Automi, Linguaggi e Traduttori. Laurea e Laurea on-line. 2005-06.
Docente: Alberto Pettorossi.
Programmi Java o C++ da sviluppare e documentare:
1. Membership of type 1 languages. 2. Parsing of type 2 languages.

Programma dettagliato del corso e domande d'esame (Tra parentesi quadre sono indicati i riferimenti bibliografici).

1. Turing Machines: basic notions [P3: Chapter 2, pages 1–4]. Decidability and semidecidability. Halting Problem and Post Correspondence Problem. [P3: Chapter 4. Sections 1 and 3] Chomsky Hierarchy and Kuroda Theorem [P2: pages 0.1, 0.2, 3] Recursiveness of context-sensitive languages. [P2: page 4]

2. Context-free languages, Chomsky normal form, and Greibach normal form. [P2: pages 5, 5.6–5.7]. Pushdown automata. Deterministic context-free languages and deterministic pushdown automata [P2: 5.22–5.29.1]. Decidability results for context-free languages and deterministic context-free languages [P3: Chapter 4, Section 3].

3. Avoiding useless symbols ('from below' and 'from above'), ε-productions, unit-productions, and left-recursion [P2: pages 5.1–5.6]. Pumping Lemma for context-free languages. Closure properties of context-free languages, ambiguity, and inherent ambiguity [P2: pages 5.19–5.20]. Self-embedding property [P2: pages 5.15–5.16]

4. Regular languages, Regular Expressions, Finite Automata. Kleene's Theorem: NFA=FA. [P2: pages FA 4.2, 4.3, 7, 8, 9.2, 10.2, 10.2.1, 10.2.2, 10.4]. Arden rule. Pumping Lemma for regular languages. Moore Theorem. Myhill-Nerode Theorem. Decidability results for regular languages. Closure properties of regular languages [P2: pages 6, FA 4.2, FA 10.5–10.6, FA 11]

5. Left-linear and right-linear finite automata [P2: pages 0.4.1, 0.4.2]

6. Backtracking and parsing for regular languages [P3: Chapter 7: last three pages. PC: Section 7.1]

7. Parsing for context free languages: Cocke-Younger-Kasami parser, Chop-Expand parser [P2: Chapter 5, pages 1–10, 5.30–5.32. PC: Section 7.2]. Earley parser (facoltativo) [P2: pages 5.39–5.44]

8. LL(1), LL(2), LR(0), SRL(1), LR(1), LALR(1) parsing [P2: pages 5.33–5.38] [PC: Sections 6.1] [D]. Operator precedence parsing (facoltativo). [DB: Sections 4.2, 4.3, 4.4, 4.5, 4.6]

9. String matching with finite automata and Knuth-Morris-Pratt algorithm [D]. Basic notions of program correctness; Hoare's triples (facoltativo) [P1: Chapter 8. Section 3].

Riferimenti bibliografici.
[P1] Pettorossi, A.: Quaderni di Informatica. Parte I, 2nd Edition, Aracne, 2004.
[P2] Pettorossi, A.: Quaderni di Informatica. Parte II, 2nd Edition, Aracne, 1993.
[P3] Pettorossi, A.: Theory of Computation. Part III, 2nd Edition, Aracne, 1994.
[PC] Pettorossi, A.: Programming in C++. Aracne, 2001.
[D] Dispense distribuite durante il corso e disponibili presso il docente.
[DB] A.V. Aho, R. Sethi, J.D. Ullman: Compilers: Principles, Techniques and Tools, Addison-Wesley 1986.

Per gli studenti on-line. Anziché seguire le dispense del docente, per i punti da 1 a 6, si dà facoltà allo studente on-line di studiare i capitoli: 1, 2, 3, 4, 5, 6 e le sezioni: 7.1, 7.2, 7.4.3, 7.4.4 del libro (in italiano): J. E. Hopcroft, R. Motwani, J.D. Ullman: "Automi, Linguaggi e Calcolabilità" Addison-Wesley, Pearson Education Italia, 1997. Si potranno omettere le sezioni: 4.2.2, 4.2.3 e 4.2.4. Le dimostrazioni dei teoremi sono facoltative. Lo studente on-line deve saper applicare gli algoritmi di cui punto 3, 4, 6 e 7 del programma. In particolare, occorre che egli sappia costruire: (i) l'automa finito deterministico equivalente a un dato automa finito nondeterministico, (ii) l'automa finito deterministico corrispondente a una data espressione regolare e (iii) l'espressione regolare corrispondente a un dato automa finito deterministico.
