Problem 1

Theorem (A). Let $\mathcal{L}$ be a first-order language and let ϕ be an $\mathcal{L}$-formula such that the term t is substitutable for the variable x in ϕ. We have

$$T \vdash \phi \quad \Rightarrow \quad T \vdash \phi_t^x$$

for any $\mathcal{L}$-theory T.

Problem a

Prove Theorem **(A)** by constructing a derivation of ϕ_t^x from a derivation of ϕ. Name the (logical) axioms and the inference rules involved in the derivation.

——————————- Solution:

Assume $T \vdash \phi$, and let y be a variable not occurring in ϕ. We have the following T-derivation.

1.	ϕ	
2.	$y = y \to \phi$	1,PC
3.	$y = y \to \forall x\phi$	QR, no free x in $y = y$
4.	$y = y$	E1
5.	$\forall x\phi$	4,3,PC
6.	$\forall x\phi \to \phi_t^x$	Q1
7.	ϕ_t^x	5,6,PC

Thus, $T \vdash \phi_t^x$.

The derivation above makes it easy to see that the following theorem also holds.

Theorem (A0). Let $\mathcal{L}$ be a first-order language and let ϕ be an $\mathcal{L}$-formula such that the term t is substitutable for the variable x in ϕ. We have

$$T \vdash \phi \quad \Leftrightarrow \quad T \vdash \forall x\phi\ .$$

for any $\mathcal{L}$-theory T.

————————————-

Let $\circ$ be a binary function symbol, and let a, b and e be constant symbols. Let $\mathcal{L}_{BS}$ be the first-order language $\{a, b, e, \circ\}$, and let B be the $\mathcal{L}_{BS}$-theory consisting of the following non-logical axioms:

B1 $\forall x\,[\,x = e \circ x\,]$

B2 $\forall x\,[\,x = x \circ e\,]$

B3 $\forall xyz\,[\,x \circ (y \circ z) = (x \circ y) \circ z\,]$

B4 $\forall x\,[\,e \neq a \circ x \ \wedge \ e \neq b \circ x\,]$

B5 $\forall xy\,[\,x \neq y \ \rightarrow \ (a \circ x \neq a \circ y \wedge b \circ x \neq b \circ y)\,]$

Theorem (B). $B \vdash \forall x[e \circ x = x \circ e]$.

Problem b

Prove Theorem **(B)** by giving a B-derivation of $\forall x[e \circ x = x \circ e]$. Name the logical and the non-logical axioms involved in the derivation. You may refer to Theorem **(A)**. Hint: You will need the logical axiom

$$x_1 = y_1 \ \wedge \ x_2 = y_2 \ \rightarrow \ (x_1 = x_2 \ \rightarrow \ y_1 = y_2) \tag{E3}$$

——————————- Solution:

We have the following B-derivation.

1.	$x = e \circ x$	B1, A0
2.	$x = x \circ e$	B2, A0
3.	$x_1 = y_1 \ \wedge \ x_2 = y_2 \ \rightarrow \ (x_1 = x_2 \ \rightarrow \ y_1 = y_2)$	E3
4.	$x = e \circ x \ \wedge \ x = x \circ e \ \rightarrow \ (x = x \ \rightarrow \ e \circ x = x \circ e)$	3, A
5.	$x = x$	E1
6.	$e \circ x = x \circ e$	1,2,4,5,PC
7.	$\forall x[e \circ x = x \circ e]$	5,A0

——————————-

Problem c

Prove that

$$B \vdash \forall x y_1 \ldots y_n\,[\,(y_n \circ (y_{n-1} \circ \ldots (y_1 \circ e)\ldots)) \circ x \ = \ (y_n \circ (y_{n-1} \circ \ldots (y_1 \circ (x \circ e))\ldots))\,]$$

for any $n \geq 0$. Hints: Use induction on n. The case $n = 0$ follows from Theorem **(B)**.

——————————- Solution:

Assume by induction hypothesis that

$$B \vdash \forall x y_1 \ldots y_{n-1}\,[\,(y_{n-1} \circ \ldots (y_1 \circ e)\ldots) \circ x \ = \ (y_{n-1} \circ \ldots (y_1 \circ (x \circ e))\ldots)\,]$$

Let $\sigma \equiv (y_{n-1} \circ \ldots (y_1 \circ e)\ldots)$ and $\tau \equiv (y_{n-1} \circ \ldots (y_1 \circ (x \circ e))\ldots)$. Thus, we have

$$\forall x y_1 \ldots y_{n-1}\,[\,(y_{n-1} \circ \ldots (y_1 \circ e)\ldots) \circ x \ = \ (y_{n-1} \circ \ldots (y_1 \circ (x \circ e))\ldots)\,] \ \equiv \\ \forall x y_1 \ldots y_{n-1}\,[\,\sigma \circ x = \tau\,]\ .$$

In the next derivation, we will use

(*) For any B-terms s, t, u, we have $B \vdash s = t \rightarrow t = s$ and

$$B \vdash s = t \wedge t = u \rightarrow s = u\ .$$

It is not necessary to prove (*).

We have

1.	$\sigma \circ x = \tau$	ind.hyp, A0
2.	$x \circ (y \circ z) = (x \circ y) \circ z$	B3, A0
3.	$y_n \circ (y \circ z) = (y_n \circ y) \circ z$	2, A
4.	$y_n \circ (\sigma \circ z) = (y_n \circ \sigma) \circ z$	3, A
5.	$y_n \circ (\sigma \circ x) = (y_n \circ \sigma) \circ x$	4, A
6.	$y_n = y_n$	E0
7.	$y_n = y_n \ \wedge\ (\sigma \circ x) = \tau \quad \rightarrow \quad y_n \circ (\sigma \circ x) = (y_n \circ \tau)$	E2
8.	$y_n \circ (\sigma \circ x) = (y_n \circ \tau)$	1, 6, 7, PC
9.	$(y_n \circ \sigma) \circ x = (y_n \circ \tau)$	5, 8, (*), PC
10.	$\forall x y_1 \ldots y_{n-1} [\, (y_n \circ \sigma) \circ x = (y_n \circ \tau) \,]$	A0

Hence, the theorem holds as

$$\forall x y_1 \ldots y_{n-1} [\, (y_n \circ \sigma) \circ x = (y_n \circ \tau) \,] \quad \equiv$$
$$\forall x y_1 \ldots y_n [\, (y_n \circ (y_{n-1} \circ \ldots (y_1 \circ e) \ldots)) \circ x \ = \ (y_n \circ (y_{n-1} \circ \ldots (y_1 \circ (x \circ e)) \ldots)) \,]\ .$$

We define the *prime terms* of the language $\mathcal{L}_{BS}$ by

- e is a prime term
- $(a \circ t)$ is a prime term if t is a prime term
- $(b \circ t)$ is a prime term if t is a prime term.

Hence, e.g., $(a \circ (b \circ (b \circ e)))$ is a prime term whereas $((a \circ b) \circ (e \circ b))$ is not.

Theorem (C). For any variable-free $\mathcal{L}_{BS}$-term t there exists a prime term p such that $B \vdash t = p$.

Problem d

Prove Theorem **(C)**.

——————————- Solution:

We prove the theorem by induction over the structure of t. We have the following base cases

- $t \equiv e$
- $t \equiv a$
- $t \equiv b$

and the induction step $t \equiv t_1 \circ t_2$.

Case $t \equiv e$: We have $B \vdash e = e$ by (E1,A), and e is a prime term.

Case $t \equiv a$: We have $B \vdash a = (a \circ e)$ by (B2,A,A0), and $(a \circ e)$ is a prime term.

Case $t \equiv b$: We have $B \vdash b = (b \circ e)$ by (B2,A,A0), and $(b \circ e)$ is a prime term.

$t \equiv t_1 \circ t_2$: Strictly speaking we need the following slightly modified version of the statement in Problem **c**.

> **Lemma.**
>
> $$B \vdash \forall x y_1 \ldots y_n \, [\, (y_n \circ (y_{n-1} \circ \ldots (y_1 \circ e) \ldots)) \circ x \;=\; (y_n \circ (y_{n-1} \circ \ldots (y_1 \circ x) \ldots)) \,]$$

Assume by the induction hypothesis that we have prime terms p_1, p_2 such that $B \vdash t_1 = p_1$ and $B \vdash t_2 = p_2$. Then, p_1 is of the form $p_1 \equiv (c_n \circ \ldots (c_1 \circ e) \ldots)$ where $c_i \in \{a, b\}$ for $i = 1, \ldots n$. By Lemma, (A) and (A0), we have

$$B \vdash (c_n \circ (c_{n-1} \circ \ldots (c_1 \circ e) \ldots)) \circ p_2 \;=\; (c_n \circ (c_{n-1} \circ \ldots (c_1 \circ p_2) \ldots)) \, . \qquad (\dagger)$$

In the following derivation, we will also use

$$\text{For any } B\text{-terms } s, t, u, \text{ we have } B \vdash s = t \wedge t = u \rightarrow s = u \qquad (*)$$

Let $p \equiv (c_n \circ (c_{n-1} \circ \ldots (c_1 \circ p_2) \ldots))$. Then, p is a prime term, and we have

1.	$t_1 = p_1$	ind.hyp.
2.	$t_2 = p_2$	ind.hyp.
3.	$t_1 = p_1 \;\wedge\; t_2 = p_2 \;\;\rightarrow\;\; t_1 \circ t_2 = p_1 \circ p_2$	E2, A
4.	$t_1 \circ t_2 = p_1 \circ p_2$	1, 2, 3, PC
5.	$t_1 \circ t_2 = (c_n \circ (c_{n-1} \circ \ldots (c_1 \circ e) \ldots)) \circ p_2$	$p_1 \equiv \ldots$
6.	$(c_n \circ (c_{n-1} \circ \ldots (c_1 \circ e) \ldots)) \circ p_2 \;=\; p$	($\dagger$)
7.	$t_1 \circ t_2 = p$	5, 6, (*), PC

Thus, we have $B \vdash t = p$ for some prime term p.

Problem 2

We will use some notation from Levis & Papadimitriou's textbook: Σ^* denotes the set of all sequences over the alphabet Σ, and $|\alpha|$ denotes the length of the string α. We use ϵ to denote the empty string, and $\alpha \cdot \beta$ denotes the concatenation of the strings α and β. Occasionally, we will write $\alpha\beta$ in place of $\alpha \cdot \beta$. When convenient, we may also drop parenthesis and write e.g. $\alpha\beta\gamma$ in place of $(\alpha\beta)\gamma$.

We will now define the $\mathcal{L}_{BS}$-structure $\mathfrak{B}$. The universe of $\mathfrak{B}$ is the set $\{0,1\}^*$, that is, the set of all bit sequences: $\epsilon, 0, 1, 00, 01, 10, 11, 000, 001, \ldots$. Furthermore,

- $e^{\mathfrak{B}} = \epsilon$ (the empty string)
- $a^{\mathfrak{B}} = 0$ (the string where the one and only bit is 0)
- $b^{\mathfrak{B}} = 1$ (the string where the one and only bit is 1)

and $\circ^{\mathfrak{B}} = \cdot$ (the concatenation operator). Hence, we have e.g. that $\epsilon \circ^{\mathfrak{B}} 100 = 100 \circ^{\mathfrak{B}} \epsilon = 100$ and $101 \circ 1001 = 1011001$. It is obvious that $\mathfrak{B}$ is a model for the theory B, that is, $\mathfrak{B} \models B$.

Problem a

Is B a consistent theory? Give a short answer, and justify the answer by referring to a theorem in Leary's textbook.

——————————- Solution:

We have $\mathfrak{B} \models B$. Thus, B has a model, and then by the Soundness Theorem for first-order logic, B is consistent.

——————————

Problem b

Do we have $B \vdash a \neq b$? Do we have $B \vdash a = b$? Justify your answers.

——————————- Solution:

We have $\mathfrak{B} \models B$ and $\mathfrak{B} \not\models a = b$, and thus, $B \nvdash a = b$ by the Soundness Theorem for first-order logic.

We will now define the $\mathcal{L}_{BS}$-structure $\mathfrak{A}$. The universe of $\mathfrak{A}$ is the set $\{0\}^*$, that is, the set containing the following sequences: $\epsilon, 0, 00, 000, 0000, \ldots$. Furthermore,

- $e^{\mathfrak{A}} = \epsilon$
- $a^{\mathfrak{A}} = 0$
- $b^{\mathfrak{A}} = 0$

and $\circ^{\mathfrak{A}} = \cdot$ (the concatenation operator).

Now, we have $\mathfrak{A} \models B$ and $\mathfrak{A} \not\models a \neq b$, and thus, $B \nvdash a \neq b$ by the Soundness Theorem for first-order logic.

——————————-

We say that α is a sub string of β iff there exists γ_1 and γ_2 such that $\gamma_1\alpha\gamma_2 = \beta$.

Problem c

Give an $\mathcal{L}_{BS}$-formula θ such that

$$\mathfrak{B} \models \theta[s[y|\alpha][x|\beta]] \Leftrightarrow \alpha \text{ is a sub string of } \beta\ .$$

Give an $\mathcal{L}_{BS}$-formula η such that

$$\mathfrak{B} \models \eta[s[x|\alpha]] \quad \Leftrightarrow \quad \alpha \in \{0\}^* \ .$$

———————————- Solution:

$$\theta(y, x) \ \equiv \ \exists uv[(u \circ y) \circ v = x] \quad \text{and} \quad \eta(x) \ \equiv \ \neg\theta(b, x) \ .$$

———————————-

Problem d

Give an $\mathcal{L}_{BS}$-formula Add such that $\mathfrak{B} \models \mathit{Add}[s[x|\alpha]]$ holds if and only if

- α is of the form $\alpha \equiv 1\gamma_1 1\gamma_2 1\gamma_3 1$ where $\gamma_1, \gamma_2, \gamma_3 \in \{0\}^*$, and
- $|\gamma_1| + |\gamma_2| = |\gamma_3|$.

(Hint: Use the formulas from Problem **b**.)

———————————- Solution:

$$\mathit{Add}(x) \ \equiv \ \exists uvw \, [\, \eta(u) \ \wedge \ \eta(v) \ \wedge \ \eta(w) \ \wedge \ x = b \circ u \circ b \circ v \circ b \circ w \circ b \ \wedge \ u \circ v = w \,]$$

———————————

Theorem (D). There exist $\mathcal{L}_{BS}$-formulas Mul and Exp such that

- $\mathfrak{B} \models \mathit{Mul}[s[x|\alpha]]$ if and only if
  - α is of the form $\alpha \equiv 1\gamma_1 1\gamma_2 1\gamma_3 1$ where $\gamma_1, \gamma_2, \gamma_3 \in \{0\}^*$, and
  - $|\gamma_1| \times |\gamma_2| = |\gamma_3|$
- $\mathfrak{B} \models \mathit{Exp}[s[x|\alpha]]$ if and only if
  - α is of the form $\alpha \equiv 1\gamma_1 1\gamma_2 1\gamma_3 1$ where $\gamma_1, \gamma_2, \gamma_3 \in \{0\}^*$, and
  - $|\gamma_1|^{|\gamma_2|} = |\gamma_3|$.

The proof of Theorem **(D)** is involved, and you are *not* asked to prove this theorem.

Let $\mathcal{L}_{NT}$ be the first-order language of number theory, that is, $\mathcal{L}_{NT} = \{0, S, +, \times, E, <\}$, and let $\mathfrak{N}$ be the standard $\mathcal{L}_{NT}$-structure. Both $\mathcal{L}_{NT}$ and $\mathfrak{N}$ are known from Leary's textbook. Furthermore, let Σ be an alphabet containing all the symbols of the first-order languages $\mathcal{L}_{NT}$ and $\mathcal{L}_{BS}$.

Problem e

Argue that there exists a recursive (Turing computable) function $f : \Sigma^* \to \Sigma^*$ such that for any $\mathcal{L}_{NT}$-sentence ϕ

$$\mathfrak{N} \models \phi \quad \Leftrightarrow \quad \mathfrak{B} \models f(\phi) \ .$$

You may refer to Theorem **(D)**.

———————————- Solution:

We say that a $\mathcal{L}_{NT}$-formula ϕ is *pure* if any atomic sub formula of ϕ is in one of the forms

- $0 = x$
- $x = y$
- $x < y$
- $S(x) = y$
- $x + y = z$
- $x \times y = z$
- $x \, E \, y = z$.

For any $\mathcal{L}_{NT}$-formula ϕ, there exists a pure formula ϕ_0 such that $\mathfrak{N} \models \phi \leftrightarrow \phi_0$ (we even have $\models \phi \leftrightarrow \phi_0$). Moreover, such a pure formula ϕ_0 can be constructed from ϕ by an algorithm, and thus, we have a recursive function *pure* such that $pure(\phi) = \phi_0$.

[*Example:* Let $\phi \equiv s(s(0)) < x + y$, and let

$$\phi \equiv \exists z_0, z_1, z_2, z_3 \, [\, 0 = z_0 \;\wedge\; S(z_0) = z_1 \;\wedge\; S(z_1) = z_2 \;\wedge\; x + y = z_3 \;\wedge\; z_2 < z_3 \,] \; .$$

Then, ϕ_0 is a pure formula such that $\mathfrak{N} \models \phi \leftrightarrow \phi_0$. *End of example.*]

Next, we define a function f' recursively over the structure of a pure formula:

- $f'((\alpha \vee \beta)) = (f'(\alpha) \vee f'(\beta))$
- $f'((\neg \alpha)) = (\neg f'(\alpha))$
- $f'((\forall x)(\alpha)) = (\forall x)((\eta(x) \rightarrow f'(\alpha)) \wedge (\neg \eta(x) \rightarrow f'(\alpha^x_e)))$ where η is the formula from Problem **c**, and α^x_e is α where all free occurrences of x is replaced by the term e. (This definition reflects that the sequence 0^n represents the natural number n, and any sequence containing the bit 1 represents the natural number 0.)

For the atoms we define f' as follows:

- $f'(0 = x) = e = x$
- $f'(x = y) = x = y$
- $f'(x < y) = \exists z[z \neq e \wedge x \circ z = y]$
- $f'(S(x) = y) = x \circ a = y$
- $f'(x + y = z) = Add(b \circ x \circ b \circ y \circ b \circ z \circ b)$ where *Add* is the formula from Problem **d**
- $f'(x \times y = z) = Mul(b \circ x \circ b \circ y \circ b \circ z \circ b)$ where *Mul* is the formula in Theorem (D)
- $f'(x \, E \, y = z) = Exp(b \circ x \circ b \circ y \circ b \circ z \circ b)$ where *Exp* is the formula in Theorem (D).

The function f' is computable, and for any pure formula ϕ, we have

$$\mathfrak{N} \models \phi \quad \Leftrightarrow \quad \mathfrak{B} \models f'(\phi) \; .$$

Finally, let $f(\phi) = f'(pure(\phi))$. Then, f is a recursive function such that

$$\mathfrak{N} \models \phi \quad \Leftrightarrow \quad \mathfrak{B} \models f(\phi) \; .$$

——————————-

Let

$$Th(\mathfrak{B}) \;=\; \{\, \phi \mid \phi \text{ is an } \mathcal{L}_{BS}\text{-sentence and } \mathfrak{B} \models \phi \,\}$$

and

$$Pr(B) \;=\; \{\, \phi \mid \phi \text{ is an } \mathcal{L}_{BS}\text{-sentence and } B \vdash \phi \,\} \; .$$

Problem f

Is the set $Th(\mathfrak{B})$ recursive (decidable)? Is the set $Th(\mathfrak{B})$ recursively enumerable (semi-decidable)? Is the set $Pr(B)$ recursively enumerable (semi-decidable)? Justify your answers.

——————————- Solution:

The set $Th(\mathfrak{B})$ recursive is not recursive. *Justification:* Assume $Th(\mathfrak{B})$ is recursive. Let f be the computable function from Problem **e**. We can decide if $\mathfrak{N} \models \phi$ for an arbitrary sentence ϕ by the following algorithm: Compute $f(\phi)$ and check if $f(\phi) \in Th(\mathfrak{B})$. If $f(\phi) \in Th(\mathfrak{B})$, we have $\mathfrak{N} \models \phi$; and if $f(\phi) \notin Th(\mathfrak{B})$, we have $\mathfrak{N} \not\models \phi$. Thus, we have a recursive set of axioms T such that $T \vdash \phi$ iff $\mathfrak{N} \models \phi$. (Simply let T be set of all sentences true in $\mathfrak{N}$.) This contradicts Gödel's (first) Incompleteness Theorem.

The set $Th(\mathfrak{B})$ is not recursively enumerable. *Justification:* Assume $Th(\mathfrak{B})$ is recursively enumerable. Then we can decide if $\psi \in Th(\mathfrak{B})$ (for an arbitrary sentence ψ) by enumerating all the the sentences $Th(\mathfrak{B})$. Either ψ or $\neg\psi$ will show in the enumeration. If ψ shows up, then $\psi \in Th(\mathfrak{B})$; and if $\neg\psi$ shows up, then $\psi \notin Th(\mathfrak{B})$. This contradicts that $Th(\mathfrak{B})$ is not recursive. (Indeed, for any language $\mathcal{L}$ and any $\mathcal{L}$-structure $\mathfrak{A}$, the set $Th(\mathfrak{A})$ is recursively enumerable if and only if it is recursive.)

The set $Pr(B)$ is recursively enumerable since the set B (the axioms) is recursively enumerable.

——————————-

Problem g

Is the theory $B \cup \{a \neq b\}$ complete or incomplete? Justify your answer.

——————————- Solution:

The theory is incomplete. *Justification:* We have $\mathfrak{B} \models B \cup \{a \neq b\}$. If the theory were complete, the set $Th(\mathfrak{B})$ would be recursive.

——————————-