CS-C2160 Theory of Computation, Spring 2022
Problem Set 11 (Lecture weeks 11 and 12)

Instructions

- Classroom Problems C11.1–C11.3 will be discussed and solved onsite at the tutorial sessions in lecture week 11. No credit is given for these problems.
- Homework Problems H11.1–H11.3 you should solve on your own, and be available to present your solutions at one of the tutorial sessions in lecture week 12. In order to get course credit, you need to indicate your solved problems on the signup sheet circulated at the beginning of the session.
- Supplementary Problems S11.1–S11.3 provide further illustration and extension of the course material, but will usually not be covered at the tutorials. You are however invited to work on these problems too, and discuss them with the course staff. Sample solutions are provided on MyCourses.

Classroom Problems

C11.1 Prove that the following problems are undecidable:

(i) Given a Turing machine M, does M recognise a regular language?

(ii) Given a Turing machine M, does M accept exactly one input string?

(iii) Given Turing machines M_1 and M_2, is $\mathcal{L}(M_1) \cap \mathcal{L}(M_2) = \emptyset$?

C11.2 Design an unrestricted grammar that generates the language

$$\{w \in \{a, b, c\}^* \mid w \text{ contains equally many } a\text{'s, } b\text{'s and } c\text{'s}\}.$$

C11.3 Which of the following claims are true and which are false?

(i) The computation of a deterministic Turing machine halts on every input.

(ii) The complement of any decidable language is semidecidable.

(iii) The intersection of any two semidecidable languages is decidable.

(iv) The problem of determining if a Turing machine accepts at least 7 strings is undecidable.

(v) The problem of determining if a Turing machine has at least 7 states is undecidable.

(vi) The problem of determining if a Turing machine runs for at least 7 steps on all inputs of length $|x| \leq 7$ is undecidable.

CS-C2160 Theory of Computation, Spring 2022
Problem Set 11 (Lecture weeks 11 and 12)

Homework Problems

H11.1 Prove, by using Rice's theorem, that the decision problem

> Given an arbitrary Turing machine M, is the language recognised by M finite?

i.e., the language

$$\{c_M \mid M \text{ is a Turing machine and } \mathcal{L}(M) \text{ is finite}\}$$

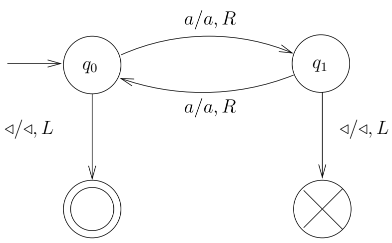
is undecidable. Define the semantic property you apply precisely, and explain why it is not a trivial one.

H11.2 Design an unrestricted grammar that generates the language

$$\{ww \mid w \in \{a, b\}^*\}.$$

Show how the string *abab* can be derived in your grammar.

H11.3 Use the construction described at Lecture 11 to design an unrestricted grammar whose derivations simulate the computations of the following simple Turing machine that recognises the language $\{a^{2k} \mid k \geq 0\}$.



Give a derivation for the string aa in your grammar, and explain why the string aaa cannot be derived in it.

Supplementary Problems

S11.1 Show that all context-sensitive languages can be recognised by linear-bounded automata. (Make use of the fact that in applying the grammar's production rules, the length of the sentential form under consideration can never decrease, except in the special case of the empty string.) Deduce from this result the fact that all context-sensitive languages are decidable.

S11.2 Show that every language generated by an unrestricted grammar can also be generated by a grammar where no terminal symbols occur on the left hand side of any production.

S11.3 Show that every context-sensitive grammar can be put in a normal form where the productions are of the form $S \to \varepsilon$ or $\alpha A \beta \to \alpha \omega \beta$, where A is a nonterminal symbol and $\omega \neq \varepsilon$. (S denotes here the start symbol of the grammar.)