Universität Koblenz-Landau

FB 4 Informatik

Prof. Dr. Viorica Sofronie-Stokkermans
Dipl. Inform. Markus Bender

February 2, 2014

Exercises for
“Advances in Theoretical Computer Science”
Exercise sheet 14

The structure, content, type of exercises and distribution of points in this last exercise sheet give an indication about how the written exam(s) could be structured. Please note that the real exams will also contain other types of exercises. For preparing please look at all the exercises on the homework sheets which were discussed in the exercise session.

The submission of the solutions is not compulsory. If you want to submit your solutions, please do so until Friday, 7.02.2014 at 10:00. (Joint solutions prepared by up to three persons are allowed.) We can discuss the solutions in the Question/Answer session (date/time and place will be announced soon).

Please note that since this is only an example of how the written exam could look, we did not check if it is suitable for 120 minutes (but we will do so for the real exam). Therefore please do not draw any conclusion out of the time it took you to solve this exercise sheet.

1 Multiple choice (8+12+16+4 = 40p)

For each correct cross you get 2 points. For each wrong cross you lose 1 point. The minimum for each subtask is 0 points, i.e. you cannot carry negative points into the next subtask.

1.1 Register machines

Please indicate (by crossing the right box) whether the following statements are true or false:

In the LOOP program “loop x_1 do S end” the number of times S is executed might be influenced by the fact that S changes x_1.	true ☐ false ☐
The if-then-else instruction can be simulated with a WHILE program.	true ☐ false ☐
Every WHILE program terminates.	true ☐ false ☐
For every WHILE program there exists an equivalent GOTO program.	true ☐ false ☐

1.2 Primitive recursive functions, μ-recursive functions

Please indicate (by crossing the right box) whether the following statements are true or false:

For every $k \in \mathbb{N}$, the function $f : \mathbb{N}^k \to \mathbb{N}$ defined by $f(n_1, \ldots, n_k) = \prod_{i=1}^{k} n_i$ is primitive recursive.	true ☐ false ☐
Every total function is primitive recursive.	true ☐ false ☐
For every primitive recursive function there exists a LOOP program which computes it.	true ☐ false ☐
Every μ-recursive function is primitive recursive.	true ☐ false ☐
Every μ-recursive function is total.	true ☐ false ☐
Every WHILE computable function is μ-recursive.	true ☐ false ☐

1.3 Complexity theory

Please indicate (by crossing the right box) whether the following statements are true or false:

Assume f is a computable function. Then every language in DSPACE($f(n)$) is decidable.	true ☐ false ☐
Assume f is a computable function. Then NSPACE($f(n)$) $\subseteq$ NTIME($f(n)$).	true ☐ false ☐
Every problem in NP can be solved deterministically in exponential time.	true ☐ false ☐
Every NP-hard problem is in NP.	true ☐ false ☐
If there exists an NP-hard problem which is in P then P = NP.	true ☐ false ☐
Let L_1, L_2 be languages with $L_1 \preceq_{\mathsf{pol}} L_2$ and $L_2 \in$ P. Then $L_1 \in$ P.	true ☐ false ☐
3-CNF-SAT is NP-complete.	true ☐ false ☐
DNF-SAT = $\{w \mid w$ satisfiable propositional formula in DNF $\}$ is NP-complete.	true ☐ false ☐

1.4 Decidability

Please indicate (by crossing the right box) whether the following statements are true or false:

The question whether a Turing machine halts on a certain input is decidable.	true ☐ false ☐
The question whether a Turing machine halts on input 0 is undecidable.	true ☐ false ☐

2 Register machines ((4 + 6) + (2 + 4 + 4) = 20p)

Exercise 2.1

The following instructions will be considered to be GOTO instructions in this exercise:

$x_i := c$	$x_i := c \text{ op } x_j$	$\text{goto } l$	Here: x_i, x_j, x_k are registers
$x_i := x_j$	$x_i := x_j \text{ op } c$	$\text{if } x_i = 0 \text{ goto } l$	c is a constant
	$x_i := x_j \text{ op } x_k$		$\text{op} \in \{+, -, *\}$
			and l is a label.

A GOTO program has the form

$$l_1 : B_1, \ldots, l_k : B_k \quad (k \geq 1)$$

where $B_1, \ldots, B_k$ are GOTO instructions and $l_1, \ldots, l_k$ are labels.

Let $\mathsf{sqrt} : \mathbb{N} \to \mathbb{N}$ be defined by

$$\mathsf{sqrt}(n) = \lfloor \sqrt{n} \rfloor$$

(1) Write a pseudocode program for sqrt which only uses the arithmetical operators $+, -, *$ in assignments and only uses comparisons such as $\leq, <, \geq, >, =$ in tests or as loop/for/while conditions.

(2) Write a GOTO program which computes sqrt.

Exercise 2.2

Let P be the following GOTO program:

$$
\begin{array}{rl}
1: & x_5 := x_1; \\
2: & x_4 := x_2; \\
3: & \text{if } x_1 = 0 \text{ goto } 11; \\
4: & x_2 := x_4; \\
5: & \text{if } x_2 = 0 \text{ goto } 9; \\
6: & x_3 := x_3 + 1; \\
7: & x_2 := x_2 - 1; \\
8: & \text{if } x_6 = 0 \text{ goto } 5; \\
9: & x_1 := x_1 - 1; \\
10: & \text{if } x_6 = 0 \text{ goto } 3; \\
11: & x_2 := x_4; \\
12: & x_1 := x_5; \\
13: & x_4 := 0; \\
14: & x_5 := 0
\end{array}
$$

(1) Which value does P compute on input $x_1 = 2, x_2 = 3$?

(2) Which function $f : \mathbb{N}^2 \to \mathbb{N}$ is computed by P?

(3) Use the transformation presented in the lecture to construct a WHILE-IF program with only one WHILE loop which has the same semantics as P.

3 Recursive functions (((1+1+5) + (1+5)) + 7 = 20p)

In the next exercises we use the following notation:

- $\circ$ is function composition.
- if $j \leq k$, π_j^k is the projection function defined by $\pi_j^k(n_1, \ldots, n_k) = n_j$.
- $(+1) : \mathbb{N} \to \mathbb{N}$ is defined by: $(+1)(n) = n + 1$.
- $(-1) : \mathbb{N} \to \mathbb{N}$ is defined by: $(-1)(n) = \begin{cases} 0 & \text{if } n = 0 \\ n - 1 & \text{otherwise} \end{cases}$.
- $* : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is defined for all $n_1, n_2 \in \mathbb{N}$ by: $*(n_1, n_2) = n_1 * n_2$.
- $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is defined for all $n_1, n_2 \in \mathbb{N}$ by: $+(n_1, n_2) = n_1 + n_2$.
- $- : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ is defined for all $n_1, n_2 \in \mathbb{N}$ by $-(n_1, n_2) = n_1 - n_2 = \begin{cases} 0 & \text{if } n_1 \leq n_2 \\ n_1 - n_2 & \text{otherwise} \end{cases}$.
- for all $s, k \in \mathbb{N}$, $c_s^k : \mathbb{N}^k \to \mathbb{N}$ is defined for all $\mathbf{n} \in \mathbb{N}^k$ by: $c_s^k(\mathbf{n}) = s$.

In what follows we will assume known that all the functions above are primitive recursive.

Exercise 3.1

(1) Consider the following primitive recursive function:

$$f_1 = \mathcal{PR}[(+1) \circ (+ \circ (c_4^0, 0)), * \circ ((-1) \circ \pi_1^2, \pi_2^2)]$$

(a) Which is the arity of f_1?

(b) What does f_1 compute if all arguments are equal to 1?

(c) Which (concrete) function is computed by f_1?

(2) Consider the following μ-recursive function:

$$f_2 = \mu g, \text{ where } g(n, i) = \begin{cases} n + 2 & \text{if } i = 0 \\ \mu_j(j + 2 - n = 0) & \text{if } i = 1 \\ 2 & \text{if } i \geq 2 \end{cases}$$

(a) Which is the arity of f_2?

(b) Which is the (concrete) function computed by f_2?

Exercise 3.2

Let $f : \mathbb{N} \to \mathbb{N}$ be defined as follows:

$$f(n) = \begin{cases} 1 & \text{if } n = 0 \\ 2 & \text{if } n = 1 \\ f(n-1) * f(n-2) & \text{if } n \geq 2 \end{cases}$$

Prove that f is primitive recursive.
You can use any of the results proved in the lecture for this.

4 Decidability, Rice's theorem (4+6 = 10p)

Exercise 4.1

- State the theorem of Rice.
- Prove that the following problem is undecidable using the theorem of Rice:

$$L = \{n \mid M_n \text{ accepts only words which have length greater than } 10\}$$

5 Complexity (2+4+2+2 = 10p)

Exercise 5.1

A zoo acquires for the first time n animals $x_1, \dots, x_n$. A list L of enemies is provided, containing sets $\{x_i, x_j\}$ consisting of two animals which cannot be placed in the same cage. The zoo has k cages.

We are interested in the problem of deciding whether the animals can be placed in the zoo such that they are all safe. Thus,

$$\text{ZOO} = \{(L, n, k) \mid \begin{array}{l}\text{the } n \text{ animals with enemy list } L \text{ can be placed}\\ \text{on } k \text{ cages such that they are all safe.}\end{array} \}$$

(1) One of the two triples (L_i, n_i, k_i) is an instance of ZOO. Which is this?

$(\{\{1,2\},\{1,3\},\{2,3\},\{3,4\}\},4,3)$	☐
$(\{\{1,2\},\{1,3\},\{1,4\},\{2,3\},\{2,4\},\{3,4\}\},4,3)$	☐

(2) Let f be the function which associates with every undirected graph $G = (V, E)$ – where $V = \{v_1, \dots, v_m\}$ – the tuple (L_G, n_G, k_G) where:

$$L_G = E, n_G = m, k_G = 3$$

Prove that f defines a polynomial reduction of 3-colorability to *ZOO*.

(3) Is ZOO in NP? Briefly justify your answer (you do not need to construct a Turing machine for this).

(4) In the lecture we will study the k-colorability problem:

k-colorability $= \{G \mid G$ undirected graph which is colorable with at most k colors $\}$.

We know that the 3-colorability problem is an NP-complete problem.
Prove or refute the following: ZOO is an NP-complete problem.

The submission of the solutions is not compulsory. If you want to submit your solutions, please do so until 7.02.14, 10:00 s.t.. Joint solutions prepared by up to three persons are allowed. Please do not forget to write your name on your solution.
Submission possibilities:

- By e-mail to mbender@uni-koblenz.de with the keyword "Homework ACTCS" in the subject.
- Put it in the box in front of Room B 222.