Problem 1 CS 181 HW 6

Suppose that $F, G = \{0,1\}^* \to \{0,1\}$ are context free.
For each one of the following definitions of the function H,
either prove that H is always context free or give a
counterexample of context free F, G that would make H not
context free.

② $H(x) = F(x) \wedge G(x)$

Proof: We want to show that context free grammar is not closed
under intersection

If we are given F, G are context free, but $F(x) \wedge G(x)$
may not be a context free

① Let $\mathcal{L}_1 = \{a^n b^m c^m : m, n \geq 0\}$

$\mathcal{L}_2 = \{a^n b^n c^m : m, n \geq 0\}$, both F & G are context free

② The intersection $\mathcal{L} = \{a^n b^n c^n : n \geq 0\}$ because $\mathcal{L}_1$ says that
the number of b's need to be equal to the number of
c's and $\mathcal{L}_2$ says that the number of a's needs to be
the same as the number of b's. Therefore, the intersection
of $\mathcal{L}_1$ and $\mathcal{L}_2$ require both conditions to be true which leads to

$\mathcal{L} = \{a^n b^n c^n : n \geq 0\}$

③ However, $\mathcal{L} = \{a^n b^n c^n : n \geq 0\}$ is not context free
by pumping lemma which is stated by the problem
specification.

If $c = a$, then we can specify the grammar as follows

① $\mathcal{L}_2 = \{a^n b^n a^m \mid n, m \geq 0\}$ is generated from this grammar

$S \rightarrow XA$

$X \rightarrow aXb \mid \varepsilon$

$A \rightarrow Aa \mid \varepsilon$

② $\mathcal{L}_1 = \{a^n b^m a^m \mid n, m \geq 0\}$ is generated from this grammar

$S \rightarrow AX$

$X \rightarrow aXb \mid \varepsilon$

$A \rightarrow Aa \mid \varepsilon$

$\mathcal{L} = \{a^n b^n a^n \mid n \geq 0\}$ is not context free

Problem 1

④ $H(x) = F(x^R)$ where x^R is the reverse of x

$\therefore x^R = x_{n-1} x_{n-2} \cdots x_0$ for $n = |x|$

Proof: We want to show the reverse of a CFG is also context free.

- Let $G = (V, T, P, S)$ = (Variable, Terminal, Rule, Start) be a context free grammar in CNF form such that $L = L(G)$
- Let $G' = (V, T, P', S)$ such that $L(G')$ = Reverse $(L(G))$
- According to CNF, every rule in G is either of the following forms:
  - ↳ $A \to BC$ variables $\in \{A, B, C\}$
  - ↳ $A \to a$ terminals $\in \{a\}$

- For $G' = (V, T, P', S)$ and P' are defined as follows:
  - ↳ Every rule in P of the form $A \to a$. is also contained in P'
  - ↳ For every rule in P of the form $A \to BC$, the rule $A \to CB$ is contained in P'

Here is an example if P, the rules contain

$S \to aSb \mid ab$; $G = \{ a^i b^i \mid i \geq 1 \}$

The CNF of G is shown as follows :

$S \to AC \mid AB$

$C \to SB$

$A \to a$

$B \to b$

The CNF of G' is shown as follows :

$S \to CA \mid BA$

$C \to BS$

$A \to a$

$B \to b$

So $G' = \{ b^i a^i \mid i \geq 1 \}$

$\therefore H(x) = F(x^R)$ is context free ∎

Problem 1

⑤ $H(x) = \begin{cases} 1 & x = uv \text{ st. } F(u) = G(v) = 1 \\ 0 & \text{otherwise} \end{cases}$

Show or disprove that concatenation of 2 context free languages is also context free.

Proof : We want to show that concatenation of 2 context free language is also context free.
If F and G are context free, the concatenation of F and G are also context free.

① Let $\mathcal{L}_1 = \{ a^n b^n \mid n \geq 0 \}$

$\mathcal{L}_2 = \{ c^m d^m \mid m \geq 0 \}$

② The concatenation between $\mathcal{L}_1$ and $\mathcal{L}_2$ is
$\mathcal{L} = \{ a^n b^n c^m d^m \mid n, m \geq 0 \}$ because $\mathcal{L}_1$ says that the number of a's need to be equal to the number of b's and $\mathcal{L}_2$ says that the number of c's need to be equal to the number of d's. Therefore, the concatenation of $\mathcal{L}_1$ and $\mathcal{L}_2$ result as $\mathcal{L} = \{ a^n b^n c^m d^m \mid n, m \geq 0 \}$

③ $\mathcal{L} = \{ a^n b^n c^m d^m \mid n, m \geq 0 \}$ is also context free. as illustrated as the following example :

① $\mathcal{L}_1 = \{a^n b^n \mid n \geq 0\}$ is generated from this grammar

$S_1 \rightarrow a S_1 b$

$S_1 \rightarrow \varepsilon$

② $\mathcal{L}_2 = \{c^m d^m \mid m \geq 0\}$ is generated from this grammar

$S_2 \rightarrow c S_2 d$

$S_2 \rightarrow \varepsilon$

③ $\mathcal{L} = \{a^n b^n c^m d^m \mid n, m \geq 0\}$ is generated from this grammar

$S \rightarrow S_1 S_2$

$\therefore$ $\mathcal{L}$ is also context free ∎

Problem 1

⑥ $H(x) = \begin{cases} 1 & x = uu \text{ st. } F(u) = G(u) = 1 \\ 0 & \text{otherwise} \end{cases}$

Proof :

We want to show that the concatenation of 2 exactly the same context free language is not context free.

① Let $\mathcal{L}_1 = \{a^n b^n \mid n \geq 0\}$

$\mathcal{L}_2 = \{a^n b^n \mid n \geq 0\}$

② The concatenation between $\mathcal{L}_1$ and $\mathcal{L}_2$ is

$\mathcal{L} = \{a^n b^n a^n b^n \mid n \geq 0\}$ because $\mathcal{L}_1$ and $\mathcal{L}_2$ represents the same context free language. $\mathcal{L}_1$ says that the number of a's need to be equal to the number of b's and $\mathcal{L}_2$ says that the number of a's need to be equal to the number of b's. Therefore, the concatenation of $\mathcal{L}_1$ and $\mathcal{L}_2$ result as $\mathcal{L} = \{a^n b^n a^n b^n \mid n \geq 0\}$

③ $\mathcal{L} = \{a^n b^n a^n b^n \mid n \geq 0\}$ is not context free. according to Doubles is not a context free grammar which is proved in lecture as illustrated as follows.

So $H(x)$ is not context free ∎

Claim: Doubles is not context free grammar.

Proof: Suppose Doubles is context free.

$\exists P_0, P_1$ such that $s = 0^{P_0} 1^{P_0} ; 0^{P_0} 1^{P_0}$.

According to pumping lemma, for every grammar G,

$\exists P_0 \geq P_1$ such that $\forall s \in L_G$, $|s| \geq P_0$, $\exists s = axyzb$

ⓐ $ax^i y z^i b \in L_G \; \forall i \geq 0$

ⓑ $|xyz| \leq P_1$

ⓒ $|xz| \neq 0$

Case 1: x, z are on the same side of the separator ;

If ax^2yz^2b, then one side has changed, the other side hasn't changed.

$\Rightarrow ax^2yz^2b \notin \mathcal{L} \Rightarrow\Leftarrow$ Contradiction.

Case 2: x touches the left side while z touches the right side.

$|xyz| \leq P_1 \leq P_0 \Rightarrow x$ is only 1's

$\Rightarrow z$ is only 0's

$\Rightarrow ax^2yz^2b$ has more 1's to the left than right

$\Rightarrow ax^2yz^2b \notin \mathcal{L} \Rightarrow\Leftarrow$ Contradiction.

Case 3: x or z contains the separator ;

Pumping lemma would lead to multiple separators, but we are only allow to have one separator.

$\Rightarrow$ Doubles is not context free ∎

Problem 2 Exercises 10.2

Prove that the function $F: \{0,1\}^* \to \{0,1\}$ such that $F(x) = 1$ iff $|x|$ is a power of two is not context free.

Proof by contradiction:

Suppose G is context free, So by pumping lemma, $\exists P_0 \geq P_1$ st. $\forall s \in L_G$, $|s| \geq P_0$, $\exists s = axyzb$

(a) $a x^i y z^i b \in L_G$ $\forall i \geq 0$

(b) $|xyz| \leq P_1$

(c) $|xz| \neq 0$

Let consider $s = 0^{2^{P_0}} \in \mathcal{L}$

If $\mathcal{L}$ is context free then by applying pumping lemma to $0^{2^{P_0}}$ and $2^{P_0} \geq P_0$, then we have $0^{2^{P_0}} = axyzb$

By property (b), $x = 0^k$, $y = 0^j$ for some $k, j > 0$,

then $k + j = |xz| \leq |xyz| \leq P_0$

By property (c), $k + j = |xz| > 0$, this means

$0 < k + j \leq P_0$

By property (a), we try to pump x and z to be $a x^2 y z^2 b \in \mathcal{L}$, then we have

$$|a x^2 y z^2 b| = 2^{P_0} + \underbrace{(k+j)}_{\substack{\text{the pumped} \\ x \,\&\, z}} \leq 2^{P_0} + P_0 < 2^{P_0} + 2^{P_0} = 2^{P_0+1}$$

when $P_0 > 0$.

The pumped string $a x^2 y z^2 b$ has length strictly between 2^{P_0} and 2^{P_0+1}. There is no such string in $\mathcal{L}$, then our assumption about G is context free st. $a x^2 y z^2 b \in \mathcal{L}$ creates contradiction.

$\therefore$ $F(x) = 1$ iff $|x|$ is a power of 2 is not context free $\blacksquare$

Problem 3

a) Give a quantified integer statement to express the following: "The only solution to $x^a - y^b = 1$ is $x=3, a=2, y=2, b=3$"

$\Rightarrow \equiv \neg A \vee B$

Ans:

$$\exists x \exists a \exists y \exists b$$

$$((x^a - y^b = 1) \wedge \forall x_1 \forall a_1 \forall y_1 \forall b_1 \; ((\overbrace{(x_1^{a_1} - y_1^{b_1} = 1)}^{A} \Rightarrow$$

$$\underbrace{((x_1 = x) \wedge (a_1 = a) \wedge (y_1 = y) \wedge (b_1 = b))}_{B}) \wedge$$

$$(x_1 = 3) \wedge (a_1 = 2) \wedge (y_1 = 2) \wedge (b_1 = 3))$$

Define Macros:

$$x^a \equiv \forall k \in \mathbb{N}, (((k = a) \vee (\exists \ell \in \mathbb{N}, k = x\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$$

$$y^b \equiv \forall k \in \mathbb{N}, (((k = b) \vee (\exists \ell \in \mathbb{N}, k = y\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$$

$$x_1^{a_1} \equiv \forall k \in \mathbb{N}, (((k = a_1) \vee (\exists \ell \in \mathbb{N}, k = x_1\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$$

$$y_1^{b_1} \equiv \forall k \in \mathbb{N}, (((k = b_1) \vee (\exists \ell \in \mathbb{N}, k = y_1\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$$

Problem 3

b) 1729 is the smallest natural number that can be expressed as sum of 2 cubes in two different ways

Ans

$$\forall n \in \mathbb{N} \; ((\overbrace{\exists x \exists y \exists w \exists z (x^3 + y^3 = n \wedge w^3 + z^3 = n \wedge x \neq w \wedge x \neq z \wedge y \neq w \wedge y \neq z)}^{A})$$

$$\Rightarrow \underbrace{(n \geq 1729)}_{B})$$

$\Rightarrow \equiv \neg A \vee B$

Defined Macro

$x^3 \equiv \forall k \in \mathbb{N}, (((k = 3) \vee (\exists \ell \in \mathbb{N}, k = x\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$

$y^3 \equiv \forall k \in \mathbb{N}, (((k = 3) \vee (\exists \ell \in \mathbb{N}, k = y\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$

$w^3 \equiv \forall k \in \mathbb{N}, (((k = 3) \vee (\exists \ell \in \mathbb{N}, k = w\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$

$z^3 \equiv \forall k \in \mathbb{N}, (((k = 3) \vee (\exists \ell \in \mathbb{N}, k = z\ell)) \vee (\forall m \in \mathbb{N}, n \neq mk))$

Exercise 11.2

Part 1

Let FindProof : $\{0,1\}^* \to \{0,1\}$ be the following function. On input a Turing machine V (which we think of as the verifying algorithm for a proof system) and a string $x \in \{0,1\}^*$,

FindProof$(V, x) = 1$ iff $\exists\, w \in \{0,1\}^*$ st. $V(x, w) = 1$

Proof :

Let show a reduction from HaltOnZero to FindProof (the function that verifying algorithm for a proof system. First, let P be a TM that return a tuple of output. We can take a high level description of it. We now describe the reduction. The reduction R transform an input program M for HaltOnZero to the following program:

N(X) :

(a) Run M on 0 → If HaltOnZero = 1, then return (b)
If HaltOnZero = 0, won't go down to (b)

(b) Return P = (1, X) → input for FindProof

We set R(M) = N (ex. the program above). The reduction does not run M on 0 but just computes the description of the program N. Clearly, the program is computable.

Analysis: We need to argue that for any program,
$HaltOnZero(M) = FindProof(R(M)) = FindProof(N)$
There are 2 cases. If $HaltOnZero(M) = 1$, then N is
equivalent to $P = (1, X)$ so $FindProof(N) = FindProof(1, X)$
$= 1$

On the other hand, if $HaltOnZero(M) = 0$, then N would also not halt on any input, and in particular would not be equivalent to TM P so that $FindProof(N) = 0$

Exercise 11.2

Part 2 Prove that there exists a TM V st. V halts on aery input $\mathbf{x}$, V but the function $FindProof_V$ defined as $FindProof_V(x) = FindProof(V, x)$ is uncomputable

Proof :

Let show a reduction from HaltOnZero to $FindProof_V(x)$ which is finding a proof (the number of steps it takes) for a TM that halt on input u.

First, let P be a TM that output an always halting program.

We can take a high level description of it.

The reduction R transforms an input program M for HaltOnZero to the following program :

$N(x)$:

(a) Run M on 0

(b) Return $P(x)$

$P(x = z)$:
return z

$\}$ This is a program that halt on every input.

The function of P act as a decider which is a machine that always halts for every input, in our case, no matter what input, we will return the input as output.

We set $R(M) = N$. The reduction does not run M on 0.
but just computes the description of the program N.
Clearly, the transformation is computable.
Analysis: We need to argue that for any program M,
$\text{HaltOnZero}(M) = \text{FindProof}_V(R(M)) = \text{FindProof}_V(N)$
There are 2 cases. If $\text{HaltOnZero}(M) = 1$, then N is
equivalent to P which halt on every
input, so $\text{FindProof}_V(N) = 1 = \text{FindProof}_V(x)$
On the other hand, if $\text{HaltOnZero}(M) = 0$, then
N is not equivalent to P, then N would not halt on
any input, then $\text{FindProof}_V(N) = 0$

which means
loop infinitely
ex. while(true) { }