

CS 3723, Spring 2002, Selected Answers

1. Only Java normally supports garbage collection, to recover storage that was dynamically allocated, but no longer has any active pointer (or reference) to it. In C and C++ one has to explicitly free or delete such storage.

3. Here is a complete program:

```
/* Evaluate.java: parser with exam question grammar:
 *   P  ----> E
 *   E  ----> dig | '(' E E Op ')'
 *   Op ----> '+' | '-' | '*' | '/' | '^'
 *   dig ----> '0' | '1' | '2' | '3' | '4' | '5' | '6' | '7' | '8' | '9'
 */
import java.io.*;

public class Evaluate {
    private char next; // = ' ';

    // P: recognize a program
    public int P() {
        return E();
    }

    // E: recognize a statement
    private int E() {
        scan();
        int res = 0, res1, res2;
        if (Character.isDigit(next)) {
            res = next - '0';
        }
        else if (next == '(') {
            res1 = E();
            res2 = E();
            scan();
            res = eval(res1, res2, next);
            scan(); // for the ')'
            if (next != ')') error(1);
        }
        else error(3);
        return res;
    }

    private int eval(int r1, int r2, char op) {
        switch (op) {
            case '+': return r1 + r2;
            case '-': return r1 - r2;
            case '*': return r1 * r2;
            case '/': return r1 / r2;
            case '^': int res = r1;
                       while (r2 != 1) {
```

```
                res = res*r1; r2--;
            }
            return res;
        default:
            error(4);
            return 0;
    }
}

private void scan() {
    do {
        try {
            next = (char)System.in.read();
        } catch (IOException e){
            error(2);
        }
    } while ( next == ' ' || next == '\n');
}

// error: error encountered during the parse
private void error(int n) {
    System.out.println("ERROR " + n + ", Token: " + next );
    System.exit(1);
}

public static void main(String[] args) {
    Evaluate evaluate = new Evaluate();
    System.out.println(evaluate.P());
}
}
```

7. Here are the new rules:

```
years(N, H, Y) :- reigns(N, A, B), house_of(N, H), Y >= A, Y =< B.

rulers_of_house(H) :- reigns(N, A, B), house_of(N, H),
    write(N), write(', from '), write(A),
    write(' to '), write(B), nl, fail.
```