Foundations of Computing: Discrete Mathematics

Exam
January 7th, 2015

Instructions (Read Carefully)

What to fill in. In the multiple-choice questions, there is one and only one correct answer. You should only check 1 box. For the open-answer questions, write or draw your answer in the blank space below the question.

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).

- IMPORTANT -
Only information written on pages 2-8 will be evaluated.
Anything else that you hand-in will NOT be considered for the final evaluation!

1. Answer the following multiple choice questions:

(a) (*2 pt.*) Given any two integers a and b with $a \neq 0$, let $a|b$ denote that a divides b. Which of the following statements is TRUE?

- ☐ An integer n is divisible by 6 if and only if it is divisible by 3.
- ☐ For all integers a, b and c, where $a \neq 0$, $a|bc$ if and only if $a|b$ and $a|c$.
- ☐ If r and s are integers, then $r|s$ if and only if $r^2|s^2$.
- ☐ For all integers a, b and c, where $a \neq 0$, $a|(b+c)$ if and only if $a|b$ and $a|c$.

(b) (*2 pt.*) Suppose $S = \{a, b, \{a\}\}$, and let $\mathcal{P}(S)$ be the power set of the set S. Consider the following six statements and determine which ones are TRUE:

$$(i)\ \{b\} \in S \qquad (ii)\ \{a\} \subseteq \mathcal{P}(S) \qquad (iii)\ \{a, b\} \in \mathcal{P}(S)$$

$$(iv)\ \{a, b\} \in S \qquad (v)\ \{\{a\}\} \in \mathcal{P}(S) \qquad (vi)\ \{a, \{a\}\} \in \mathcal{P}(S)$$

- ☐ iii, v, vi
- ☐ ii, iii, vi
- ☐ ii, iv, v
- ☐ i, ii, vi

(c) (*2 pt.*) Let f be a function from A to B, where $A = \{-3, -2, -1, 0, 1, 2\}$ and f is defined by $f(x) = x^2$. For which set B is the f onto?

- ☐ $\{-9, -4, -1, 0, 1, 4\}$.
- ☐ $\{0, 1, 4, 9\}$.
- ☐ $\{0, 1, 4\}$.
- ☐ $\{-3, -2, -1, 0, 1, 2\}$.

(d) (*2 pt.*) Let $R = \{(0,2), (1,1), (1,2), (2,0), (2,1), (2,2)(2,3), (3,2), (3,3)\}$ be a relation on the set $S = \{0, 1, 2, 3\}$. Which one of the following statements is true?

- ☐ R is symmetric, but neither reflexive nor transitive.
- ☐ R is symmetric and reflexive, but not transitive.
- ☐ R is reflexive and transitive, but not symmetric.
- ☐ R is transitive, but neither symmetric nor reflexive.

(e) (2 *pt.*) Which one of the following is TRUE for any simple undirected graph G with at least two vertices?

- ☐ No two vertices have the same degree.
- ☐ At least two vertices have the same degree.
- ☐ At least three vertices have the same degree.
- ☐ All vertices have the same degree.

(f) (2 *pt.*) Which of the following grammars generates the language:

$$L = \{a^i b^j d^k \quad | \quad i, j, k \geq 0 \quad \wedge \quad j < k\}$$

☐
$$\begin{array}{lcl} S & \rightarrow & aS \mid SA \mid Ad \\ A & \rightarrow & bd \mid d \end{array}$$

☐
$$\begin{array}{lcl} S & \rightarrow & aS \mid A \\ A & \rightarrow & bAd \mid Ad \mid d \end{array}$$

☐
$$\begin{array}{lcl} S & \rightarrow & AB \\ A & \rightarrow & aAb \mid \epsilon \\ B & \rightarrow & Bd \mid d \end{array}$$

☐
$$\begin{array}{lcl} S & \rightarrow & AB \\ A & \rightarrow & aA \mid \epsilon \\ B & \rightarrow & bBd \mid d \end{array}$$

(g) (2 *pt.*) Which of the following languages is *not* decidable?

- ☐ $\{w|$ w is a Java program that includes the package `java.io.*`$\}$
- ☐ $\{w|$ w is a Java program that always prints "Hello World!"$\}$
- ☐ $\{w|$ w is a Java program that compiles without errors$\}$
- ☐ $\{(w, k)|$ w is a Java program that doesn't crash in the first k clock cycles of running$\}$

(h) (2 *pt.*) A team of six software developers have to divide a set of roles between them. They need three coders, one tester, one scrum master, and one product owner. One of the members of the team does not have the experience to be scrum master, but otherwise all other assignments of roles are allowed. In how many ways can they constitute their team?

- ☐ 720
- ☐ 120
- ☐ 100
- ☐ 54

(i) (*2 pt.*) A gambler has a special "cheating coin" with weights shifting inside, such that the previous outcome is more likely than the opposite. Specifically, if the coin just came up heads, then the probability of it coming up heads again on the next throw is 70%. Likewise, if it just came up tails, then the probability for tails again on the next throw is also 70%. What is the probability of getting the same outcome three times in a row, i.e., three heads or three tails, on three consecutive throws of the coin?

- ☐ 49%
- ☐ 34.3%
- ☐ 12.5%
- ☐ Not enough information is available to answer this question.

End of multiple choice section

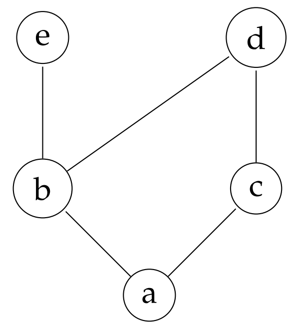
The following questions are "open-answer", which means that you must write an answer instead of checking a box. Be brief but precise, your correct use of mathematical notation is an important aspect.

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

$$((\sim A \rightarrow B) \wedge \sim B) \rightarrow A$$

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

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



(a) (*1 pt.*) List all ordered pairs contained in the relation.

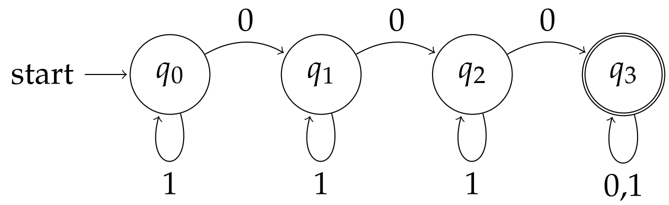
(b) (*1 pt.*) Find the maximal and minimal elements.

(c) (*1 pt.*) Is there a greatest element?

(d) (*1 pt.*) Is there a least element?

4.

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



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

$$\{w : w \in \{0,1\}^* \text{ and } w \text{ ends with } 00\}$$

5. (*4 pt.*) Let f be the function recursively defined by

$$f(n) = \begin{cases} 1 & \text{, if } n \leq 2 \\ 2 \cdot f(n-2) + f(n-1) & \text{, otherwise} \end{cases}$$

Prove, using induction, that $f(n)$ is odd for all positive integers n.

Some useful information for the exam

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

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

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

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

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 $1 \leq i \leq n$.

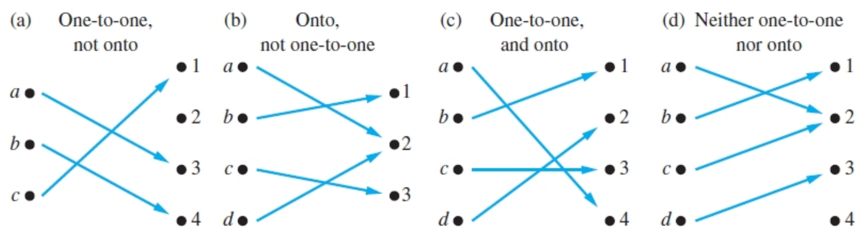
Functions. Given two non-empty sets A and B, a *function* f from A to B is an assignment of exactly one element of B to each element of A.

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 $\mathrm{lcm}(a,b)$.

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.

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]*

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

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

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.