'\ E_1.addr);\ \} \\
\quad \mid\ (\ E_1\ ) & \{\ E.addr = E_1.addr;\ \} \\
\quad \mid\ \textbf{id} & \{\ E.addr = top.get(\textbf{id}.lexeme);\ \}
\end{array}
$$

Exercise 11-3

Example

a = b + - c



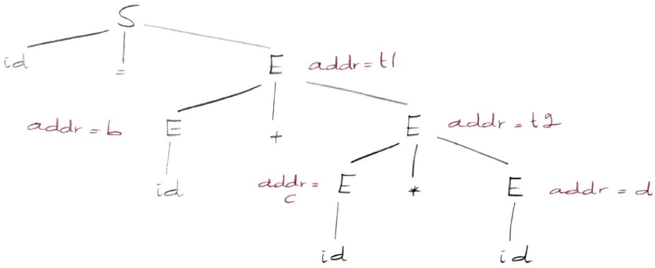
t2 = minus c

t1 = b + t2

a = t1

Example

a = b + (c * d)



t2 = c * d

t1 = b + t2

a = t1

Translating Array References

- Array variables have an offset attribute in the symbol table indicating the base address of the array.
- To access a particular entry, we need to calculate its relative address.

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Row major:

21	21	21	21	21	21	...

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Row major: | 21 | 21 | 21 | 21 | 21 | 21 | ...

$A[4] = 0 + (4 \times 21 \times 4) = 336$

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Row major:

21	21	21	21	21	21	...

$A[4] = 0 + (4 \times 21 \times 4) = 336$

$A[4, 5] = 336 + (5 \times 4) = 356$

Column major:

11	11	11	11	11	11	...

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Row major: | 21 | 21 | 21 | 21 | 21 | 21 | ...

$A[4] = 0 + (4 \times 21 \times 4) = 336$
$A[4, 5] = 336 + (5 \times 4) = 356$

Column major: | 11 | 11 | 11 | 11 | 11 | 11 | ...

$A[5] = 0 + (5 \times 11 \times 4) = 220$

Exercise 11-5

Example

An integer array $A[i, j]$ has index i ranging from 0 to 10 and index j ranging from 0 to 20. Integers take 4 bytes each. Suppose array A is sorted starting at byte 0. Find the location of $A[4, 5]$ assuming row major and column major order.

Row major:

21	21	21	21	21	21	...

$A[4] = 0 + (4 \times 21 \times 4) = 336$
$A[4, 5] = 336 + (5 \times 4) = 356$

Column major:

11	11	11	11	11	11	...

$A[5] = 0 + (5 \times 11 \times 4) = 220$
$A[4, 5] = 220 + (4 \times 4) = 236$

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$

$A[3, 4] = 1320 + (4 \times 11 \times 8) = 1672$

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$
$A[3, 4] = 1320 + (4 \times 11 \times 8) = 1672$
$A[3, 4, 5] = 1672 + (5 \times 8) = 1712$

Column major:

5×5	5×5	5×5	5×5	5×5	5×5	...

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$
$A[3, 4] = 1320 + (4 \times 11 \times 8) = 1672$
$A[3, 4, 5] = 1672 + (5 \times 8) = 1712$

Column major:

5×5	5×5	5×5	5×5	5×5	5×5	...

$A[5] = 0 + (5 \times 5 \times 5 \times 8)$

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$
$A[3, 4] = 1320 + (4 \times 11 \times 8) = 1672$
$A[3, 4, 5] = 1672 + (5 \times 8) = 1712$

Column major:

5×5	5×5	5×5	5×5	5×5	5×5	...

$A[5] = 0 + (5 \times 5 \times 5 \times 8)$
$A[4, 5] = A[5] + (4 \times 5 \times 8)$

Exercise 11-7

Example

A real array $A[i, j, k]$ has index i ranging from 0 to 4 and index j ranging from 0 to 4, and index k ranging from 0 to 10. Reals take 8 bytes each.Find the location of $A[3, 4, 5]$.

Row major:

5×11	5×11	5×11	5×11	5×11	5×11	...

$A[3] = 0 + (3 \times 5 \times 11 \times 8) = 1320$
$A[3, 4] = 1320 + (4 \times 11 \times 8) = 1672$
$A[3, 4, 5] = 1672 + (5 \times 8) = 1712$

Column major:

5×5	5×5	5×5	5×5	5×5	5×5	...

$A[5] = 0 + (5 \times 5 \times 5 \times 8)$
$A[4, 5] = A[5] + (4 \times 5 \times 8)$
$A[3, 4, 5] = A[4, 5] + (3 \times 8)$

Exercise 11-4

$$
\begin{array}{lcll}
S & \rightarrow & \textbf{id} = E \; ; & \{\ gen(\ top.get(\textbf{id}.lexeme)\ '='\ E.addr);\ \} \\
& | & L = E \; ; & \{\ gen(L.array.base\ '['\ L.addr\ ']'\ '='\ E.addr);\ \} \\
E & \rightarrow & E_1 + E_2 & \{\ E.addr = \textbf{new}\ Temp\,(); \\
& & & \quad gen(E.addr\ '='\ E_1.addr\ '+'\ E_2.addr);\ \} \\
& | & \textbf{id} & \{\ E.addr = top.get(\textbf{id}.lexeme);\ \} \\
& | & L & \{\ E.addr = \textbf{new}\ Temp\,(); \\
& & & \quad gen(E.addr\ '='\ L.array.base\ '['\ L.addr\ ']');\ \} \\
L & \rightarrow & \textbf{id}\ [\ E\ ] & \{\ L.array = top.get(\textbf{id}.lexeme); \\
& & & \quad L.type = L.array.type.elem; \\
& & & \quad L.addr = \textbf{new}\ Temp\,(); \\
& & & \quad gen(L.addr\ '='\ E.addr\ '*'\ L.type.width);\ \} \\
& | & L_1\ [\ E\ ] & \{\ L.array = L_1.array; \\
& & & \quad L.type = L_1.type.elem; \\
& & & \quad t = \textbf{new}\ Temp\,(); \\
& & & \quad L.addr = \textbf{new}\ Temp\,(); \\
& & & \quad gen(t\ '='\ E.addr\ '*'\ L.type.width); \\
& & & \quad gen(L.addr\ '='\ L_1.addr\ '+'\ t);\ \}
\end{array}
$$

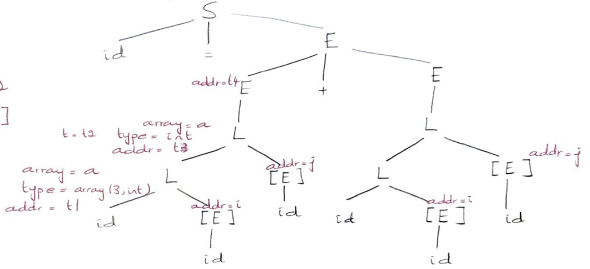
Exercise 11-4

Example

A is a 2X3 array of ints. type = array (2, array (3, int)) width = 20

B is a 4X5 array of ints. type = array (4, array (5, int)) width = 80

x = a[i][j] + b[i][j]



t1 = i * 12
t2 = j * 4
t3 = t1 + t2
t4 = a [t3]
t5 = i * 20
t6 = j * 4
t7 = t5 + t6
t8 = b [t7]
t9 = t4 + t8
x = t9

Exercise 11-7

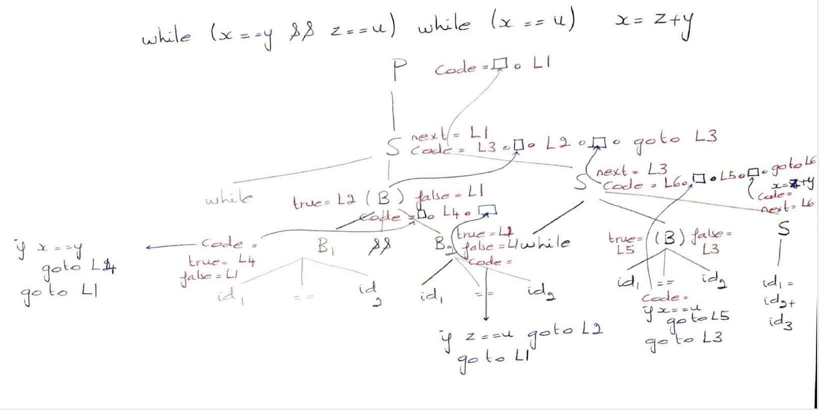
$$
\begin{array}{lllll}
P & \longrightarrow & S & & S.next = newlabel() \\
 & & & P.code & = S.code \circ label(S.next) \\[1em]
S & \longrightarrow & \mathbf{id}_1 \mathbf{=} \mathbf{id}_2 + \mathbf{id}_3 & S.code & = gen(\mathbf{id}_1.addr\ '='\ \mathbf{id}_2.addr\ '+'\ \mathbf{id}_3.addr) \\[1em]
S & \longrightarrow & \mathbf{while}\ (B)\ S_1 & & B.true = newlabel(); B.false = S.next \\
 & & & & S_1.next = newlabel() \\
 & & & S.code & = label(S1.next)\ \circ\ B.code \\
 & & & & \circ\ label(B.true)\ \circ\ S_1.code \\
 & & & & \circ\ gen('\texttt{goto}'\ S_1.next) \\[1em]
B & \longrightarrow & B_1\ \&\&\ B_2 & & B_1.true = newlabel(); B_1.false = B.false; \\
 & & & & B_2.true = B.true; B_2.false = B.false; \\
 & & & B.code & = B_1.code \circ label(B_1.true) \circ B_2.code \\[1em]
B & \longrightarrow & \mathbf{id}_1 == \mathbf{id}_2 & B.code & = gen('\texttt{if}'\ \mathbf{id}_1.addr\ '=='\ \mathbf{id}_2.addr\ '\texttt{goto}'\ B.true) \\
 & & & & \circ\ gen('\texttt{goto}'\ B.false)
\end{array}
$$

Give the value of `P.code` as a result of parsing the string:

$$\mathit{while}(x == y\ \ \&\&\ \ z == u)\ \ \mathit{while}\ \ (x == u)\ \ x = z + y$$

Exercise 11-7

Example



Exercise 11-7

Example

```
L3: if x == y goto L4
    goto L1
L4: if z == u goto L2
    goto L1
L2: L6: if x == u goto L5
        goto L3
L5: x = z + y
    goto L6
    goto L3
L1:
```

Today's Plan

1. Intermediate Representations
2. Translating to Intermediate Code
3. Recap

Covered Topics

1. Intermediate Code
2. Translating to Intermediate Code.