Name: Matriculation Number:

Midterm Exam
General CS 1 (320101)

October 21, 2008

You have one hour(sharp) for the test;
Write the solutions to the sheet.

The estimated time for solving this exam is 55 minutes, leaving you 5 minutes for revising your exam.

You can reach 29 points if you solve all problems. You will only need 27 points for a perfect score, i.e. 2 points are bonus points.

Different problems test different skills and knowledge, so do not get stuck on one problem.

	To be used for grading, do not write here									
prob.	1.1	1.2	1.3	1.4	2.1	2.2	3.1	3.2	Sum	grade
total	2	2	3	4	3	5	6	4	29	
reached										

Good luck to all students who take this test

1 Mathematical Foundations

2pt
2min

Problem 1.1 (Greek Alpabet)

Fill in the blanks in the table of Greek letters. Note that capitalized names denote capital Greek letters.

Symbol	θ	τ	ν	ι				
Name					gamma	chi	xi	rho

Solution:

Symbol	θ	τ	ν	ι	γ	χ	ξ	ρ
Name	theta	tau	nu	iota	gamma	chi	xi	rho

Problem 1.2 (Properties of Sets) 2pt 3min

Prove that:

1. $A \cap ((B \cup C)) = A \cap B \cup A \cap C$
2. $A \cup B \cap C = ((A \cup B)) \cap ((A \cup C))$

Use MathTalk throughout the proof.

Solution:

1.
$$\begin{aligned} A \cap ((B \cup C)) &= \{x \mid x \in A \wedge (x \in B \vee x \in C)\} \\ &= \{x \mid (x \in A \wedge x \in B) \vee (x \in A \wedge x \in C)\} \\ &= A \cap B \cup A \cap C \end{aligned}$$

2.
$$\begin{aligned} A \cup B \cap C &= \{x \mid x \in A \vee (x \in B \wedge x \in C)\} \\ &= \{x \mid (x \in A \vee x \in B) \wedge (x \in A \vee x \in C)\} \\ &= ((A \cup B)) \cap ((A \cup C)) \end{aligned}$$

Problem 1.3 (Sets and Functions) 3pt
5min

Let A and B be sets such that:

- A: $\forall S.S \textit{ is a set} \Rightarrow (A \subset S)$
- B: $B = \mathcal{P}(A)$

and let $f\colon A \to B$ be a total injective function from A to B.
Your task is to:

1. state when a function $f\colon A \to B$ is called injective. Use math-talk.
2. identify A and B.
3. give an example for f or explain why f does not exist.

Solution:

1. $\forall x, y \in A. f(x) = f(y) \Rightarrow x = y$
2. $A = \emptyset$, $B = \{\emptyset\}$
3. $f = \emptyset$

Problem 1.4 (Bernoulli inequality) 4pt 5min

Prove by induction the Bernoulli inequality:

$$(1+x)^n \geq nx$$

where $n \in \mathbb{N}$, $x \in \mathbb{Q}$, and $x \geq -1$

Hint: You can acomplish this by proving a stronger statement first, namely that the left hand side is greater or equal to $nx+1$.

Solution:

Proof:

P.1 We have two cases

P.1.1 $n = 0$:

P.1.1.1 $(1+x)^0 = 1 \geq 1 = 0x + 1$. □

P.1.2 Step case: $n \Longrightarrow n+1$:

P.1.2.1 Assume $(1+x)^n \geq nx + 1$.

P.1.2.2 $(1+x)^{n+1} = (1+x)(1+x)^n \geq (1+x)(nx+1) = 1+nx+x+nx^2 = (n+1)x+1+nx^2 \geq (n+1)x+1$, since $nx^2 \geq 0$. □

P.2 We have proven that $(1+x)^n \geq nx+1$ thus $(1+x)^n \geq nx$. □

2 Abstract Data Types and Abstract Procedures

3pt
5min

Problem 2.1 (ADT for trains)

Write an ADT for train configurations. Each train has a locomotive (engine) in the front and a number of cars attached to it. Each car can be either a passenger car or a cargo car. Cargo cars are characterized by capacity which can be either 'small' or 'large'.

Using your representation of trains, write down a train with three passenger cars after the engine and two cargo cars at the end, having large and small capacity, respectively.

Solution:

1. $\langle\{\mathbb{T}, \mathbb{C}\}, \{[\text{large}\colon \mathbb{C}], [\text{small}\colon \mathbb{C}], [\text{loc}\colon \mathbb{T}], [\text{pass\_c}\colon \mathbb{T} \to \mathbb{T}], [\text{carg\_c}\colon \mathbb{C} \times \mathbb{T} \to \mathbb{T}]\}\rangle$
2. $\text{carg\_c}(\text{small}, \text{carg\_c}(\text{large}, \text{pass\_c}(\text{pass\_c}(\text{pass\_cloc}))))$

Problem 2.2 (Abstract Procedures) 5 pt 15 min

Given the ADT for natural numbers

$$\langle\{\mathbb{N}\}, \{[o\colon \mathbb{N}], [s\colon \mathbb{N} \to \mathbb{N}]\}\rangle$$

and the following procedures:
$\langle f{::}\mathbb{N} \times \mathbb{N} \to \mathbb{N}\,;\, \{f(o,o) \rightsquigarrow o, f(o,y) \rightsquigarrow g(o,y), f(s(x),y) \rightsquigarrow s(g(x,s(y)))\}\rangle$
$\langle g{::}\mathbb{N} \times \mathbb{N} \to \mathbb{N}\,;\, \{g(o,o) \rightsquigarrow o, g(x,o) \rightsquigarrow f(x,o), g(x,s(y)) \rightsquigarrow s(s(f(x,y)))\}\rangle$

1. Show the computation process for:
   $f(s(s(o)), o)$
   and
   $f(s(o), s(s(o)))$
2. What arithmetic expression does f compute and what arithmetic expression does g compute?
3. Do f and g terminate for all inputs?

Solution:

1. $f(s(s(o)), o) \rightsquigarrow s(g(s(o), s(o)))$
   $\rightsquigarrow s(s(s(f(s(o), o))))$
   $\rightsquigarrow s(s(s(s(g(o, s(o))))))$
   $\rightsquigarrow s(s(s(s(s(f(o, o)))))))$
   $\rightsquigarrow s(s(s(s(s(o)))))$

   $f(s(o), s(s(o))) \rightsquigarrow s(g(o, s(s(s(o)))))$
   $\rightsquigarrow s(s(s(f(o, s(s(o))))))$
   $\rightsquigarrow s(s(s(g(o, s(s(o))))))$
   $\rightsquigarrow s(s(s(s(f(o, s(o)))))))$
   $\rightsquigarrow s(s(s(s(g(o, s(o)))))))$
   $\rightsquigarrow s(s(s(s(s(f(o, o))))))))$
   $\rightsquigarrow s(s(s(s(s(o)))))))$
2. Both f and g compute the same function $f(x, y) = g(x, y) = x + 2(x + y)$
3. f and g terminate for all inputs.

---

3 Programming in Standard ML

6pt
10min

Problem 3.1 (Frequency of characters in a list)

Write an SML function that given a string returns the frequency of characters in that string. The signature of the function is `fn : string -> (char * int) list`

For example

```
freq "Red␣Riding␣Hood";

val it = [(#"R",2),(#"e",1),(#"d",3),(#"␣",2),(#"i",2),(#"n",1),
          (#"g",1),(#"H",1), (#"o",2)] : (char * int) list
```

Solution:

```
(*Removes a character from a list*)
fun remove (a, []) = [] |
       remove(a, h::t) = if a = h then remove (a, t) else h::remove(a, t);

(*Counts the occurence of a character in a list*)
fun count(a, []) = 0 |
       count(a, h::t) = if a = h then 1+count(a, t) else count(a, t);


(*The function that returns a list of character and frequency pairs*)
fun freq_help([]) = [] |
       freq_help(h::t) = (h, count(h, h::t))::freq_help(remove(h, t));


(*Final function that explodes the string*)
fun freq(x) = freq_help(explode(x));
```

Problem 3.2 (Find My Children) 10pt 10min

Suppose you have 2 lists given, the first one contains husband-wife pairs, the second one contains mother-child pairs. Write an SML function `FatherChildren` that returns a list of all children for a given father, or `nil` if the father has no children yet. If the father is not in the list, you raise a `NoFather` exception. Assume that there are no two fathers with the same name, and there are no two mothers with the same name.

The signature of the function is

```
fn : string * (string * string) list * (string * string) list -> string list
```

ex:

```
val x = [("Brad", "Angelina"), ("Ramratan", "Shashi"), ("Dragi", "Vesna")];
val y = [("Angelina", "Shiloh"),("Angelina", "Knox"), ("Angelina", "Vivienne")
        , ("Shashi", "Richa"),("Vesna", "Pavlinka")];

FatherChildren("Brad", x, y);
val it = ["Shiloh","Knox","Vivienne"] : string list
```

Solution:

```
exception NoFather;

(*Finds a mother for a given father*)
fun findMother(f, nil) = raise NoFather |
       findMother(f, (a:string, b:string)::l) = if f = a then b else findMother(f, l);

(*Returns all the children for a given mother*)
fun findChildren(m, []) = [] |
       findChildren(m, (a:string, b:string)::l) =
      if m = a then b::findChildren(m, l) else findChildren(m, l);

(*Final function*)
fun FatherChildren(f:string, x, y) = findChildren(findMother(f, x), y);
```
