Midterm Exam
General CS 1 (320101)

October 30, 2007

LAST NAME(s):
FIRST NAME(s):
MATRICULATION NUMBER:

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

You can reach 28 points if you solve all problems. You will only need 25 points for a perfect score, i. e. 3 points are bonus points.

You have ample time (60 minutes), so take it slow and avoid rushing to mistakes!

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

To be used for grading, do not write into this box										
prob.	1.1	1.2	1.3	2.1	2.2	3.1	3.2	4.1	Sum	grade
total	2	3	3	4	3	6	3	4	28	
reached										

1 Mathematical Foundations

2pt 3min

Problem 1.1: Fill in the blanks in the table of Greek letters. Note that capitalized names denote capital Greek letters.

Symbol	Δ	χ	Ψ	κ				
Name					Omega	mu	Theta	omicron

3pt 9min

Problem 1.2 (Properties of Relations)
Let R and S be (non empty) relations on some given set A. Prove or refute each of the three statements

1. If R is reflexive then R^{-1} is reflexive
2. If R and S are transitive then $R \cup S$ is transitive
3. If R is symmetric then all subsets of R are symmetric

3pt 6min

Problem 1.3 (An induction with fractions)
Using induction, prove that for all $n \in (\mathbb{N} \backslash \{0\})$, the following equivalence holds:

$$u_n := \sum_{i=1}^{n} \frac{1}{i(i+1)} = \frac{n}{n+1}$$

.

2 Abstract Data Types and Abstract Procedures

4pt 8min

Problem 2.1 (ADT for rational numbers)
Define an abstract data type for rational numbers. Write the numbers $\frac{2}{3}$ and $-\frac{1}{2}$ using your definition. Also define all other data types you need to construct rationals.

Note: Make sure that there is only one representation for the integer 0!

A proper data type for rational numbers would not be allowed to have more than one representation for one value. As reducing fractions is a bit out of scope here, you can ignore this problem and permit e. g. two distinct representations for $\frac{2}{4}$ and $\frac{1}{2}$, although these rational numbers are actually equal.

3pt 7min

Problem 2.2: Given the abstract data type

$$\langle \{\mathbb{A}, \mathbb{B}\}, \{[a\colon \mathbb{A}], [b\colon \mathbb{B}], [f\colon \mathbb{A} \to \mathbb{A}], [g\colon \mathbb{A} \times \mathbb{B} \to \mathbb{B}], [h\colon \mathbb{A} \times \mathbb{B} \to \mathbb{A}]\} \rangle$$

Which of the following mappings are substitutions?

- $\sigma_1 := [f(x_{\mathbb{A}})/x_{\mathbb{A}}], [b/y_{\mathbb{B}}]$
- $\sigma_2 := [g(a,b)/x_{\mathbb{A}}], [g(a,b)/y_{\mathbb{B}}], [f(x_{\mathbb{A}})/z_{\mathbb{A}}]$
- $\sigma_3 := [f^{i+1}(x_{\mathbb{A}})/f^i(x_{\mathbb{A}})]$, $i \in \mathbb{N}$ with $f^0(x_{\mathbb{A}}) = x_{\mathbb{A}}$ and $f^{i+1}(x_{\mathbb{A}}) = f(f^i(x_{\mathbb{A}}))$

Note: $f^k(x)$ here just means f applied to x for k times consecutively, i. e. $\underbrace{f(f(\ldots f}_{k \text{ times}}(x)))$.

---

3 Programming in Standard ML

Problem 3.1 (Calculate call charges)

6pt
12min

A telephone company computes the call charges in the following way:

1. price a is paid in the first minute
2. price b is paid in the next nine minutes
3. price c is paid for the remaining time

Example: for a call duration of 14 minutes the costumer pays $a + 9b + 4c$. Call charges vary for different destinations.

Given a list of tuples containing destinations (strings) and call charges (list with 3 int numbers) write a function that computes the net cost of a call. The function takes the input in the format `charge(destination:string, duration:int, the list described above)`.

Example:

```
charge("Bremen", 15, [("Bremen",   [3, 2, 1]),
                      ("Bogdanci", [5, 4, 2]),
                      ("Kolkata",  [8, 7, 5])]);
→ val it = 26 : int
```

The functions should raise an exception for an invalid duration or an invalid destination.

Note: The function should work for arbitrary tariff lists; the one above is given as an example only!

If one destination occurs more than once in the list, you can just take the first one you find and ignore subsequent ones.

Problem 3.2 (Cartesian and cascading functions)

3pt
6min

Write an SML function `curry` that takes a binary Cartesian function and returns the cascading variant, and a function `descartes` that does the inverse.

4 Codes

Problem 4.1 (A Morse string code)

4pt
8min

Given the alphabets $A = \{\texttt{J}, \texttt{A}, \texttt{C}, \texttt{O}, \texttt{B}, \texttt{S}\}$ and $B = \{., -\}$, consider the following restriction μ' of the Morse code to A:

J	.---
A	.-
C	-.-.
O	---
B	-...
S	...

Answer the following questions, always by proving, giving a counter-example, or referring to a definition or theorem in the lecture:

1. Is the code μ' a character code?
2. Is μ' a string code?
   (a) If it is not a string code, suggest a change by modifying at most one codeword (without extending B for that!) and prove that the resulting code μ'' is a string code.