Theoretical Computer Science (M21276)

Part B/4: More undecidable problems
(Nov 20-24, 2023)

Question 1. Clearly state one undecidable problem. What are inputs/outputs of the problem? Give examples of input instances.

Question 2. Explain on an example of a polynomial equation with one variable/two variables why Hilbert's tenth problem is partially decidable.

Question 3. For each of the following instances of the Post correspondence problem, find a solution or state that no solution exists.

a) $\{(a, abbbbb), \ (bb, b)\}$ over the alphabet $\{a, b\}$

b) $\{(10, 100), (0, 01), (0, 00)\}$ over the alphabet $\{0, 1\}$

c) $\{(ab, a), (ba, b), (a, ba), (b, ab)\}$ over the alphabet $\{a, b\}$

d) $\{(1, 111), (0111, 0), (10, 0)\}$ over the alphabet $\{0, 1\}$

e) $\{(ab, aba), (ba, abb), (b, ab), (abb, b), (a, bab)\}$ over the alphabet $\{a, b\}$

Question 4. Show that for $|\Sigma| = 1$, the Post correspondence problem is decidable. That is there is an algorithm that can decide whether or not an input instance over a single-letter alphabet has an solution. Characterise the instances with a negative answer and in other cases describe how we can obtain a solution.

Question 5. Explain the difference between bounded and unbounded tiling problem. Why the bounded tiling problem is decidable? Why we can't use a solution for the bounded tiling problem to solve the unbounded tiling problem.

Question 6. Show how the solvability of the Halting problem would imply the solution of the conjecture about perfect numbers: n is perfect if it is equal to the sum of its divisors (except itself), e.g. $6 = 1 + 2 + 3$, 28, $\ldots$. There is the conjecture there are no odd perfect numbers.

Question 7. The number of all computable functions is countable (give a reason for that!). However, one student didn't realise that and tried to prove that it is uncountable! Find a problem in his arguments.
Suppose we have the following effective enumeration of all the computable functions that take a single argument:
$$f_0, f_1, f_2, \ldots, f_n, \ldots.$$
For each of the following functions g, explain what is wrong with the following diagonalisation argument claiming to show that g is a computable function that isn't in the list.

"Since the enumeration is effective, there is an algorithm to transform each n into the function f_n. Since each f_n is computable, it follows that g is computable. It is easy to see that g is not in the list. Therefore, g is a computable function that isn't in the list."

a) $g(n) = f_n(n) + 1$.

b) $g(n) = $ if $f_n(n) = 4$ then 3 else 4.

c) $g(n) = $ if $f_n(n)$ halts and $f_n(n) = 4$ then 3 else 4.

Question 8. Show that the following problem is decidable: Is there a computable function that, when given a computable function f, m, and k, can tell whether f halts on input m in k units of time?