CECS 329, Learning Outcome Assessment 10, April 20th, Spring 2023, Dr. Ebert

NO NOTES, BOOKS, ELECTRONIC DEVICES, OR INTERPERSONAL COMMUNICATION ALLOWED. Submit solutions to at most 2 LO problems on separate sheets of paper.

Problems

LO6. Do the following.

(a) Compute the Gödel number for program $P = T(2,3), Z(4), J(1,2,4), S(1)$. Write your answer as a sum of powers of two minus 1.

(b) Provide the URM program P whose Gödel number equals

$$2^{79} + 2^{89} + 2^{104} + 2^{121} - 1.$$

LO7. Answer and solve the following.

(a) Which of the following is *not* needed by a universal program P_U in order to simulate the computation of program P_x on input y?

i. the maximum index of any register used by P_x

ii. the maximum number of configurations used in the computation of P_x on input y

iii. the number of instructions of P_x

iv. the length of each configuration used in the computation of P_x on input y

(b) A universal program P_U is simulating a program that has 253 instructions and whose Gödel number is

$$x = 2^{79} + 2^{102} + 2^{120} + 2^{129} + 2^{c_5} + \cdots + 2^{c_{253}} - 1.$$

If the current configuration of the computation of P_x on some input has encoding

$$\sigma = 2^2 + 2^4 + 2^{10} + 2^{19} + 2^{24} - 1,$$

then provide the next configuration of the computation *and* its τ encoding.

LO8. Do the following.

(a) In one or more complete sentences, describe what is asserted by the Church-Turing Thesis.

(b) Consider the function

$$g(x) = \begin{cases} 1 & \text{if } \phi_x \text{ is total} \\ 0 & \text{otherwise} \end{cases}$$

In other words, $g(x) = 1$ iff program P_x halts on all of its inputs. We want to prove that $g(x)$ is undecidable, meaning there is no URM program that computes g. To do this, let's assume that $g(x)$ is computable by some URM program G. Then define the function $f(x)$ as follows.

$$f(x) = \begin{cases} P_x(x) + 1 & \text{if } g(x) = 1 \\ 0 & \text{if } g(x) = 0 \end{cases}$$

Provide an informal description of the steps you would use to compute $f(x)$ using pencil and paper. Why does this imply that $f(x)$ is URM computable?

(c) Let e denote the Gödel number of the URM program F that computes $f(x)$ from part b. In other words $F = P_e$. Show that a contradiction arises when we try to compute $f(e)$ using F.

LO9. An instance of the decision problem `Odd Range` is a Gödel number x, and the problem is to decide if function ϕ_x has a range that consists only of odd natural numbers. Mathematically speaking, $g(x) = 1$ iff $E_x \subseteq \{1, 3, 5, \ldots\}$. Consider the function

$$g(x) = \begin{cases} 1 & \text{if } E_x \subseteq \{1, 3, 5, \ldots\} \\ 0 & \text{otherwise} \end{cases}$$

(a) Evaluate $g(x)$ for each of the following Gödel number's x. Note: 2 out of 3 correct is considered passing. **Justify your answers**.

i. $x = e_1$, where e_1 is the Gödel number of the program that computes the function $\phi_{e_1}(y) = y^3$.

ii. $x = e_2$, where e_2 is the Gödel number of the program that computes the function $\phi_{e_2}(y) = 3^y$.

iii. $x = e_3$, where e_3 is the Gödel number of the program that computes $g(x)$ (assuming that $g(x)$ is URM computable).

(b) Prove that $g(x)$ is not URM computable. In other words, there is no URM program that, on input x, always halts and either outputs 1 or 0, depending on whether or not ϕ_x has a range consisting of odd natural numbers. Do this by writing a program P that uses g and makes use of the `self` programming concept. Then show how P creates a contradiction.

LO10. Answer the following.

(a) Provide the definition of what it means to be a mapping reduction from decision problem A to decision problem B.

(b) For the mapping reduction f : `Subset Sum` $\to$ `Set Partition`, determine $f(S, t)$ for `Subset Sum` instance $(S = \{2, 10, 14, 21, 33, 38, 46\}, t = 61)$. Verify that (S, t) and $f(S, t)$ are both positive instances of their respective decision problems. Show work.