Skriftlig Eksamen
Beregnelighed (DM517)

Institut for Matematik & Datalogi
Syddansk Universitet

Mandag den 31 Oktober 2011, kl. 9–13

Alle sædvanlige hjælpemidler (lærebøger, notater etc.) samt brug af lommeregner er tilladt. Eksamenssættet består af 5 opgaver på siderne (2–6). Fuld besvarelse er besvarelse af alle opgaver. De enkelte opgavers vægt ved bedømmelsen er angivet i procent. Der må gerne refereres til resultater fra lærebogen og ugesedlerne. Specielt må man gerne begrunde en påstand med at henvise til, at den umiddelbart følger fra et resultat i Sipser (hvis dette altså er sandt!). I må gerne bruge metoder eller udvidelser af sætninger som er udledt i opgaver, der er stillet i løbet af kurset.

Bemærk dog, at det ikke er tilladt at besvare et delspørgsmål, udelukkende med en henvisning til, at det følger af en af opgaverne. Henvisninger til andre bøger (ud over lærebogen) accepteres ikke som besvarelse af et spørgsmål! **Husk at begrunde alle dine svar!**

You are allowed to use the text-book, your personal notes and a pocket calculator. The exam contains 5 problems, each on a single page. The danish version can be found on pages 2-6, the english version is on pages 7-11.
A complete solution consists of a solution for all 5 problems. How much a problem weights can be seen from the numbers given in brackets for each problem. In general you may refer to results from the Sipser and weekly notes unless it is explicitely stated that you may not. The same holds for problems solved in the exercise sessions. Of course, if you cite such a result it has to be obvious that your claim really is a straightforward consequence of that result. It will not be accepted as an answer to refer to other books than the text-book. **Recall that you have to explain all your claims!**

OPGAVE 1 (20%)

Lad $L_1 = \{a^n | 3 \text{ går op i } n\}$ og lad $L_2 = \{a^n | 2 \text{ går op i } n\}$.

Spørgsmål a:

Angiv deterministiske endelige automater (DFA'er) M_1, M_2 som opfylder at $L(M_1) = L_1$ og $L(M_2) = L_2$.

Spørgsmål b:

Angiv en DFA M_3 så $L(M_3) = L_1 \cap L_2$.

Spørgsmål c:

Angiv en NFA M_4 så $L(M_4) = L_1 \cup L_2$.

Spørgsmål d:

Angiv et regulært udtryk for $L_1 \cup L_2$.

OPGAVE 2 (25%)

Lad L_1 være mængden af strenge over $\Sigma = \{a, b, c\}$ som opfylder, at antallet af a'er er det samme som antallet af b'er. Tilsvarende er L_2 mængden af strenge over $\Sigma = \{a, b, c\}$ som opfylder at antallet af b'er er det samme som antallet af c'er. Dvs,

$$L_1 = \{w \in \{a, b, c\}^* | \#_a(w) = \#_b(w)\},$$
$$L_2 = \{w \in \{a, b, c\}^* | \#_b(w) = \#_c(w)\}.$$

Spørgsmål a:

Vis, at L_1 er kontekstfrit ved at angive, på diagramform, en PDA M som opfylder at $L_1 = L(M)$. Hint: brug stakken til at holde øje med hvilket af de to symboler der er flest af.

Spørgsmål b:

Er L_1 regulært? Begrund dit svar!

Spørgsmål c:

Hvilke af følgende sprog er kontekstfrie?

1. $L_1 \cap a^*b^*$.
2. $L_1 \cup L_2$.
3. $L_1 \cap L_2$.
4. $\overline{L_1}$.
5. $L_1 - L_2$.

Du skal begrunde dine svar! Du må bruge uden bevis at sproget $L' = \{a^n b^n c^m | n, m \geq 0, m \neq n\}$ ikke er kontekstfrit.

OPGAVE 3 (15%)

Lad L_1, L_2, L_3 være regulære sprog over et alfabet Σ. Antag, at vi har givet endelige automater M_1, M_2, M_3 hvor der gælder, at $L(M_i) = L_i$ for $i = 1, 2, 3$.

Spørgsmål a:

Beskriv kort i ord algoritmer til at afgøre nedenstående 3 spørgsmål:

(1) Er $L_1 \cap L_2 = \emptyset$?

(2) Er $L_3 = L_1 \cap L_2$?

(3) Er $L_1 \cup L_2 \cup L_3 = \Sigma^*$?

Spørgsmål b:

Kan vi stadig lave sådanne algoritmer hvis L_1, L_2, L_3 blot er angivet i form af non-deterministiske endelige automater?

OPGAVE 4 (15%)

Fibonacci tallene er givet ved følgende rekursionsligning, hvor f_k betegner det k'te Fibonacci tal: $f_1 = 1, f_2 = 1,\ f_k = f_{k-1} + f_{k-2}$ når $k \geq 3$.

Lad $L = \{a^n : n \text{ er et Fibonacci tal }\}$. Man kan vise ved hjælp af pumpelemmaet, at L ikke er regulært. Du skal ikke gøre dette.

Spørgsmål a:

Gør rede for, at L ikke er kontekstfrit.

Spørgsmål b:

Beskriv i ord en Turing maskine M som afgør L. Dvs du skal angive de væsentlige skridt som M tager når den kører på sit input.

OPGAVE 5 (25%)

For hvert af følgende problemer skal du argumentere for om de kan afgøres af en Turing maskine, eller er uafgørlige. Du må gerne anvende Rice's sætning, der hvor det er muligt. Hvis du anvender den, skal du forklare hvorfor den kan anvendes på det pågældende problem. For de problemer som er afgørlige, skal du give en kort forklaring på, hvordan de kan afgøres af en deterministisk Turing maskine.

1. Givet den universelle kodning $< M >$ for en Turing maskine M samt k strenge $w_1, w_2, \ldots, w_k$ over M's alfabet; gælder der at $w_i \in L(M)$ for mindst et $i \in \{1, 2, \ldots, k\}$ (dvs M accepterer mindst en af disse strenge)?

2. Givet den universelle kodning $< M >$ for en Turing maskine M samt et naturligt tal k; Indeholder $L(M)$ mindst k forskellige strenge?

3. Givet den universelle kodning $< M >$ for en Turing maskine M samt et naturligt tal k; stopper M på alle strenge (over dens alfabet) som har længde højst k?

4. Givet den universelle kodning $< M >$ for en non-deterministisk Turing maskine M, en streng w over M's alfabet, samt et naturligt tal k; er det muligt for M at bruge mindst k skridt når den startes på input w?

5. Givet den universelle kodning $< M >$ for en Turing maskine M, en streng w over M's alfabet og et naturligt tal k; Vil M gentage nogen tilstand mindst k gange, når den startes på strengen w?

6. Givet den universelle kodning $< M >$ for en Turing maskine M, en streng w over M's alfabet og et naturligt tal k; Vil M gentage en af sine tilstande højst k gange, når den køres på input w?

PROBLEM 1 (20%)

Let $L_1 = \{a^n | 3 \text{ divides } n\}$ and let $L_2 = \{a^n | 2 \text{ divides } n\}$.

Question a:

Describe deterministic finite automata M_1, M_2 such that $L(M_1) = L_1$ and $L(M_2) = L_2$.

Question b:

Describe a DFA M_3 such that $L(M_3) = L_1 \cap L_2$.

Question c:

Describe an NFA M_4 such that $L(M_4) = L_1 \cup L_2$.

Question d:

Describe a regular expression R such that $L(R) = L_1 \cup L_2$.

PROBLEM 2 (25%)

Let L_1 be the set of strings over $\Sigma = \{a, b, c\}$ which have the same number of a's and b's. Similarly, L_2 is the set of strings over $\Sigma = \{a, b, c\}$ with the same number of b's and c's. That is,

$$L_1 = \{w \in \{a, b, c\}^* | \#_a(w) = \#_b(w)\},$$
$$L_2 = \{w \in \{a, b, c\}^* | \#_b(w) = \#_c(w)\}.$$

Question a:

Prove that L_1 is context-free, by describing, on diagram form, a PDA M such that $L_1 = L(M)$. Hint: use the stack to keep track of which of the two symbols you have currently seen the most of.

Question b:

Is L_1 regular? Justify your answer!

Question c:

Which of the following languages are context-free?

1. $L_1 \cap a^*b^*$.
2. $L_1 \cup L_2$.
3. $L_1 \cap L_2$.
4. $\overline{L_1}$.
5. $L_1 - L_2$.

You must justify your answers! You may use, without proof, that the language $L' = \{a^n b^n c^m | n, m \geq 0, m \neq n\}$ is not context-free:

PROBLEM 3 (15%)

Let L_1, L_2, L_3 be regular languages over an alphabet Σ. Suppose that we are given deterministic finite automata's M_1, M_2, M_3 such that $L(M_i) = L_i$ for $i = 1, 2, 3$.

Question a:

Describe in words algorithms for deciding each of the following 3 questions:

(1) is $L_1 \cap L_2 = \emptyset$?

(2) is $L_3 = L_1 \cap L_2$?

(3) is $L_1 \cup L_2 \cup L_3 = \Sigma^*$?

Question b:

Can we still construct such algorithms if L_1, L_2, L_3 are only given by non-deterministic finite automata?

PROBLEM 4 (15%)

The Fibonacci numbers are given by the following recursion, where f_k denotes the k'th Fibonacci number: $f_1 = 1, f_2 = 1,\ f_k = f_{k-1} + f_{k-2}$ for $k \geq 3$.

Let $L = \{a^n : n \text{ is a Fibonacci number }\}$. It can be shown by the pumping lemma that L is not regular. You should not do this.

Question a:

Show that L is not context-free.

Question b:

Describe in words a Turing machine M which decides L. That is, you should describribe the important steps which M makes when running on it input.

PROBLEM 5 (25%)

For each of the following problems you must argue whether they are decidable by a Turing machine, or they are undecidable. You may use Rice's theorem where is applies. If you use it, you must justify that it can be applied for the problem at hand. For those problems which are decidable you must give a short explanation for how they can be decided by a deterministic Turing machine.

1. Given the universal encoding $< M >$ of a Turing machine M and k strings $w_1, w_2, \ldots, w_k$ over M's alphabet; is $w_i \in L(M)$ for at least one $i \in \{1, 2, \ldots, k\}$ (that is, does M accept at least one of these strings)?

2. Given the universal encoding $< M >$ of a Turing machine M and a natural number k; does $L(M)$ contain at least k distinct strings?

3. Given the universal encoding $< M >$ of a Turing machine M and a natural number k; does M halt on all strings (over its alphabet) which have length at most k?

4. Given the universal encoding $< M >$ of a non-deterministic Turing machine M, a string w over M's alphabet and a natural number k; does M have the possibility to use at least k steps on input w?

5. Given the universal encoding $< M >$ of a Turing machine M, a string w over M's alphabet and a natural number k; Will M repeat some state at least k times when it is started on input w?

6. Given the universal encoding $< M >$ of a Turing machine M, a string w over M's alphabet and a natural number k; Will M repeat one of its states at most k times when it is started on input w?