Name: Matriculation Number:

Final Exam
General CS I (320101)

December 16, 2013

You have two hours(sharp) for the test;
Write the solutions to the sheet.

The estimated time for solving this exam is 114 minutes, leaving you 6 minutes for revising your exam.

You can reach 112 points if you solve all problems. You will only need 100 points for a perfect score, i.e. 12 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	2.1	3.1	3.2	4.1	5.1	5.2	6.1	6.2	Sum	grade
total	10	15	25	15	5	8	10	12	12	112	
reached											

Please consider the following rules; otherwise you may lose points:

- "Prove or refute" means: If you think that the statement is correct, give a formal proof. If not, give a counter-example that makes it fail.
- Always justify your statements. Unless you are explicitly allowed to, do not just answer "yes" or "no", but instead prove your statement or refer to an appropriate definition or theorem from the lecture.
- If you write program code, give comments!

1 Mathematical Foundations

Problem 1.1 (Geometric Properties)

Find a general formulae (in n) for the geometric quantities below and prove them by induction. 10pt 10min

1. the sum of the angles of a (convex) polygon with n sides.
2. the number of diagonals of a (convex) polygon with n sides.
3. the (maximum) number of areas n (infinite) lines divide the plane in

---

Solution:

1. $180°(n-2)$
2. $\frac{n(n-3)}{2}$
3. $\frac{n(n+1)}{2}+1$

Problem 1.2 (Function properties)

Prove or refute the following: 10pt 10min

1. If $f,g\colon \mathbb{R}\to\mathbb{R}$ are injective, then $f\circ g$ is injective.
2. If $f,g,h\colon \mathbb{R}\to\mathbb{R}$ and $f\circ g\circ h$ is surjective, then f is surjective.
3. If $f,g\colon \mathbb{Z}\to\mathbb{Z}$ are injective, $f\cdot g$ (function multiplication) is surjective.

Solution:

---

2 Abstract Data Types and Abstract Procedures

Problem 2.1 (ADT for Fibonacci numbers)

The Fibonacci numbers are the sequence of numbers defined by the linear recurrence equation: 15pt 8min

$$F_n = F_{n-1} + F_{n-2} \text{ with } F_0 = 0 \text{ and } F_1 = 1$$

1 Write down a datatype $\mathbb{F}$ for Fibonacci numbers. Write down the Fibonacci numbers 0, 5 and 13 in your representation.

2 Write down three abstract procedures that compute the n^{th} Fibonacci number and have the following data types:

- $\langle \text{nthFibVal}{::}\mathbb{N}\to\mathbb{N}\,;\,\ldots\rangle$
- $\langle \text{nthFib}{::}\mathbb{N}\to\mathbb{F}\,;\,\ldots\rangle$
- $\langle \text{ValOfFib}{::}\mathbb{F}\to\mathbb{N}\,;\,\ldots\rangle$

You can consider the addition on unary natural numbers as given:
$\langle \text{add}::\mathbb{N} \times \mathbb{N} \to \mathbb{N}\,;\, \{\text{add}(n, o) \rightsquigarrow n, \text{add}(n, s(m)) \rightsquigarrow s(\text{add}(n, m))\}\rangle$

Solution: $\langle\{\mathbb{F}\}, \{[\text{one}\colon \mathbb{F}], [\text{nextfib}\colon \mathbb{F} \to \mathbb{F}]\}\rangle$

$0 = \text{one}$

$5 = \text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{one})))))$

$21 = \text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{nextfib}(\text{one})))))))$

$\langle \text{nthFibVal}::\mathbb{N} \to \mathbb{N}\,;\, \text{nthFibVal}(o) \rightsquigarrow s(o), \text{nthFibVal}(s(o)) \rightsquigarrow s(o), \text{nthFibVal}(s(s(n))) \rightsquigarrow \text{add}(\text{nthFibVal}(s(n))$

$\langle \text{nthFib}::\mathbb{N} \to \mathbb{F}\,;\, \{\text{nthFib}(o) \rightsquigarrow \text{one}, \text{nthFib}(s(n)) \rightsquigarrow \text{nextfib}(\text{nthFib}(n))\}\rangle$

$\langle \text{ValOfFib}::\mathbb{F} \to \mathbb{N}\,;\, \text{ValOfFib}(\text{one}) \rightsquigarrow s(o), \text{ValOfFib}(\text{nextfib}(\text{one})) \rightsquigarrow s(o), \text{ValOfFib}(\text{nextfib}(\text{nextfib}(n))) \rightsquigarrow \text{ad}$

Problem 2.2 (Substitutions)

For the purposes of this problem we disregard sorts in constructor terms and assume a suitable abstract data type. 7pt

5min

1. Apply the substitutions

$$\sigma := [f(a)/x], [g(a, f(a), b)/z], \qquad \tau := [a/x], [h(b, c)/y], [c/z]$$

to the terms $s := g(x, f(y), z)$ and $t := h(f(x), g(y, z, x))$

2. Prove or refute that substitution composition is commutative (i.e. $\sigma_1 \circ \sigma_2 = \sigma_2 \circ \sigma_1$).

Solution:

1. Applying the substitutions:

$$\begin{aligned}
\sigma(s) &= g(f(a), f(y), g(a, f(a), b)) \\
\sigma(t) &= h(f(f(a)), g(y, g(a, f(a), b), f(a))) \\
\tau(s) &= g(a, f(h(b, c)), c) \\
\tau(t) &= h(f(a), g(h(b, c), c, a))
\end{aligned}$$

2. To refute commutativity we apply the two substitutions in different order to the term s:

$$\sigma(\tau(s)) = g(a, f(h(b, c)), c) \neq g(f(a), f(h(b, c)), g(a, f(a), b)) = \tau(\sigma(s))$$

Obviously the two expressions are different.

---

3 Programming in Standard ML

Problem 3.1 (Partitions and Sums)

20pt

20min

1. Design an SML function that takes an integer k and a list L and returns a list containing all the sublists of L of length k (i.e. the set of subsets of cardinality k if L interpreted as a set). Signature and example:

```
val select = fn : int * 'a list −> 'a list list
− select(2,[1,2,3,4]);
val it = [[1,2],[1,3],[1,4],[2,3],[2,4],[3,4]] : int list list
```

2. Now design an SML function which takes as argument an integer k and a list L containing integers. The function returns a list of all possible sums of k elements in the list. The result should not contain duplicate integers (each possible sum should be only once in the list). Signature and example:

```
val sums = fn : int * int list −> int list
 − sums(2,[1,2,3,4]);
val it = [3,4,5,6,7] : int list
```

Explanation: $3 = 1 + 2$, $4 = 1 + 3$, $5 = 1 + 4 = 2 + 3$, $6 = 2 + 4, 3 + 3$, $7 = 3 + 4$.

Hint: You can use the select function that you defined earlier (even if you didn't succeed in that task).

Solution:

```
Control.Print.printLength := 1000;

fun select(0,ls) = [[]]
  | select(k,[]) = []
  | select(k,a::ls) = (map (fn x=>a::x) (select(k−1,ls)))@select(k,ls);

fun member(a,[]) = false
  | member(a,x::ls) = if a=x then true else member(a,ls);

fun distinct([]) = []
  | distinct (a::ls) = if member(a,ls) then distinct(ls) else a::distinct(ls);

fun sum(ls) = foldl op+ 0 ls;

fun sums(k,ls) = distinct(map sum (select(k,ls)));
```

Problem 3.2 (Ways of Splitting)

15pt

15min

Two of the most powerful (and evil) civilizations have joined hands. Alexander the Great and Genghis Khan have reached an agreement that they will take over the world. But before conquering, they want to distribute the lands between them so that they both get equal areas of lands. They don't have to conquer everything. Given the areas of both empires and the list of areas of the territories between them, find the **number of ways** they can divide the lands such that in the end they both have the same area of their newly expanded empires.

Let A be the area of the territory of Alexander the Great, G the area of the territory of Genghis Khan and L the list of the areas of all the territories they want to conquer. Write a SML function split that takes $|A-G|$ as its first argument and L as its second argument.

```
val split = fn : int * int list -> int
```

Note: We round the areas to their integer values.

Hint: Think of what might happen to the difference when one region is being considered by Genghis Khan or Alexander the Great or none.

```
split(0,[]);
val it = 1 : int

split(2,[1]);
val it = 0 : int

split(1,[1,2,3]);
val it = 3 : int
```

Solution:

```
fun split (0, nil) = 1
  | split (n, nil) = 0
  | split (n, h::t) = split(n-h,t) + split (n+h,t) + split (n,t);
```

---

4 Formal Languages and Codes

Problem 4.1 (Formal Languages and Concatenation and Intersection)

Given two formal languages $L_1 := \{01^{[n]} \,|\, n \in \mathbb{N}\}$ and $L_2 := \{21^{[n]}0^{[n]} \,|\, n \in \mathbb{N}\}$ over the alphabet $A := \{0,1,2\}$: 5pt

5min

1. Write down $L_3 = \mathrm{conc}(L_1, L_2)$.
2. Give a language L_4 whose intersection with L_3 is empty.

Solution:

1. $L_3 = \{01^{[n]}21^{[k]}0^{[k]} \,|\, n \in \mathbb{N}, k \in \mathbb{N}\}$
2. $L_4 = \{1\}$ for example.

Problem 4.2 (String code)

Given the following code from $\{a, v, c, t, o, n, s, i\}$ to $B = \{0,1\}$. 5pt

5min

a	v	c	t	o	n	s	i
00	10	110	0110	0100	1000	111	1100

1. Encode the string `vacations` with this code?
2. Now try decoding the string you got in the previous part. Is the given it a string code? Explain. If it is not a string code alter it (without changing B) so that it becomes a prefix code.

Solution:

1. 10 00 110 00 0110 1100 0100 1000 111
2. It is not a string code. `n` = 1000 = `va`. Make it a prefix code by e.g. changing `v` to 101 and `c` to 1101.

---

5 Boolean Algebra

Problem 5.1 (Boolean Algebra)

Simplify the following expressions using boolean equivalences. Please briefly mention which identity you are using in each step. 5pt

5min

1. $\overline{x} * (x + y) + (y + x * x) * (x + \overline{y})$
2. $(\overline{x * y} * (\overline{x} + y)) * (\overline{y} + y)$

Solution:

1.

formula	identity
$(\overline{x} * x + \overline{x} * y) + (y + x * x) * (x + \overline{y})$	distributivity
$(\overline{x} * x + \overline{x} * y) + (y + x) * (x + \overline{y})$	identity
$(\overline{x} * x + \overline{x} * y) + (y + x) * (\overline{y} + x)$	commutativity
$(\overline{x} * x + \overline{x} * y) + x$	combining
$\overline{x} * y + x$	opposite

2.

formula	identity
$((\overline{x} + \overline{y}) * (\overline{x} + y)) * (\overline{y} + y)$	deMorgan
$(\overline{x} + \overline{y}) * (\overline{x} + y)$	identity addition
$\overline{x}$	combining

Problem 5.2 (Practising Quine McCluskey)

Use the algorithm of Quine McCluskey and explain the intermediate steps to determine 7pt

10min

the minimal polynomial of the following function.

$x1$	$x2$	$x3$	$x4$	f
F	F	F	F	T
F	F	F	T	T
F	F	T	F	T
F	F	T	T	T
F	T	F	F	F
F	T	F	T	F
F	T	T	F	F
F	T	T	T	T

$x1$	$x2$	$x3$	$x4$	f
T	F	F	F	T
T	F	F	T	T
T	F	T	F	F
T	F	T	T	F
T	T	F	F	F
T	T	F	T	T
T	T	T	F	F
T	T	T	T	T

State the cost of the solution.

Solution:

QMC_1 :

$$\begin{array}{rcl}
M_0 & = & \overline{x_1}\,\overline{x_2}\,\overline{x_3}\,\overline{x_4}, \overline{x_1}\,\overline{x_2}\,\overline{x_3}\,x_4, \overline{x_1}\,\overline{x_2}\,x_3\,\overline{x_4}, \overline{x_1}\,\overline{x_2}\,x_3\,x_4, \overline{x_1}\,x_2\,x_3\,x_4, x_1\,\overline{x_2}\,\overline{x_3}\,\overline{x_4}, x_1\,\overline{x_2}\,\overline{x_3}\,x_4, x_1\,x_2\,\overline{x_3}\,x_4, x_1\,x_2\,x \\
M_1 & = & \overline{x_1}\,\overline{x_2}\,\overline{x_3}, \overline{x_1}\,\overline{x_2}\,\overline{x_4}, \overline{x_2}\,\overline{x_3}\,\overline{x_4}, \overline{x_1}\,\overline{x_2}\,x_4, \overline{x_2}\,\overline{x_3}\,x_4, \overline{x_1}\,\overline{x_2}\,x_3, x_1\,\overline{x_2}\,\overline{x_3}, x_1\,\overline{x_3}\,x_4, \overline{x_1}\,x_3\,x_4, x_2\,x_3\,x_4, x_1\,x_2\,x_4 \\
P_1 & = & \emptyset \\
M_2 & = & \{\overline{x_1}\,\overline{x_2}, \overline{x_2}\,\overline{x_3}\} \\
P_2 & = & \{\overline{x_1}\,x_3\,x_4, x_1\,\overline{x_3}\,x_4, x_1\,x_2\,x_4\} \\
M_3 & = & \emptyset \\
P_3 & = & \{\overline{x_1}\,\overline{x_2}, \overline{x_2}\,\overline{x_3}\}
\end{array}$$

QMC_2 :

	FFFF	FFFT	FFTF	FFTT	FTTT	TFFF	TFFT	TTFT	TTTT
$\overline{x_1}\,\overline{x_2}$	T	T	T	T	F	F	F	F	F
$\overline{x_2}\,\overline{x_3}$	T	T	F	F	F	T	T	F	F
$\overline{x_1}\,x_3\,x_4$	F	F	F	T	T	F	F	F	F
$x_1\,\overline{x_3}\,x_4$	F	F	F	F	F	F	T	T	F
$x_1\,x_2\,x_4$	F	F	F	F	F	T	F	T	T

Final result: There are two solutions, both having optimal cost 14:

1. $f = \overline{x_1}\,\overline{x_2} + \overline{x_1}\,x_3\,x_4 + x_1\,\overline{x_3}\,x_4 + x_1\,x_2\,x_4$
2. $f = \overline{x_1}\,\overline{x_2} + \overline{x_2}\,\overline{x_3} + \overline{x_1}\,x_3\,x_4 + x_1\,x_2\,x_4$

---

6 Propositional Logic

Problem 6.1 (Natural Deduction)

Use the propositional natural deduction calculus to prove the Hilbert-Calculus axioms $P \Rightarrow Q \Rightarrow P$, and $(P \Rightarrow Q \Rightarrow R) \Rightarrow (P \Rightarrow Q) \Rightarrow P \Rightarrow R$ 10pt 10min

---

Solution:

$$
\dfrac{\dfrac{\begin{array}{c}[P]^1\\ [Q]^2\\ P\end{array}}{Q \Rightarrow P}\;{\Rightarrow}I^2}{P \Rightarrow Q \Rightarrow P}\;{\Rightarrow}I^1
\qquad\qquad
\dfrac{\dfrac{\dfrac{\dfrac{\dfrac{\begin{array}{c}[P \Rightarrow Q \Rightarrow R]^1\\ [P \Rightarrow Q]^2\\ [P]^3\\ Q\end{array}}{Q \Rightarrow R}\;{\Rightarrow}E}{R}\;{\Rightarrow}E}{P \Rightarrow R}\;{\Rightarrow}I^3}{(P \Rightarrow Q) \Rightarrow P \Rightarrow R}\;{\Rightarrow}I^2}{(P \Rightarrow Q \Rightarrow R) \Rightarrow (P \Rightarrow Q) \Rightarrow P \Rightarrow R}\;{\Rightarrow}I^1
$$

Problem 6.2 (Tableau proof)

Use the tableau method to prove $(P \Rightarrow Q) \Rightarrow (R \Rightarrow P) \Rightarrow R \Rightarrow Q$. 5pt

Solution: 5min

$$
\begin{array}{c}
(P \Rightarrow Q) \Rightarrow (R \Rightarrow P) \Rightarrow R \Rightarrow Q^{\mathsf{f}}\\
P \Rightarrow Q^{\mathsf{t}}\\
(R \Rightarrow P) \Rightarrow R \Rightarrow Q^{\mathsf{f}}\\
R \Rightarrow P^{\mathsf{t}}\\
R \Rightarrow Q^{\mathsf{f}}\\
R^{\mathsf{t}}\\
Q^{\mathsf{f}}\\
\begin{array}{c|c}
\begin{array}{c} R^{\mathsf{f}}\\ \bot \\ \ \end{array} &
\begin{array}{c} P^{\mathsf{t}}\\ \begin{array}{c|c} P^{\mathsf{f}} & Q^{\mathsf{t}}\\ \bot & \bot \end{array}\end{array}
\end{array}
\end{array}
$$
