King Saud University
College of Computer and Information Sciences
Computer Science Department

CSC 340: Programming Language and Compilation
Exercises: Three Address Code and Code Generation

----

1. **Exercises 8.2.1 (page 516) [1]**

   Generate code for the following three-address statements assuming all variables are stored in memory locations.

   a) x = 1
   b) x = a
   c) x = a + 1
   d) x = a + b
   e) The two statements

      ○ x = b * c
      ○ y = a + x
      ○

2. **Exercise 8.2.3 (Page 517) [1]**

   Generate code for the following three-address sequence assuming that p and q are in memory locations:

```
y = *q
q = q + 4
*p = y
p = p + 4
```

3. **Exercise 8.2.4 (Page 517) [1]**

   Generate code for the following sequence assuming that x, y, and z are in memory locations:

```
if x < y goto L1
z = 0
goto L2

L1: z = 1
```

----

4. Exercise 8.4.1 (Page 531) [1]
Figure 8.10 is a simple matrix-multiplication program.

- **a) Translate the program into three-address statements of the type we have been using in this section. Assume the matrix entries are numbers that require 8 bytes, and that matrices are stored in row-major order.**
- **b) Construct the flow graph for your code from (a).**
- **c) Identify the loops in your flow graph from (b).**

```
for (i=O; i<n; i++)
   for (j=O; j<n; j++)
      c[i][j] = 0.0;
for (i=O; i<n; i++)
   for (j=O; j<n; j++)
      for (k=O; k<n; k++)
         ￼c[i][j] = c[i][j] + a[i][k]*b[k][j];
```

Figure 8.10: A matrix-multiplication algorithm

[1] **Book:** “Compilers Principles, techniques, & tools”, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffry D. Ullman