Name/DoB: ______________

Foundations of Computing: Discrete Mathematics

Reexam
February 23rd, 2015

Instructions (Read Carefully)

What to check. In the multiple-choice questions, there is one and only one correct answer. You should only check 1 box.

Useful Definitions. At the end of this document, you can find some definitions that could be useful for answering some questions.

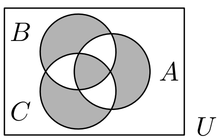
Info about you. Write *clearly* your full name and your date of birth on every page (top-right).

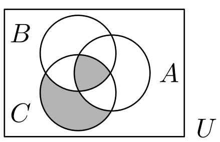
1. Answer the following multiple choice questions:

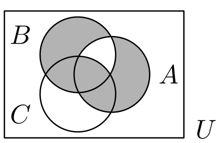
(a) (*2 pt.*) In this problem, we will be using base-7 numbers, instead of base-10. What is $346_7 + 165_7$ when expressed in base-7?

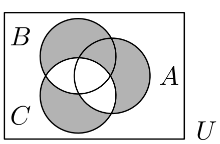
- [ ] 1330_7
- [x] 544_7
- [ ] 511_7
- [ ] 277_7

(b) (*2 pt.*) Let A and B and C be subsets of a universal set U. Which of the following Venn diagrams describes the set $(C^c \cap (A \cup B)) \cup (C - (A \cup B))$?

- [ ] 

- [ ] 

- [ ] 

- [x] 

(c) (*2 pt.*) Suppose $f : \mathbb{N} \to \mathbb{N}$, where $\mathbb{N}$ is the set of natural numbers and

$$f(n) = \begin{cases} n^2 & \text{if } n < 8 \\ n+1 & \text{if } n \geq 5 \end{cases}$$

Which of the following statements is TRUE?

- [ ] f is not a function because $f(3) = 9$ and $f(8) = 9$.
- [ ] f is a function.
- [x] f is not a function because $f(6)$ is equal to both 36 and 7.
- [ ] f is not a function because there is no natural number n such that $f(n) = 2$.

(d) (*2 pt.*) Let $\mathcal{S}$ be the relation $\{(1,2),(1,3),(2,3),(2,4),(3,1)\}$, and let R be the relation $\{(2,1),(3,1),(3,2),(4,2)\}$. Find $R o S$.

- ☐ $\{(2,2),(2,3),(3,2),(3,3),(3,4),(4,3),(4,4)\}$.
- ☐ $\{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(2,4),(3,1),(3,2),(4,2),(1,4),(3,4)\}$.
- ☐ $\{(1,1),(1,2),(1,3),(2,1),(2,2),(2,3),(2,4),(3,1)\}$.
- ☑ $\{(1,1),(1,2),(2,1),(2,2)\}$.

(e) (*2 pt.*) A number $n \geq 3$ of persons meet at a party, and some of them shake hands. It is known that at least one person does not shake hands with everybody (that is, maybe he/she shakes hands with someone, but not with everyone). What is the maximum number of persons that may have shaken hands with everyone?

- ☐ $n/2$
- ☐ $n-1$
- ☑ $n-2$
- ☐ None of the above.

(f) (*2 pt.*) Let us define an Autonomous Turing Machine (ATM) to be a turing machine that doesn't take an input, but simply starts on a blank tape. Which of the following languages about ATMs is decidable?

- ☑ $\{w|$ w is an ATM that never uses more than 4 million tape cells$\}$
- ☐ $\{w|$ w is an ATM that terminates in the accept state$\}$
- ☐ $\{w|$ w is an ATM that writes "Hello World" on the tape$\}$
- ☐ $\{w|$ w is an ATM that writes an infinite list of prime numbers on the tape $\}$

(g) (*2 pt.*) An ITU username is lucky, if it has exactly four characters (there are 26 characters from a-z), and three consecutive characters are identical. E.g.,

- aaab is lucky, since there are three consecutive a's.
- ccbc is not lucky; even though there are three c's, they are not consecutive.
- beee is lucky, since there are three consecutive e's.
- abdc is not lucky

How many lucky usernames are there?

- ☐ $67600 = \binom{26}{3} \cdot 26$
- ☐ $5200 = \binom{26}{3} \cdot 2$
- ☑ $1300 = \binom{26}{2} \cdot 4$
- ☐ $650 = \binom{26}{2} \cdot 2$

The following questions are "open-answer", which means that you must write an answer instead of checking a box. Be brief but precise, your correct use of mathematical notation is an important aspect.

2. (*4 pt.*) Prove that the following formula is a contradiction:

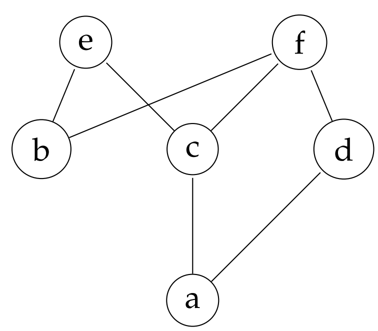
$$(\sim (A \to B)) \land (B \lor \sim A)$$

by constructing the truth table for all sub expressions. $\sim A$ means the negation of A.

Solution:

A	B	$A \to B$	$\sim (A \to B)$	$\sim A$	$B \lor \sim A$	$(\sim (A \to B)) \land (B \lor \sim A)$
T	T	T	F	F	T	F
T	F	F	T	F	F	F
F	T	T	F	T	T	F
F	F	T	F	T	T	F

3. The following is the Hasse diagram of a partial order.



(a) (*1 pt.*) List all ordered pairs contained in the relation.
Solution:
$(a,a),(b,b),(c,c),(d,d),(e,e)$
$(a,c),(a,d),(a,e),(a,f)$
$(b,e),(b,f)\ (c,e),(c,f)\ (d,f)$

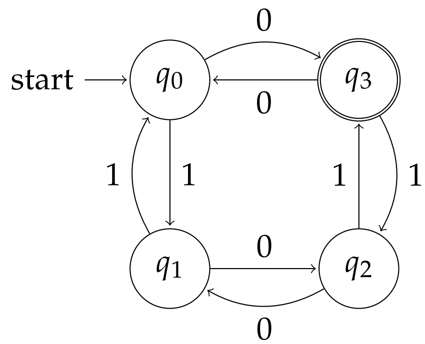
(b) (*1 pt.*) Find the maximal and minimal elements.
Solution: Maximal: $\{e,f\}$ Minimal: $\{a,b\}$

(c) (*1 pt.*) Is there a greatest element?
Solution: No.

(d) (*1 pt.*) Is there a least element?
Solution: No.

4.

(a) (*2 pt.*) What is the language recognized by the following deterministic finite-state automaton?



Solution:

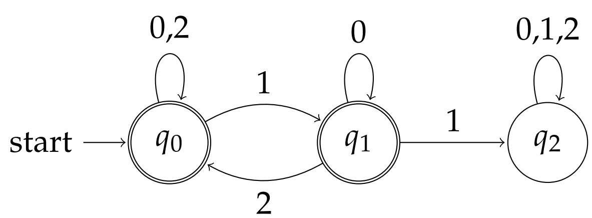
$$L = \{w \in \{0,1\}^* \mid w \text{ has an odd number of 0s and an even number of 1s}\}$$

(b) (*2 pt.*) Build (draw) a deterministic finite-state automaton that recognizes the following language:

$$\{w : w \in \{0,1,2\}^* \text{ and } w \text{ has at least one 2 between any pair of 1s }\}$$

e.g. 0012001001 is not in the language, since there are no 2's between the second and third 1.

Solution: There are several correct answers to this. Here is one of the simplest:



5. (*4 pt.*) Construct a context free grammar that generates the same language as the regular expression "a(ab | c)*ab+"

Solution: There are several possible solutions. Here is one:

$$\begin{array}{lcl} S & ::= & aA \\ A & ::= & abA \mid cA \mid B \\ B & ::= & abB \mid ab \end{array}$$

6. (*4 pt.*) Let the "Tribonacci sequence" be defined by $T_1 = T_2 = T_3 = 1$ and $T_n = T_{n-1} + T_{n-2} + T_{n-3}$ for $n \geq 4$. Prove using induction that $T_n < 2^n$ for all $n \in \mathbb{N}$.

Solution:

Basis: $n \in \{1, 2, 3\}$

$$T_1 = 1 < 2^1$$
$$T_2 = 1 < 2^2$$
$$T_3 = 1 < 2^3$$

IH: $P(k) =$ "$T_i < 2^i, \quad \forall i \in \{1, ..., k\}$"

Assuming $P(k)$, we need to prove $P(k+1)$.

$$\begin{aligned}
T_{k+1} &= T_k + T_{k-1} + T_{k-2} && \text{by definition}\\
&< 2^k + 2^{k-1} + 2^{k-2} && \text{by the induction hypothesis}\\
&= 2^{k+1}\left(\frac{1}{2} + \frac{1}{4} + \frac{1}{8}\right) && \text{rewriting}\\
&< 2^{k+1} && \text{since } \frac{7}{8} < 1
\end{aligned}$$

Together with $P(k)$, this proves $P(k+1)$.

Some useful information for the exam

Logics. Here are some of the rules for arguments in propositional logic.

(Modus Ponens) $\begin{array}{l} p \\ p \to q \\ \hline \therefore q \end{array}$ (Modus Tollens) $\begin{array}{l} \neg q \\ p \to q \\ \hline \therefore \neg p \end{array}$

(Addition) $\begin{array}{l} p \\ \hline \therefore p \vee q \end{array}$ (Simplification) $\begin{array}{l} p \wedge q \\ \hline \therefore p \end{array}$ (Conjuntion) $\begin{array}{l} p \\ q \\ \hline \therefore p \wedge q \end{array}$

(Or Elimination) $\begin{array}{l} p \vee q \\ \neg q \\ \hline \therefore p \end{array}$ $\begin{array}{l} p \vee q \\ \neg p \\ \hline \therefore q \end{array}$

Sets. A set is an (unordered) collection of objects, called *elements* or *members*.
The *union* of two sets A and B is the set

$$A \cup B = \{x \ : \ x \in A \vee x \in B\}.$$

The *intersection* of A and B is the set

$$A \cap B = \{x \ : \ x \in A \wedge x \in B\}.$$

Given n sets $A_1, A_2, \ldots, A_n$,

$$\bigcup_{i=1}^{n} A_i = A_1 \cup \ldots \cup A_n \qquad \bigcap_{i=1}^{n} A_i = A_1 \cap \ldots \cap A_n.$$

The *difference* of two sets A and B, denoted by $A - B$ (or by $A \setminus B$), is the set containing those elements in A but not in B.

The *Cartesian product* of two or more sets $A_1, A_2, \ldots, A_n$, denoted by $A_1 \times A_2 \times \ldots \times A_n$, is the set of all ordered n-tuples $(a_1, a_2, \ldots, a_n)$, where $a_i \in A_i$ for $1 \leq i \leq n$.

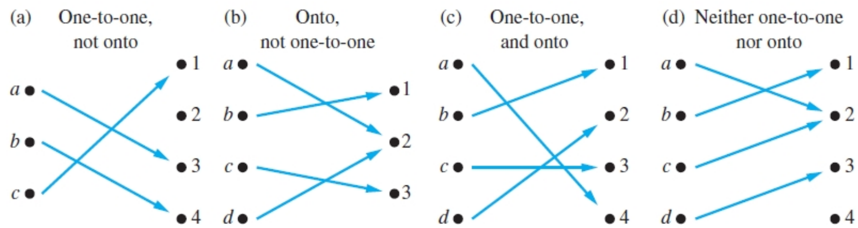
Functions. Given two non-empty sets A and B, a *function* f from A to B is an assignment of exactly one element of B to each element of A.

A function $f : A \to B$ is *onto* (or a surjection) if and only if for every element $b \in B$ there is an element $a \in A$ such that $f(a) = b$.

A function $f : A \to B$ is *one-to-one* (or an injunction) if $f(a) = f(b)$ implies $a = b$ for all a and b in the domain of f.

A function f is a *bijection* if it is both one-to-one and onto.

Example:



Relations. A relation $\mathcal{R}$ on a set A is a subset of the cartesian product $A \times A$.

A relation $\mathcal{R}$ on A is *reflexive* whenever

$$\forall a \in A.\ (a, a) \in \mathcal{R}$$

A relation $\mathcal{R}$ on A is called *symmetric* if

$$\forall a, b \in A, (a, b) \in \mathcal{R} \Rightarrow (b, a) \in \mathcal{R}.$$

A relation $\mathcal{R}$ on A is *antisymmetric* if

$$\forall a, b \in A,\ ((a, b) \in \mathcal{R} \wedge (b, a) \in \mathcal{R}) \Rightarrow a = b.$$

A relation $\mathcal{R}$ on A is *transitive* if

$$\forall a, b, c \in A,\ ((a, b) \in \mathcal{R} \wedge (b, c) \in \mathcal{R}) \Rightarrow (a, c) \in \mathcal{R}.$$

The *reflexive closure* of a binary relation $\mathcal{R}$ on A is the smallest reflexive relation on A that contains $\mathcal{R}$.

The *symmetric closure* of a binary relation $\mathcal{R}$ on A is the smallest symmetric relation on A that contains $\mathcal{R}$.

The *transitive closure* of a binary relation $\mathcal{R}$ on A is the smallest transitive relation on A that contains $\mathcal{R}$.

Probability Theory *Bayes' Theorem* allows to manipulate conditional probabilities:

$$p(A_i|B) = \frac{p(B|A_i)p(A_i)}{p(B)}$$

such that $p(B) = p(B|A_1)p(A_1) + p(B|A_2)p(A_2)$.

Choose r objects from n	Order matters, not all elements (r-permutations)	Order matters, all elements (permutations)	Order does not matter, not all elements (combinations)
Without repetitions	$P(n,r) = \frac{n!}{(n-r)!}$	$P(n,n) = n!$	$C(n,r) = \binom{n}{r} = \frac{n!}{r!\,(n-r)!}$
With repetitions	n^r	$\frac{n!}{n_1!n_2!\cdots n_k!}$ where $n = n_1 + n_2 + \ldots + n_k$	$\binom{n+r-1}{r}$

Number Theory. Given two integers a and b, with $a \neq 0$, we say that a *divides* b if there is an integer c such that $b = ac$, or in other words, if $\frac{b}{a}$ is an integer. If a divides b then a is a *factor* (or *divisor*) of b, and b is said to be a *multiple* of a.

The *greatest common divisor* of two integers a and b, denoted by $\gcd(a,b)$, is the largest integer that divides both a and b.

The *Euclidean algorithm* provides a very efficient way to compute the greatest common divisor of two integers.

Given two positive integers a and b, the smallest positive integer that is a multiple of both a and b is the *least common multiple*, denoted by $\operatorname{lcm}(a,b)$.

The Quotient-Remainder Theorem. Let a be an integer and d a positive integer. Then there exist unique integers q and r, with $0 \leq r < d$, such that $a = dq + r$.

The value d is called the *divisor*, a is the *dividend*, q is the *quotient*, and r is the *remainder*. Then $q = a \text{ div } d$, $r = a \bmod d$. Remember that the remainder cannot be negative.

Graph Theory. A graph $G = (V,E)$ is a structure consisting of a set of *vertices* (or nodes) V, and a set of *edges* E connecting some of these vertices.

Handshake Theorem. Let G be an undirected graph. Then,

$$\sum_{v \in V} deg(v) = 2m$$

where m is the number of edges of G and V is the set of vertices.

Let n be a nonnegative integer, and v, w two vertices in an undirected graph G.

A *walk from v to w* is an alternating sequence of vertices and edges

$$v_0 e_1 v_1 e_2 \cdots v_{n-1} e_n v_n$$

Algorithm 4.8.2 Euclidean Algorithm

[Given two integers A and B with $A > B \geq 0$, this algorithm computes $\gcd(A, B)$*. It is based on two facts:*

1. $\gcd(a, b) = \gcd(b, r)$ *if a, b, q, and r are integers with* $a = b \cdot q + r$ *and* $0 \leq r < b$.
2. $\gcd(a, 0) = a$.*]*

Input: A, B *[integers with $A > B \geq 0$]*

Algorithm Body:

$a := A, b := B, r := B$

[If $b \neq 0$, compute a mod b, the remainder of the integer division of a by b, and set r equal to this value. Then repeat the process using b in place of a and r in place of b.]

while $(b \neq 0)$

  $r := a \bmod b$

[The value of a mod b can be obtained by calling the division algorithm.]

  $a := b$

  $b := r$

end while

[After execution of the **while** *loop,* $\gcd(A, B) = a$.*]*

$\gcd := a$

Output: gcd *[a positive integer]*

going from $v = v_0$ to $w = v_n$. We can repeat edges and vertices.

A *trail from v to w* is a walk from v to w with no repeated edges.

A *path from v to w* is a trail with no repeated vertices. Thus it is a sequence of vertices and edges with no repeated edges nor vertices.
A *circuit* is a trail that starts and ends at the same vertex, and has length greater than zero.

A circuit is *simple* if it does not contain repeat vertices (except the first and last).

An undirected graph is called *connected* if there is a walk between every pair of distinct vertices of a graph. Otherwise, it is called *disconnected*.

A *tree* is an undirected simple graph G that satisfies any of the following equivalent conditions:

1. G is connected and has no cycles.
2. G has no cycles, and a simple cycle is formed if any edge is added to G.
3. G is connected, but is not connected if any single edge is removed from G.

4. Any two vertices in G can be connected by a unique simple path.

A *trivial tree* is a graph that consists of a single vertex. A graph is called a *forest* if, and only if, it does not have any circuit and is not connected.

Decidability.

- A program can either *accept* an input or *reject* it.
- The set of strings accepted by a program P is the language *recognised* by P . A language is *Turing recognisable* if there exists some program recognising it.
- A program may *loop*, because either it terminates (accepting or rejecting), or it doesn't terminate.
- A program may fail to accept an input by either entering a rejecting configuration or by looping.
- A non-looping program is called a *decider*, it always accepts or rejects an input.
- A decider that recognises a language L is said to decide L. A language is *decidable* if some program decides it.