Formal languages and compilers – TEST 1 – a.y. 2010-2011

Exercise 1 [marks: 12]

Provide the minimum DFA to recognize the language generated by the following regular expression:

$$(0^* \mid 1^* \mid (01)^*)$$

Exercise 2 [marks: 8]

Let r_1 and r_2 be defined as

$$r_1 = (r \mid s)^*,$$
$$r_2 = (r^* \mid s^*),$$

where r and s stay for arbitrary regular expressions. Say whether or not r_1 and r_2 denote the same language. Justify your answer. È SUFFICIENTE UN CONTROESEMPIO

Exercise 3 [marks: 6]

Say whether or not the language

$$\{a^n b^n : n > 0\}$$

is a regular language. Justify your answer. PUMPING LEMMA [RISP: NO]

Exercise 4 [marks: 4]

Let

$$N = (S^N, \Sigma^N, move^N, s_0^N, F^N) \qquad \text{and} \qquad M = (S^M, \Sigma^M, move^M, s_0^M, F^M)$$

be two deterministic finite state automata. Define an automaton $(S, \Sigma, \text{move}, \text{s}_0, \text{F})$ that recognizes the language $\mathcal{L}(N) \cap \mathcal{L}(M)$.

4) $S = \bigcup_{i,j} \{(S_i^N \times S_j^M)\}$ $\quad \Sigma = \Sigma^N \cap \Sigma^M$ $\quad S_0 = (S_0^N \times S_0^M)$

$F = \bigcup_{i,j} \{ (S_i^N \times S_j^M) : S_i^N \in F^N , S_j^M \in F^M \}$

$move(S_i^N \times S_j^M, a) = S_{i'}^N \times S_{j'}^M$ se $move_N(S_i^N, a) = S_{i'}^N$

$move_M(S_j^M, a) = S_{j'}^M$

con $a \in \Sigma$

Formal languages and compilers – TEST 2 – a.y. 2010-2011

Exercise 1 [marks: 10]

Let $\mathcal{G}$ be the grammar with start symbol S, set of terminal symbols $\{\mathsf{id}, :\}$, and with the following productions:

$$\begin{array}{lll} S & \rightarrow & G \\ G & \rightarrow & P \mid PG \\ P & \rightarrow & \mathsf{id} : R \\ R & \rightarrow & \mathsf{id}\, R \mid \varepsilon \end{array}$$

1. Compute *first* and *follow*.
2. Define the grammar $\mathcal{G}'$ obtained by left factorizing $\mathcal{G}$, and say, justifying your answer, whether $\mathcal{G}'$ is LL(1) or not.

Exercise 2 [marks: 15]

Let $\mathcal{G}$ be the grammar with productions:

$$E \quad \rightarrow \quad \mathsf{not}\, E \mid E \Rightarrow E \mid (E) \mid \mathsf{id}$$

where E is the single non-terminal symbol.

1. Show that $\mathcal{G}$ is ambiguous.
2. Define a SRL grammar $\mathcal{G}'$ such that $\mathcal{L}(\mathcal{G}') = \mathcal{L}(\mathcal{G})$ and the ambiguity of $\mathcal{G}$ is resolved by imposing that:
   - the operator "$\Rightarrow$" is right associative;
   - the operator "not" has higher priority than "$\Rightarrow$".
3. Show the parsing table and the parse tree obtained when parsing the string $\mathsf{id} \Rightarrow \mathsf{not\,id} \Rightarrow \mathsf{id}$.

Exercise 3 [marks: 5]

Define a SRL grammar $\mathcal{G}$ that generates the language

$$\{a^m b^n c^n d^m \mid n, m > 0\}.$$

To show that $\mathcal{G}$ is actually SRL, provide its SRL parsing table, and the parse tree obtained when parsing the string *aaabbccddd*. FARE FOLLOW X REDUCE!

$S \rightarrow aSd \mid aRd$
$R \rightarrow bc \mid bRc$