Martin Ziegler
MiJin Kim

Issued on Sep.25, 2015
Solutions due: Oct.13, 2015

CS422

Fall 2015, Assignment #2

PROBLEM 4:
Formalize the following decision problems as subsets of $\{0,1\}^*$. Which of them are (i) decidable, (ii) semi-decidable but not decidable, (iii) not semi-decidable? Prove your answers!

a) Given (the source code in Python of) some algorithm $\mathcal{A}$, input $\vec{x}$, and an integer $N \in \mathbb{N}$, does $\mathcal{A}$ on input $\vec{x}$ terminate within N steps?

b) Given (the source code in Python of) some algorithm $\mathcal{A}$, does there exist some input $\vec{x}$ on which $\mathcal{A}$ does not terminate?

c) Given some source code $\mathcal{A}$ containing a function `virus()`, does there exist some input $\vec{x}$ that makes $\mathcal{A}$ invoke said function?

d) Given a multivariate polynomial $p(x_1,\ldots,x_n)$ with integer coefficients, does it have a complex root?

PROBLEM 5:
Recall that decision problem X is called *reducible* to Y (written $X \preccurlyeq Y$) if there exists a total computable function $f : \{0,1\}^* \to \{0,1\}^*$ such that, for all $\vec{x} \in \{0,1\}^*$, it holds: $\vec{x} \in X \Leftrightarrow f(\vec{x}) \in Y$.
Like the Halting and Totality problems, H and T, the following problems X and Y are undecidable:

X) Given an algorithm $\mathcal{A}$, does it 'ignore' its input in the sense that $\mathcal{A}$ either terminates for all $\vec{x}$ or for none?

Y) Given two algorithms $\mathcal{A}$ and $\mathcal{B}$, are they equivalent in the sense that, for every $\vec{x}$, $\mathcal{A}$ on input $\vec{x}$ eventually terminates iff $\mathcal{B}$ on input $\vec{x}$ does (although not necessarily after the same number of steps)?

a) Prove $T \preccurlyeq Y$.

b) Prove $T \preccurlyeq X$.

c) Prove $X \preccurlyeq T$.

PROBLEM 6:

a) Devise a `LOOP` program with two arguments n,m computing integer division $\lfloor n/(m+1) \rfloor$.

b) Devise a `LOOP` program with one argument n computing 2^n.

c) Have your program simulated* and record the running times for $n = 1,2,3,4,\ldots$.

d) Devise a `LOOP` program with argument n computing the exponential tower $2^{2^{\cdot^{\cdot^{2}}}}$ of height n.

*e.g. on `http://www.eugenkiss.com/projects/lgw/`