C. Exercises on Compilers and Programming Languages

This is a preparatory exercise for the course. As such none of the tasks is mandatory and there is no hand-in due next week.

Objectives

- To recall basics of lexical and syntactic language specifications, syntactic categories
- To recall basics of type checking

I estimate about 4 hours to complete all these tasks is sufficient, plus possible extra reading time, if you forgot the details of regular expressions, grammars, etc.

Task 1. Explain what are the languages described by the following regular expressions:

A. $(ab)^*$

B. $1(0|1)^*$

C. $(\,(a(d|c)e)\_)^+$

Task 2. Write regular expressions describing the following languages:

A. A language of comma separated words; each word consisting of one or more occurences of the letter 'a'.

B. A language containing four words; each word is one of the keywords: if, then, else, while

C. White space consisting of spaces (␣), tab characters(\t), and new line characters (\n).

Task 3. Write a regular expression specifying identifiers as in the following quote from the ISO C standard: *An identifier is a sequence of letters and digits; the first character must be a letter. The underscore _ counts as a letter. Upper- and lowercase letters are different.*

Task 4. Explain in English what is the language described by this grammar:

$$
\begin{aligned}
S &\rightarrow T\ U\ \text{a}\ V \\
T &\rightarrow \text{John} \mid \text{Mary} \mid \text{Alice} \\
U &\rightarrow \text{reads} \mid \text{writes} \\
V &\rightarrow \text{book} \mid \text{letter} \mid \text{poem}
\end{aligned}
$$

Task 5. Write a grammar representing the language of balanced parentheses of three kinds, so "(", "{", and "[", where they can be arbitrarily nested as long as they are always balanced with a closing parenthesis of the same kind.

Task 6. Draw an abstract syntax tree (guess!) for the following Java expression and annotate nodes with types.

```
("3" + "4").length()*5
```

Task 7. Consider the following five example programs. Which part of the compiler will report an error in each of the programs ? (1) lexer/scanner (2) parser (3) name and type analysis? Or is the program statically correct?

A.
```
class Main {
    int main() {
        return x;
    }
}
```

B.
```
class Main {
    int main() {
        int x = 0;
        if (x == 0) {
            String x = "1";
            return x;
        }
        return 1;
    }
}
```

C.
```
class Main {
    void f() {
        int 1x = 0;
    }
}
```

D.
```
class Main {
    int main() {
        int x = 0;
        if (x == 0) {
            String y = "1";
            return x;
        }
        return 1;
    }
}
```

E.
```
class Main {
    int main() {
        if (x == 0) {
            String y = "1";
            return 0;
        }
    }
```

Task 8. Indicate an expression, a statement and a declaration in the program D of Task 7.