**Assignment 8**
**CSCI-661 Foundations of Computer Science Theory**
**due Friday April 26, 2024**

1. **(5 points)**

   Let $\Sigma = \{a, b, c\}$. For every listed class of languages over $\Sigma$, you have to state whether the class is closed under complementation, concatenation, and intersection. Write "yes" in a box if the class is closed under the operation. Write "no" if the class is not closed under the operation.

| closed under | Complementation | Concatenation | Intersection |
|---|---|---|---|
| The co-Turing-recognizable languages | | | |
| The context-free languages | | | |
| The Turing-recognizable languages | | | |
| The decidable languages | | | |
| The regular languages | | | |

2. **(4 points)** Draw a Venn diagram that shows the relationships between the following classes of languages:

   (a) the CFLs.

   (b) the decidable languages.

   (c) the languages whose complement is Turing-recognizable.

   (d) the languages whose complement is decidable.

   (e) the regular languages.

   (f) the Turing-recognizable languages.

3. **(5 points)** An unrestricted grammar is like a CFG, except that you can now have any string in $(V \cup \Sigma)^* V (V \cup \Sigma)^*$ on the left-hand side of a rule. Surprisingly, unrestricted grammars generate exactly the Turing-recognizable languages. We will not show this, but we will look at an example.

   Given the following grammar, $G$:

- $S \to ABaC$
- $Ba \to aaB$
- $BC \to DC$
- $BC \to E$
- $aD \to Da$
- $AD \to AB$
- $aE \to Ea$
- $AE \to \epsilon$

(a) Let $L(G)$ be the language generated by $G$. Give a description of $L(G)$.

(b) Give a derivation of $aaaaaaaa$. It is ok to combine very similar steps and write $\Rightarrow^k$ when you are taking $k$ very similar steps.

4. Solve Exercise 5.3 from the textbook.

5. Consider the following problems:

$$\begin{aligned} ALL_{\text{CFG}} &= \{\langle G\rangle \mid G \text{ is a CFG and } L(G) = \Sigma^*\}, \\ EQ_{\text{CFG}} &= \{\langle G_1, G_2\rangle \mid G_1, G_2 \text{ are CFGs and } L(G_1) = L(G_2)\}. \end{aligned}$$

$ALL_{\text{CFG}}$ is undecidable (you don't have to show this).

$EQ_{\text{CFG}}$ is undecidable (you will show this in the second part of the problem).

(a) Consider the following faulty proof that $EQ_{\text{CFG}}$ is decidable. $M$ is a Turing Machine that supposedly decides $EQ_{\text{CFG}}$.

$M$ = *"On input $\langle G_1, G_2\rangle$, where $G_1$ and $G_2$ are CFGs:*

*i. Construct CFG $G_3$ such that $L(G_3) = (L(G_1) \cap \overline{L(G_2)}) \cup (L(G_2) \cap \overline{L(G_1)})$.*

*ii. Run TM $R$ from Theorem 4.8. on $\langle G_3\rangle$.*

*iii. If $R$ accepts,* accept*; otherwise,* reject*.*

What is wrong with the above argument that $EQ_{\text{CFG}}$ is decidable?

(b) Use the undecidability of $ALL_{\text{CFG}}$ to show that $EQ_{\text{CFG}}$ is also undecidable. Use Sipser notation. Alternatively, it is also acceptable to use the "box inside a box" visual description we used in class. For either approach, you must clearly indicate that you are doing a proof by contradiction, and at the conclusion of your argument indicate the contradiction you have reached.

6. The following language:

$$E_{TM} = \{< M > | M \text{ is a TM and } L(M) = \emptyset\}$$

is known to be undecidable.

Use this information to prove by contradiction that the following language is undecidable:

$$X = \{< M_1, M_2 > | M_1 \text{ and } M_2 \text{ are TMs and } L(M_1) \cap L(M_2) = \emptyset\}.$$