Free University of Bozen-Bolzano – Faculty of Computer Science
Master of Science in Computer Science
Theory of Computing – A.Y. 2008/2009
Final exam – 30/1/2009 – Part 1

Time: 90 minutes

This is a closed book exam: the only resources allowed are blank paper, pens, and your head. Explain your reasoning. Write clearly, in the sense of logic, language and legibility. The clarity of your explanations affects your grade. Write your name and ID on every solution sheet. Good luck!

Problem 1.1 [6 points] You must give a brief explanation of your answer to receive full credit.

(a) Decide whether the following statement is TRUE or FALSE: The complement of a recursive-enumerable language is recursively-enumerable.

(b) Let M_2 be a 2-tape (deterministic) TM, and let M_1 be the result of converting M_2 into a (1-tape deterministic) TM. Sketch using two to three sentences the main idea underlying the conversion. How are the running times of M_1 and M_2 related to each other?

(c) Decide whether the following statement is TRUE or FALSE: For all languages L_1, L_2, and L_3, if there exist a reduction from L_1 to L_3 and a reduction from L_2 to L_3, then there exists a reduction from L_1 to L_2.

Problem 1.2 [6 points] Construct a TM M that accepts the language $L = \{n\#w \mid n$ is a number represented in binary with the least significant digit on the *right*, and $w \in \{\mathtt{a}, \mathtt{b}, \mathtt{c}\}^*$ with $|w|_\mathtt{a} + |w|_\mathtt{b} = n\}$, where $|w|_x$ denotes the number of occurrences of x in w.

E.g.: `10#accbc` $\in L$, `0#` $\in L$, `10#accbcb` $\notin L$ `10#ccac` $\notin L$.

Show the sequence of IDs of M on the input strings "`10#acbc`" and "`10#cb`".

Problem 1.3 [6 points] The *extraction* $L_1 \ominus L_2$ of two languages L_1 and L_2 is defined as:

$$L_1 \ominus L_2 = \{vw \mid vw_2w \in L_1, \text{ for some } w_2 \in L_2\}$$

Show that the class of recursively enumerable languages is closed under the *extraction* operation, i.e., that if L_1 and L_2 are recursively enumerable, then so is $L_1 \ominus L_2$.

[*Hint*: Show how to construct, from two (deterministic) TMs M_1 accepting L_1 and M_2 accepting L_2, a (possibly multi-tape) non-deterministic TM N accepting $L_1 \ominus L_2$. You need not detail completely the construction of N, but you should provide sufficient information to make clear that you would be able to detail the construction, if you were given enough time to do so.]

Problem 1.4 [6 points]

(a) Let f and g be primitive recursive functions. Show that the following predicate p is primitive recursive:

$$p(x) = \begin{cases} 1 & \text{if } f(i) > g(j), \text{ for all } 1 \leq i \leq x \text{ and } 1 \leq j \leq x \\ 0 & \text{otherwise} \end{cases}$$

(b) Show that the following function f is primitive recursive:

$$f(x) = \begin{cases} 2 & \text{if } x = 0 \text{ or } x = 1 \\ 3 \cdot f(x-1) \cdot f(x-2) & \text{if } x \geq 2 \end{cases}$$

Problem 1.5 [6 points]

(a) Let f be a total number-theoretic function with $n+1$ variables. Provide the definition of the $(n+1)$-variable function gn_f such that $gn_f(\vec{x}, y)$ encodes the values of $f(\vec{x}, i)$ for $1 \leq i \leq y$.

(b) Let g and h be total number-theoretic functions, respectively with n and $n+2$ variables. Define the $(n+1)$-variable function f obtained from g and h by *course-of-values recursion*.

Solutions to the TOC exam of 30/1/2003, Part 1

E7.2

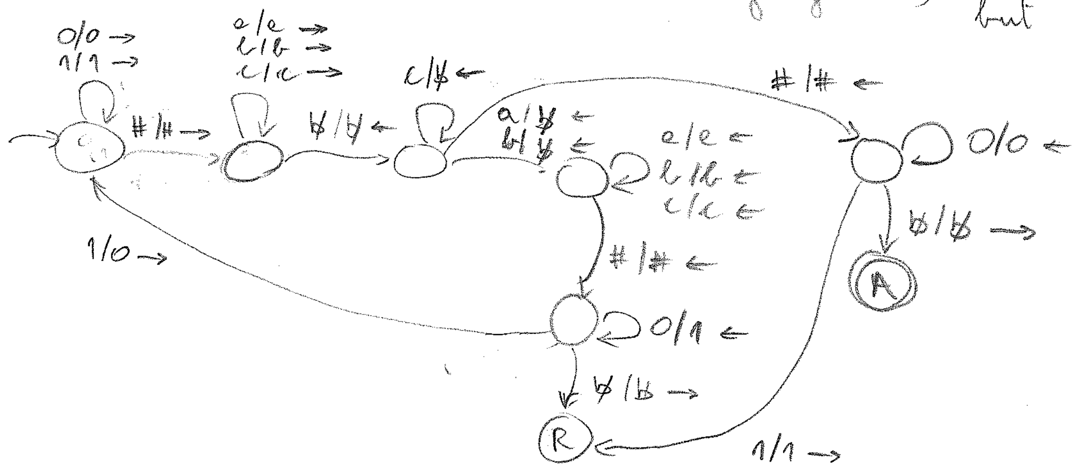
1.1 a) FALSE. Consider, e.g. L_u

b) M_1 has 4 tracks, 2 for the 2 tapes, 2 with a marker for the 2 head positions. For each move of M_2, one scan back and forth of M_1.
M_1 has quadratic running time in the running time of M_2

c) FALSE: e.g. L_1 a RE language
L_2 a REC language
L_3 a non RE language

$L_1 < L_3$
$L_2 < L_3$
but $L_1 \not< L_2$

1.2.



1.3. N is a 3-tape NTM working as follows, when given an input string x on tape 1:

1) guess a prefix v of x and copy it to tape 3
2) guess an arbitrary string w_2 on tape 2
3) Copy w_2 to tape 3 immediately after v
4) Run M_2 on w_2 on tape 2
If M_2 accepts, then proceed.
If M_2 rejects or loops, then this non-deterministic run of N will also reject or loop

5) Copy the remaining part w of x from tape 1 to tape 3, immediately after w_2. Tape 3 now contains $v w_2 w$.
6) Run M_1 on $v w_2 w$, and accept if M_1 accepts.
Otherwise, this non-deterministic run of N will reject or loop.

E7.3

1.4 a) $p(x) = \prod_{i=1}^{x} \cdot \prod_{j=1}^{x} gt(f(i), g(j))$

Since f, g, gt are PRFs

the composition of PRFs is a PRF

the bounded product of a PRF is a PRF

we get that also p is a PRF.

b) We define an auxiliary function $h(x) = gn_1(f(x), f(x+1))$

$$\begin{cases} h(0) = gn_1(f(0), f(1)) = gn_1(2,2) = 2^3 \cdot 3^3 = 8 \cdot 27 = 216 \\ h(x+1) = gn_1(f(x+1), f(x+2)) = \\ \quad = gn_1(f(x+1), 3 \cdot f(x+1) \cdot f(x)) = \\ \quad = gn_1(dec(1, h(x)), 3 \cdot dec(1, h(x)) \cdot dec(0, h(x))) \end{cases}$$

Since gn_2 and dec are PRF, this is a definition of h by PR.

$f(x) = dec(0, h(x))$

Hence f is a PRF

1.5 a) $gn_f(\vec{x}, y) = \prod_{i=0}^{y} pr(i)^{f(\vec{x}, i)+1}$

b)

$$\begin{cases} f(\vec{x}, 0) = g(\vec{x}) \\ f(\vec{x}, y+1) = h(\vec{x}, y, gn_f(\vec{x}, y)) \end{cases}$$

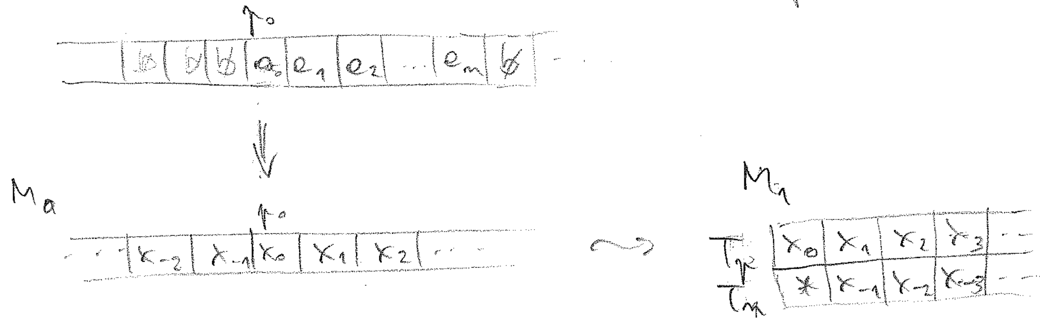
Exercises in preparation of Midterm Exam

Exercise 1: Consider a TM $M_0 = (Q_0, \Sigma, \Gamma_0, \delta_0, q_0, \not b, F_0)$.

Show that $\mathcal{L}(M_0)$ is also accepted by a TM M_1 that never moves left of its initial position (i.e., a by a TM with a semi-infinite tape).

Idea: M_1 is a two track TM: $M_1 = (Q_1, \Sigma, \Gamma_1, \delta_1, q_1, \not b, F_1)$

Let us call p_0 the initial tape position of M_0



- The states of M_1 are all the states of M_0, with an additional component P or N, indicating whether M_1 is currently working on the track representing the positive or negative portion of the tape of M_0: $Q_1 = (Q_0 \times \{P, N\}) \cup \{q_1, q_1'\}$

- Γ_1 is the set of pairs of symbols of Γ_0, plus symbols with $*$ on T_n

  $\Gamma_1 = \Gamma_0 \times (\Gamma_0 \cup \{*\})$

  The $*$ on T_1 is used to detect when M_1 reaches the leftmost tape position.

- Initially, Γ_1 writes $*$ on T_1 of the leftmost position

  $\delta(q_1, [\begin{smallmatrix} x \\ \not b \end{smallmatrix}]) = (q_1', [\begin{smallmatrix} x \\ * \end{smallmatrix}], R) \quad \forall x \in \Sigma$

  $\delta(q_1', [\begin{smallmatrix} x \\ \not b \end{smallmatrix}]) = ([q_0, R], [\begin{smallmatrix} x \\ \not b \end{smallmatrix}], L) \quad \forall x \in \Sigma$

- For the transitions of M_1, we need to distinguish 4 cases:

  1) M_0 is to the right of $p_0 \rightsquigarrow M_1$ works on track T_p
  2) " " left " " T_n
  3) M_0 is on $p_0 \rightsquigarrow M_1$ is on $[\begin{smallmatrix} - \\ * \end{smallmatrix}]$

Let $\delta_0(q, X) = (q', Y, d)$ be a transition of M_0 E7.5

Then we have

1) $\delta_1([q,P], [\begin{smallmatrix} X \\ Z \end{smallmatrix}]) = ([q',P], [\begin{smallmatrix} Y \\ Z \end{smallmatrix}], d)$ for every $Z \in \Gamma_0$ (i.e. $Z \neq *$)

2) $\delta_1([q,N], [\begin{smallmatrix} Z \\ X \end{smallmatrix}]) = ([q',N], [\begin{smallmatrix} Z \\ X \end{smallmatrix}], \bar{d})$ for every $Z \in \Gamma_0$

where $\bar{d} = L$ if $d = R$
$d = R$ if $d = L$

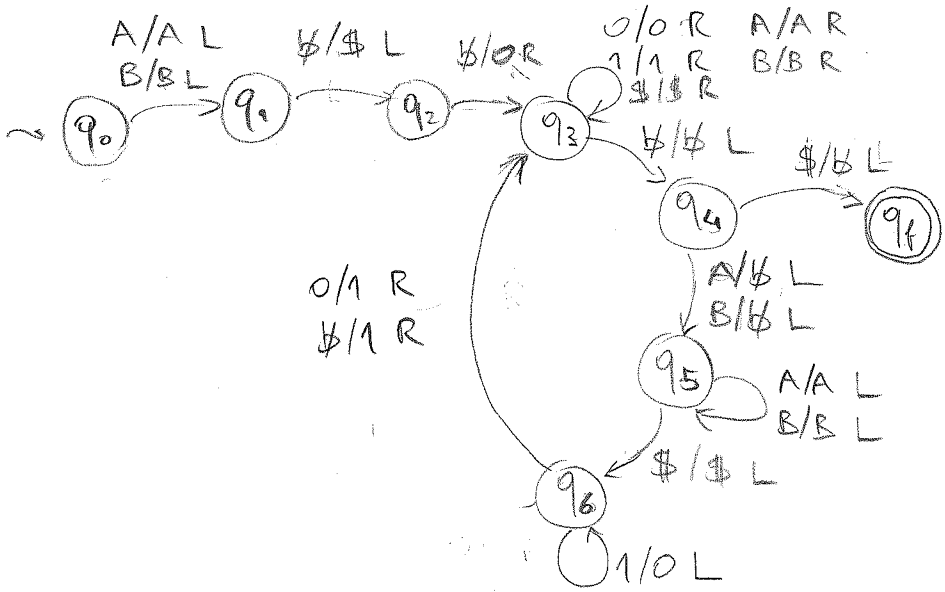
3) if M_0 moves right, i.e $d = R$
$\delta_1([q,-], [\begin{smallmatrix} X \\ * \end{smallmatrix}]) = ([q',P], [\begin{smallmatrix} Y \\ * \end{smallmatrix}], R)$

if M_0 moves left, i.e $d = L$
$\delta_1([q,-], [\begin{smallmatrix} X \\ * \end{smallmatrix}]) = ([q',N], [\begin{smallmatrix} Y \\ * \end{smallmatrix}], R)$

- Final states of M_1 : $F_1 = F_0 \times \{P, N\}$

Exercise 2: Construct a TM that computes the length of its input string, represented as a binary number (with the least significant digit on the right). Assume $\Sigma = \{A, B\}$ (E7.6)

Idea: we write a counter to the left of the input separated by a \$.
We repeatedly move to the right of the input, delete the last symbol, come back and increment the counter



E 7.7

<u>Exercise 3</u>: For a TM M with input alphabet Σ, let $\langle M, w\rangle$ denote the encoding $\mathcal{E}(M)$ of M followed by input w.

Consider the language $L = \{\langle M, w\rangle \mid M$ when started on an input string w, eventually does three consecutive transitions in which it moves the head in the same direction$\}$

a) Show that L is recursively enumerable

b) Show that L is not recursive

a) We reduce L to L_u.

The reduction R is a TM that takes as input $\langle M, w\rangle$ and produces as output $R(\langle M, w\rangle) = \langle M', w\rangle$ such that

$$\langle M, w\rangle \in L \quad \text{iff} \quad \langle M', w\rangle \in L_u .$$

We describe how R has to transform $\mathcal{E}(M)$ to obtain $\mathcal{E}(M')$:

- R has to add to the states of M a second component that counts how many consecutive transitions M has made in the same direction:

  the values of the counter component are $-3, -2, -1, 1, 2, 3$

- the transitions of M are modified to update the counter:

  if $\delta(q, x) = (q', y, d)$ is a transition of M, then M' contains transitions $\delta'([q, c], x) = ([q', c'], y, d)$ for each $c \in \{-2, 1, 0, 1, 2\}$, with c' given by:

d \ c	-2	-1	0	1	2
L	-3	-2	-1	-1	-1
R	1	1	1	2	3

- the final states of M' are exactly those where the counter is ± 3, i.e. $F' = Q \times \{3, -3\}$.

E7.8

b) We reduce the halting problem L_H to L

The reduction R is a TM that takes as input $\langle M, w\rangle$ and produces as output $R(\langle M, w\rangle) = \langle M', w\rangle$ such that

$$\langle M, w\rangle \in L_H \quad \text{iff} \quad \langle M', w\rangle \in L$$

We describe how R has to transform $\delta(M)$ to obtain $\delta(M')$:

- the final states of M are made non-final in M'
- from a final or blocking state of M we add to M' a transition to a new state from which M' makes 3 transitions to the right
- we have to make sure that M' never does 3 consecutive transitions in the same direction (except the ones above):

  Hence:

  if M does an R-move, then
  M' does an R-L-R move

  if M does an L move, then
  M' does an L-R-L move

- the tape symbol is changed only in the first of the three moves, while the other two leave the tape unchanged
- for the dummy moves, additional states are needed, and these need to be distinct for each state of M.

Formally:

if $\delta(q, x) = (p, y, R)$
then
$\delta'(q, x) = (p'_R, y, R)$
$\delta'(p'_R, -) = (p''_R, -, L)$
$\delta'(p''_R, -) = (p, -, R)$

if $\delta(q, x) = (p, y, L)$
then
$\delta'(q, x) = (p'_L, y, L)$
$\delta'(p'_L, -) = (p''_L, -, R)$
$\delta'(p''_L, -) = (p, -, L)$

Exercise 4: Let $g(x)$ be a PRF.

a) Show that the following predicate is a PRF:

$$f(x,y) = \begin{cases} 1 & \text{if } g(i) < g(x) \text{ for all } 0 \leq i \leq y \\ 0 & \text{otherwise} \end{cases}$$

$$f(x,y) = \prod_{i=0}^{y} \mathrm{lt}(g(i), g(x))$$

b) Let f be defined by

$$f(x) = \begin{cases} 1 & \text{if } x = 0 \\ 2 & \text{if } x = 1 \\ 3 & \text{if } x = 2 \\ f(x-3) + f(x-1) & \text{if } x \geq 3 \end{cases}$$

- Give the values $f(4)$, $f(5)$, $f(6)$.

$$f(3) = f(0) + f(2) = 1 + 3 = 4$$
$$f(4) = f(1) + f(3) = 2 + 4 = 6$$
$$f(5) = f(2) + f(4) = 3 + 6 = 9$$
$$f(6) = f(3) + f(5) = 4 + 9 = 13$$

Show that f is a PRF.

We have that $f(y+1) = f(y-2) + f(y)$.

We introduce an auxiliary function h with

$$h(y) = [f(y), f(y+1), f(y+2)] = gn_2(f(y), f(y+1), f(y+2))$$

$$\begin{cases} h(0) = gn_2(f(0), f(1), f(2)) = gn_2(1,2,3) = 2^2 \cdot 3^3 \cdot 5^4 \\ h(y+1) = [f(y+1), f(y+2), f(y+3)] = \end{cases}$$

$$= [f(y+1), f(y+2), f(y) + f(y+2)] =$$

$$= [\mathrm{dec}(1, h(y)), \mathrm{dec}(2, h(y)), \mathrm{dec}(0, h(y)) + \mathrm{dec}(2, h(y))]$$

$$= gn_2(\cdots)$$

Hence h is PR. Then $f(y) = \mathrm{dec}(0, h(y))$ is also PR.