The following is from LLVM code improvement, even something as simple as the GCD program can

Much of the decision making is driven by

Here I've used “virtual registers” for all computed values.

One family of IFs deserves separate mention:

Note that this differs slightly from the version shown in the Chapter 1 example from the book: Heron's formula:

Reprise compiler phases:

Code generation (intermediate and

<table>
<thead>
<tr>
<th>Index</th>
<th>Symbol</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>4</td>
<td></td>
</tr>
</tbody>
</table>

int main() {
    int i, j, a, b, c, s;

    i := v11 := v12 := v13 :=

    while (j = i > j)
        j := v12 := v13 :=
            j := v14 :=

    if less than

    a := r1 := r2 :=

    { call
        a := r1 := r2 :=

    }

    { call
        a := r1 := r2 :=

    }

    a := r1 := r2 :=

    { return
        a := r1 := r2 :=

    }

    return

End

Middle end

...