COMP314 Final Exam Spring 2023

Time allowed: 3 hours

Total points on exam: 120

Question 1. (15 points) The language L on the genetic alphabet is defined by

$$L = \{\texttt{C}^p\texttt{A}^q\texttt{G}^r\texttt{T}^q \text{ such that } p \geqslant 5, q \geqslant 5, r \geqslant 5\}.$$

Prove that L is not regular.

Question 2. (20 points) Consider the search problem Increment5OnSome defined as follows. The input is a SISO Python program P. If there exists a positive integer M for which $P(M) = M + 5$, then M is a solution. If no such M exists, the solution is "`no`".

(Note: To keep our notation simple, the above definition ignored the distinction between integers and strings that represent integers. A more accurate description would state that `str(`M`)` is a solution if $P($`str(`M`))` = `str(`$M + 5$`)`. In your answer to this question, you can feel free to abuse notation, allowing M to simultaneously represent both an integer and its string representation.)

Prove that Increment5OnSome is uncomputable.

Question 3. (10 points) Give an example of a computational problem that is recognizable but not decidable. In two or three sentences of your own words, explain why your example is recognizable but not decidable. Rigorous mathematical proof is not required.

Question 4. Consider the decision problem PackTwoTrucks (P2T) defined as follows. P2T is similar to Packing, except we imagine simultaneously loading two trucks instead of one truck. The input consists of (i) a list of positive integer package weights separated by whitespace, then (ii) a semicolon followed by two positive integers L_1, H_1 separated by whitespace, then (iii) a semicolon followed by two positive integers L_2, H_2 separated by whitespace. We interpret L_1, H_1 as the low threshold and high threshold for the first truck, and L_2, H_2 as the thresholds for the second truck. So, the solution is "`yes`" if there is some subset S_1 of the packages whose weight sum is between L_1 and H_1 inclusive and another disjoint subset S_2 whose weight sum is between L_2 and H_2 inclusive. Note that S_1 and S_2 must be disjoint, because we cannot load the same package onto both trucks.

Important note: in this question, we insist that positive instances of *both* Packing and P2T have strictly positive weights and thresholds. Any instance with a zero or negative weight or threshold is automatically a negative instance. This condition is a little different to the textbook definition of Packing.

Examples of positive instances of P2T:

- "`5 5 2 2 20 ; 9 11 ; 3 5`"
- "`2 4 6 8 20 ; 10 10; 10 10`".

Examples of negative instances of P2T:

- "`5 5 2 2 20 ; 0 11 ; 0 5`" (because thresholds must be positive)
- "`2 4 6 8 20 ; 10 12; 14 16`"

The code below is intended to be a polyreduction from the decision variant of PACKING to P2T. Fill in the blanks to ensure this is a correct polyreduction.

```
def convertPackingToP2T(inString):
    (weightString, L, H) = inString.split(';')
    weights = [int(x) for x in weightString.split()]
    L = int(L); H = int(H)
    # define a negative instance of P2T so that we can return it if desired
    negP2Tinstance = _____ # (a) (5 points)
    # check that all weights and thresholds are strictly positive, and
    # return the negative instance if not
    if min(weights) < 1 or min(L, H) < 1:
        return negP2Tinstance
    # Add a new package, then force the second truck to be loaded with
    # just this new package.
    weights.append(_____) # (b) (3 points)
    L1 = _____ # (c) (3 points)
    H1 = _____ # (d) (3 points)
    L2 = _____ # (e) (3 points)
    H2 = _____ # (f) (3 points)
    weightString = [str(x) for x in weights]
    return ' '.join(weightString) + ';' + str(L1) + ' ' \
        + str(H1) + ';' + str(L2) + ' ' + str(H2)
```

Question 5. Both parts of this question refer to Alan Turing's 1936 *On Computable Numbers* paper.

(a) (5 points) Describe what Turing means by the word "computer" in *On Computable Numbers*.

(b) (10 points) In *On Computable Numbers*, the word "computable" can have various meanings. Briefly compare and contrast two of these meanings.

Question 6. Given an undirected graph G and nonnegative integer K, define a *Hamilton cycle with K repeats* to be the same thing as a Hamilton cycle, except that, instead of insisting that each node is visited exactly once, exactly K of the nodes must be visited exactly twice. All other nodes are visited exactly once, as with the usual definition of a Hamilton cycle. Edges may be visited any number of times.

Let KRepeatsHamCycle be a decision problem parameterized by a nonnegative integer K. The input is an undirected graph G. The solution is "`yes`" if G contains a Hamilton cycle with K repeats, and "`no`" otherwise.

(a) (5 points) Prove that KRepeatsHamCycle $\in$ NP.

(b) (10 points) Prove that KRepeatsHamCycle is NP-complete.

Question 7. (10 points) Using the algorithm described in the textbook, convert the following instance of Sat into an instance of 3Sat.

$$(\neg x_1 \vee \neg x_2 \vee x_5 \vee x_7) \wedge (x_1 \vee x_3 \vee \neg x_5 \vee x_7 \vee x_8)$$

Question 8. (15 points) Consider the decision problem NoLowerCase defined as follows. The input is a SISO Python program P. The solution is "`yes`" if P never produces any lowercase letters in its output, regardless of its input; if it is possible for $P(I)$ to contain a lowercase letter on input I, the solution is "`no`". Prove that NoLowerCase is unrecognizable. (Note that this question refers to *unrecognizability*, not *undecidability*.)

Total points on this exam: 120