

Ethica

Ordine Geometrico Demonstrata

by
Benedict de Spinoza

AS EXECUTABLE METAPHYSICS

A Functional Translation into Python Logic

by Erez Ashkenazi

Includes Definitions, Axioms, Propositions, and Logical Modeling
from Books I–V

Notes on Influence:

Benedict de Spinoza (1677) – Original Latin

R.H.M. Elwes (1883) – main structure and vocabulary base (public domain)

Edwin Curley (1985) – used for cross-checking meaning and nuance

Jonathan Bennett (2001) – used for simplified paraphrase comparison

Spinoza's Ethics as Executable Metaphysics

A Functional Translation of the Five Books into Python

This project is a systematic, proposition-by-proposition encoding of *Baruch Spinoza's Ethics* into a functional and symbolic Python model. Rather than merely paraphrasing Spinoza's Latin logic, this translation seeks to **formalize his metaphysical system** — as a **computable ontology of substance, mode, cause, and affect**.

Each Book is mapped to:

- **Definitions** → Python classes (e.g., Substance, Mode, Idea, Affect)
- **Axioms** → Logical functions (e.g., Cause, Exists, GoodIsWhatIncreasesPower)
- **Propositions** → Human-readable logic and lambda-style pseudocode

This approach enables:

- **Executable metaphysics** — test, extend, and simulate Spinoza's ideas as living code
- **Causal recursion modeling** — build ontologies where ideas reflect ideas
- **AI and cognitive architecture design** — grounded in a rigorous metaphysical foundation
- **Ethical computation** — define virtue, freedom, and joy in formal terms

"The more each thing strives to preserve itself, the more it is endowed with virtue or power." — Ethics III.20

By translating *Ethics* into a structured, modular, and programmable system, this project fulfills Spinoza's deepest aim: not to preach, but to **demonstrate reality through reason** — and now, through code.

Background & Motivation

Ethics, Spinoza's magnum opus, is one of the most ambitious attempts in Western philosophy to explain the nature of reality, the mind, emotions, and human freedom through a strictly deductive, geometric method in an absolute immanent system. Structured after Euclid's *Elements*, Spinoza builds a rational system from definitions, axioms, and propositions — aiming to deduce not only metaphysical truths but the causes of human suffering, the nature of joy, and the path to liberation.

Despite its logical rigor, *Ethics* has long remained inaccessible to systems modeling, computational analysis, or practical application due to its abstract style and symbolic density. At the same time, modern developments in artificial intelligence, formal epistemology, and computational ontology demand frameworks capable of modeling reason, emotion, and ethical action at a structural level.

This project answers that need by **translating Spinoza's entire system into functional code** — making his logic executable, his metaphysics inspectable, and his ethical insights testable.

The motivation is twofold:

1. **Philosophical:** To honor and operationalize Spinoza's vision of reality as a necessary, intelligible, and ethically structured whole — one that can now serve as a foundation for epistemic clarity, cognitive architectures, and rational autonomy in machines and humans alike.
2. **Computational:** To prototype a formal system where **ideas, affects, and adequacy** can be represented, reasoned with, and composed into recursive, causal chains — making it possible to simulate the dynamics of freedom, bondage, and joy as processes of logical transformation.

In short, the goal is to bring *Ethics* from parchment to processor — from timeless rational system to timeless executable logic.

Abstract

This work presents a full-system formal translation of *Baruch Spinoza's Ethics* into a structured and functional Python ontology. Each of the five books — from *De Deo* (On God) to *De Libertate Humana* (On Human Liberty) — is encoded using object-oriented classes, axiomatic functions, and executable propositions. By rendering Spinoza's metaphysics as a programmable structure, the system models substance, attribute, mode, idea, and affect in a logically traceable, testable form. This translation enables both philosophical analysis and computational experimentation: ideas can now be reasoned with, simulated, and composed algorithmically. The result is not merely a digitization, but an executable metaphysics — a working foundation for formal epistemology, ethical cognition, and ontological engineering.

Code Legend (Inserted for Clarity)

\Leftrightarrow	means 'if and only if'	\forall	means 'for all'	\in	means 'is element of'
\propto	means 'proportional to'	Δ	means 'change in'	\subseteq	means 'is subset of'

Glossary of Core Logical Functions and Symbols

- **conatus(x)**: Striving of entity x to persist (core of essence)
 - **delta_conatus(x)**: Increase or decrease in power of action
 - **is_adequate(idea)**: Idea contains its own cause and agrees with its object
 - **cause(effect)**: What necessarily produces an effect
 - **eternal_part(mind)**: The adequate ideas that persist beyond duration
 - **joy(x)**: Transition to greater perfection
 - **sadness(x)**: Transition to lesser perfection
 - **intellectual_love_of_God(mind)**: Eternal joy from understanding God's nature
-

* *This document does not offer a literal programming environment, but a functional and formal translation of Spinoza's Ethics into symbolic logic expressed in Python-like pseudocode. Each function expresses a necessary relational structure between ideas, causes, affects, and adequacy. This translation renders the Ethics into a causal-executable system, suitable for computational philosophy, epistemic modeling, and AI ethics research.*

The Ethics

Ethica Ordine Geometrico Demonstrata

(Translated to Python)

BOOK I: Of God

BOOK II: Of the Nature and Origin of the Mind

BOOK III: On the Origin and Nature of the Emotions

BOOK IV: Of Human Bondage, or the Strength of the Emotions

BOOK V: Of the Power of the Intellect, or of Human Liberty

BOOK I

OF GOD

DEFINITIONS

These 8 definitions are foundational — they define the **metaphysical vocabulary** of the entire system.

Definition I: Cause of itself (Causa sui)

That whose essence **involves existence**, or whose nature **cannot be conceived except as existing**.

```
def is_causa_sui(x):  
    return x.essence.includes("existence")
```

Definition II: Finite in its kind

A thing is finite if another thing of the same nature can **limit it**.

```
def is_finite(x):  
    return exists(y) such_that (same_kind(x, y) and x.limited_by(y))
```

Definition III: Substance

That which is **in itself** and is **conceived through itself** — its conception does not require the idea of another thing.

```
def is_substance(x):  
    return in_itself(x) and self_conceived(x)
```

where:

```
# in_itself(x): True if x is ontologically independent (not a mode)
# self_conceived(x): True if x can be fully understood without reference to another
    return in_itself(x) and self_conceived(x)
```

Definition IV: Attribute

That which the intellect perceives as **constituting the essence of substance**.

```
def is_attribute(x):
    return x in essence(substance)
```

Definition V: Mode

That which exists **in something else** and is conceived **through that other thing**.

```
def is_mode(x):
    return in(something_else, x)
```

Definition VI: God

A being absolutely infinite — a substance consisting of **infinite attributes**, each expressing eternal and infinite essence.

God = Substance(attributes=∞)

Definition VII: Free

That which exists **solely by the necessity of its nature**, and is determined to act **by itself alone**.

```
def is_free(x):
    return caused_only_by_self(x)
```

Definition VIII: Eternity

Existence itself, conceived as **following necessarily from the definition** of an eternal thing.

```
def is_eternal(x):  
    return existence(x) follows_from definition(x)
```

AXIOMS

7 **self-evident truths** that ground the geometry of the system:

Axiom I

Everything which exists, exists either **in itself or in something else**.

```
forall x: exists(x) ⇒ (in_itself(x) or in_another(x))
```

Axiom II

That which cannot be **conceived through another**, must be **conceived through itself**.

Axiom III

From a given **cause**, an **effect** necessarily follows; and if there is no cause, no effect can follow.

Axiom IV

The knowledge of an effect **depends on and involves** the knowledge of the cause.

Axiom V

Things which have nothing in common **cannot be understood through one another.**

Axiom VI

A true idea must **agree with its object.**

Axiom VII

If a thing can be **conceived as not existing**, its essence **does not involve existence.**

PROPOSITIONS

PROPOSITION I

"Substance is by nature prior to its modifications."

Translation:

A substance exists before (and independently of) any changes or attributes it might have.

Code:

```
def is_prior(substance, mode):  
    return True # substance > mode
```

PROPOSITION II

"Two substances having different attributes have nothing in common."

Translation:

If two substances have different essential properties, they cannot overlap.

Code:

```
def have_commonality(sub1, sub2):  
    return False if sub1.attributes != sub2.attributes else True
```

PROPOSITION III

"Things which have nothing in common cannot be the cause of one another."

Translation:

If two things share no qualities, one cannot produce or cause the other.

Code:

```
def can_cause(x, y):  
    return x.attributes == y.attributes
```

PROPOSITION IV

"Two or more distinct things are distinguished either by the difference of their attributes or by the difference of their modifications."

Translation:

Things are different if their essence (attributes) or form (modes) differ.

Code:

```
def are_distinct(x, y):  
    return (x.attributes != y.attributes) or (x.modes != y.modes)
```

PROPOSITION V

"In nature there cannot be two or more substances of the same nature or attribute."

Translation:

There is only one substance per attribute — like only one substance of extension, one of thought.

Code:

```
substances = []
```

```
def add_substance(new):
    for s in substances:
        if s.attributes == new.attributes:
            raise Exception("Duplicate attribute")
    substances.append(new)
```

PROPOSITION VI

"One substance cannot be produced by another substance."

Translation:

No substance creates another — each is self-caused (if at all).

Code:

```
def is_cause_of(x, y):
    if is_substance(x) and is_substance(y):
        return False
```

PROPOSITION VII

"It pertains to the nature of a substance to exist."

Translation:

A true substance must include existence in its essence.

Code:

```
def essence_implies_existence(x):  
    return is_substance(x)
```

PROPOSITION VIII

"Every substance is necessarily infinite."

Translation:

No substance can be finite — it has no external limits.

Code:

```
def is_finite(x):  
    return not is_substance(x)
```

PROPOSITION IX

"The more reality or being a thing has, the more attributes it has."

Translation:

A thing is more “real” the more ways it expresses existence.

Code:

```
def degree_of_reality(x):  
    return len(x.attributes)
```

PROPOSITION X

"Each attribute of a substance must be conceived through itself."

Translation:

Attributes don’t depend on each other for understanding — each is self-explanatory.

Code:

```
def is_self_conceived(attr):  
    return True
```

PROPOSITION XI

"God, or substance, consisting of infinite attributes, each of which expresses eternal and infinite essence, necessarily exists."

Translation:

God necessarily exists and is infinite in every possible attribute.

Code:

```
God = Substance(attributes="infinite")  
assert God.exists == True
```

PROPOSITION XII

"No attribute of a substance can be truly conceived from which it would follow that the substance can be divided."

Translation:

A true substance cannot be split or broken down.

Code:

```
def can_be_divided(substance):  
    return False
```

PROPOSITION XIII

"A substance which is absolutely infinite is indivisible."

Translation:

If a thing is truly infinite, it cannot be separated into parts.

Code:

```
if is_absolutely_infinite(substance):  
    assert is_indivisible(substance)
```

PROPOSITION XIV

"Besides God, no substance can be or be conceived."

Translation:

Only one substance exists: God.

Code:

```
def is_substance(x):  
    return in_itself(x) and self_conceived(x)  
  
# where:  
# in_itself(x): True if x is ontologically independent (not a mode)  
# self_conceived(x): True if x can be fully understood without reference to another  
return x == "God"
```

PROPOSITION XV

"Whatever is, is in God, and nothing can be or be conceived without God."

(Same as Proposition 22 in wording — repeated principle.)

Code:

```
def exists_in(x):  
    return "God"
```

PROPOSITION XVI

"From the necessity of the divine nature must follow infinite things in infinite ways."

Translation:

God's essence necessarily produces the entire universe and all modes.

Code:

```
def divine_expression(God):  
    return generate_modes(infinite=True)
```

PROPOSITION XVII

"God acts from the laws of His nature alone and is compelled by no one."

Translation:

God's actions are entirely self-determined.

Code:

```
def is_compelled(God):  
    return False
```

PROPOSITION XVIII

"God is the indwelling cause of all things, not a transient one."

(Same as Proposition 23 — repeated insight)

Code:

```
cause_type = "immanent"
```

PROPOSITION XIX

"God is eternal, or all the attributes of God are eternal."

Translation:

All of God's attributes are without beginning or end.

Code:

```
for attr in God.attributes:  
    assert is_eternal(attr)
```

PROPOSITION XX

"The existence of God and His essence are one and the same."

Translation:

To exist *is* part of what it means to be God — they are inseparable.

Code:

```
assert God.essence == God.existence
```

PROPOSITION XXI

"All things which follow from the absolute nature of any attribute of God must always exist and be infinite."

Translation:

Whatever necessarily follows from the very nature of any divine attribute must exist eternally and without limitation.

Code Logic:

```
def follows_from(attribute):  
    return is_infinite(attribute) and is_eternal(attribute)
```

```
# Example  
if follows_from("thought"):
```

```
print("The modes of thought are eternal and infinite")
```

PROPOSITION XXII

"Whatever is, is in God, and nothing can be nor be conceived without God."

Translation:

Everything that exists is within God, and nothing can be or even be conceived without being part of God's nature.

Code Logic:

```
def is_in_God(x):  
    return True # All things are modes of God  
  
def can_exist_without_God(x):  
    return False # Nothing exists or is conceived outside God
```

PROPOSITION XXIII

"God is the immanent, not the transitive, cause of all things."

Translation:

God does not act upon the world from outside but is the internal, ever-present cause of everything.

Code Logic:

```
def cause_of_all(x):  
    return "God" # Immanent cause, not external or mechanical  
  
cause_type = "immanent" # not transitive
```

PROPOSITION XXIV

"The essence of things produced by God does not involve existence."

Translation:

Although all things proceed from God, their essence alone does not imply they must exist — only God's essence includes necessary existence.

Code Logic:

```
def essence_implies_existence(x):  
    return x == "God"  
  
# Anything other than God has contingent existence
```

PROPOSITION XXV

"God is the efficient cause not only of the existence of things but also of their essence."

Translation:

God does not merely cause things to exist — even their very nature or definition is determined by God.

Code Logic:

```
def efficient_cause(x):  
    return "God" # for both existence and essence
```

PROPOSITION XXVI

"A thing which is conditioned to act in a particular way has been determined by another thing."

Translation:

If something acts in a certain way, it must have been determined by something else to do so.

Code Logic:

```
def is_determined(x):  
    return exists(cause_of(x))  
  
# Nothing acts freely except God
```

PROPOSITION XXVII

"A thing which has been determined by God to act in a particular way cannot render itself undetermined."

Translation:

Nothing can undo God's determination — if God caused something to be or act, it cannot free itself from that determination.

Code Logic:

```
def can_be_undetermined(x):  
    return False if cause_of(x) == "God" else True
```

PROPOSITION XXVIII

"Every individual thing, or mode, that exists, necessarily follows from the nature of God."

Translation:

All finite things necessarily emerge from God's infinite attributes — nothing is arbitrary.

Code Logic:

```
def mode_exists(x):  
    return follows_from("God")  
  
# All modes are necessary outcomes
```

PROPOSITION XXIX

"Nothing in the universe is contingent, but all things are conditioned to exist and act in a particular way by the necessity of the divine nature."

Translation:

There is no true randomness in the universe — everything is determined by God's nature.

Code Logic:

```
def is_contingent(x):  
    return False # All is necessary by God's nature
```

PROPOSITION XXX

"Intellect, and will, which are supposed to constitute God's essence, differ in no respect from His nature."

Translation:

God's intellect and will are not separate faculties but identical with His essence — not like human faculties.

Code Logic:

```
God_intellect = God_essence  
God_will = God_essence
```

PROPOSITION XXXI

"Intellect, in function and in act, is God's attribute."

Translation:

Thinking and understanding are not external additions to God — they are essential attributes of divine nature.

Code:

```
God.attributes.append("intellect")
```

```
assert "intellect" in God.attributes
```

PROPOSITION XXXII

"The will cannot be called a free cause, but only a necessary cause."

Translation:

There is no such thing as absolute free will — even will acts by necessity from its cause.

Code:

```
def is_free_cause(will):  
    return False # Will is determined by prior causes
```

PROPOSITION XXXIII

"Things could have been produced by God in no other way, and in no other order."

Translation:

The universe is exactly as it must be — nothing could be different, even in arrangement.

Code:

```
def alternative_world_possible():  
    return False # This is the only logically necessary reality
```

PROPOSITION XXXIV

"God's power is identical with His essence."

Translation:

God doesn't *have* power — God *is* power. His nature and capability are one.

Code:

```
God.power = God.essence
```

PROPOSITION XXXV

"Whatever we conceive to be in God's power, is necessarily so."

Translation:

If you can truly understand something as part of God's nature, it *must* be real.

Code:

```
def is_possible_by_God(x):  
    return necessarily_real(x)
```

PROPOSITION XXXVI

"Nothing exists from whose nature some effect does not follow."

Translation:

Everything has some causal power. If a thing exists, it necessarily produces effects.

Code:

```
def has_effect(x):  
    return x.exists  
  
# Apply to all modes  
for mode in all_things:  
    assert has_effect(mode)
```

BOOK II

OF THE NATURE AND ORIGIN OF THE MIND

DEFINITIONS

Definition I: Body

A mode which expresses in a certain and determinate manner the essence of God in so far as He is considered as an extended thing.

Body = Mode(attribute="extension", determined=True)

Definition II: Idea

A concept of the mind formed by the mind as a thinking thing.

```
def is_idea(x):  
    return x ∈ Mind.Thought
```

Definition III: Adequate Idea

An idea which, in so far as it is considered in itself, without relation to its object, has all the properties of a true idea.

```
def is_adequate(idea):  
    return idea.contains(its_cause) and matches(reality)
```

Definition IV: False Idea

An idea which is not adequate.

```
def is_false(idea):  
    return not is_adequate(idea)
```

AXIOMS

Axiom I

Man thinks.

Human ∈ ThinkingBeings

Axiom II

A mode of thinking can be caused only by another mode of thinking.

Axiom III

The body cannot determine the mind to think, nor the mind determine the body to move — except through their unity.

Mind ↔ Body only by parallelism

Axiom IV

The knowledge of an effect depends on knowledge of its cause.

Axiom V

Things that have nothing in common cannot be understood through one another.

Axiom VI

A true idea must agree with its object.

Axiom VII

If a thing can be conceived as not existing, its essence does not involve existence.

PROPOSITIONS

PROPOSITION I

"The human mind is part of the infinite intellect of God."

Translation:

The mind is a mode of the divine attribute of thought.

Code:

HumanMind ⊂ God.Thought

PROPOSITION II

"Man thinks."

Translation:

Self-evident: humans are thinking beings.

Code:

assert "thought" in Human.attributes

PROPOSITION III

"In God there is the idea of everything that is caused."

Translation:

Everything that exists is also conceived — God has an idea of everything.

Code:

for thing in Reality:

```
assert idea_of(thing) in God.Thought
```

PROPOSITION IV

"The idea of anything caused depends on the knowledge of the cause."

Translation:

You can't truly understand something without understanding what caused it.

Code:

```
def true_idea(x):
```

```
    return know(cause_of(x))
```

PROPOSITION V

"The essence of man consists in certain modes of God's attributes."

Translation:

What makes a human is a particular pattern (mode) of divine attributes — especially thought and extension.

Code:

```
Human = Mode(God.Thought, God.Extension)
```

PROPOSITION VI

"The idea of each mode God creates exists in God."

Translation:

For every thing, its idea exists within God.

Code:

for mode in Modes:

```
assert idea_of(mode) in God.Thought
```

PROPOSITION VII

"The order and connection of ideas is the same as the order and connection of things."

This is the foundational principle of parallelism.

Translation:

Ideas are structured just like physical things — there is a one-to-one match between mental and physical order.

Code:

```
∀x: order(ideas[x]) == order(things[x])
```

PROPOSITION VIII

"The idea of a singular thing caused by another singular thing includes the idea of that cause."

Translation:

If you understand a thing, the cause is part of that understanding.

Code:

```
def idea(x):  
    return [x] + idea(cause_of(x))
```

PROPOSITION IX

"The idea of a singular thing which is produced under a modification of God's attribute is also necessarily caused by God."

Translation:

If something is a mode of God, its idea is also from God — not external.

Code:

```
def idea_of(x):  
    if is_mode_of_God(x):  
        return cause(idea_of(x)) == God
```

PROPOSITION X

"The being of substance does not pertain to the essence of man."

Translation:

Humans are not substances — we're modes, not self-caused beings.

Code:

```
def is_substance(x):  
    return in_itself(x) and self_conceived(x)
```

where:

```
# in_itself(x): True if x is ontologically independent (not a mode)  
# self_conceived(x): True if x can be fully understood without reference to another  
    return x != Human # Human = mode, not substance
```

PROPOSITION XI

"The first thing which constitutes the actual being of a human mind is nothing but the idea of a singular thing which is actually existent."

Translation:

The human mind is the idea of the human body — a mental representation of a specific physical thing.

Code:

```
HumanMind = idea_of(HumanBody)
```

PROPOSITION XII

"Whatever happens in the object of the idea constituting the human mind must be perceived by the human mind."

Translation:

Anything that happens in the body is reflected in the mind — the mind perceives all changes in its object (the body).

Code:

```
def body_event(event):  
    HumanMind.perceives(event)
```

PROPOSITION XIII

"The object of the idea constituting the human mind is the body, or a certain mode of extension."

Translation:

The mind doesn't represent anything abstract — it is specifically the idea of the body.

Code:

```
HumanMind = idea_of(HumanBody) # Reaffirming Proposition XI
```

PROPOSITION XIV

"The human mind is capable of perceiving many things and is so whether the body is affected by them or not."

Translation:

The mind isn't limited to what happens in the body — it can form ideas independently (e.g. through reason or imagination).

Code:

```
def can_perceive(mind, x):  
    return True # Not limited to bodily affections
```

PROPOSITION XV

"The idea that constitutes the formal being of the human mind is not simple, but composed of many ideas."

Translation:

The mind isn't one single idea — it's a structured system of ideas about many aspects of the body.

Code:

```
HumanMind = CompositeIdea([  
    idea_of(body_part) for body_part in HumanBody.parts  
)
```

PROPOSITION XXVI

"The human mind does not perceive any external body as actually existing, except through the ideas of the modifications of its own body."

Translation:

We only experience the outside world as it modifies our own body — our mind represents those effects, not the external things directly.

Code:

```
def perceive_external(x):  
    return idea_of(affection_on(HumanBody, x))
```

PROPOSITION XXVII

"The idea of each affection of the human body does not involve an adequate knowledge of the external body."

Translation:

These ideas (sensory experiences) don't give us complete understanding of their causes — they are partial, passive, and confused.

Code:

```
def is_inadequate(idea):  
    return False if idea.based_on == "affection_only" else True
```

PROPOSITION XXVIII

"The idea of the affections of the human body, as involving both the nature of the human body and of external bodies, must be confused."

Translation:

Since each idea involves both our body and something external — but without clear causal understanding — they are confused.

Code:

```
def clarity_of_idea(idea):  
    return "confused" if involves_external(idea) else "clear"
```

PROPOSITION XXIX

"The idea of an idea of an affection of the human body does not involve adequate knowledge of the human mind."

Translation:

Even when we reflect on our thoughts or feelings (second-order ideas), if they arise from sensory affections, they remain inadequate.

Code:

```
def second_order_idea(idea):  
    return idea_of(idea)  
  
assert not is_inadequate(second_order_idea(affection_idea))
```

PROPOSITION XXX

"We can have only a very inadequate knowledge of the duration of our body."

Translation:

Because our perception is based on passing affections, we have no full grasp of how our body persists through time.

Code:

```
def knowledge_of_duration(body):  
    return "very_inadequate"
```

PROPOSITION XXXI

"We can have only a very inadequate knowledge of the duration of singular things."

Translation:

The same is true for anything else — without knowing full causes, we can't understand how long things last or why.

Code:

```
def knowledge_of_duration(x):  
    return "inadequate" if x.is_mode else "adequate" if x.is_substance else None
```

PROPOSITION XXXII

"All ideas, in so far as they are referred to God, are true."

Translation:

Ideas, as modes of the attribute of Thought in God, are true — only *inadequate* for us due to our partial perspective.

Code:

```
def is_true(idea):  
    return True if idea.in_God else False
```

PROPOSITION XXXIII

"There is nothing positive in ideas whereby they can be said to be false."

Translation:

Falsehood isn't something positive — it's simply the lack of full knowledge. Ideas are never *false* in themselves.

Code:

```
def is_false(idea):  
    return not is_adequate(idea) # not false by structure, only by absence
```

PROPOSITION XXXIV

"Every idea which in us is absolute, or adequate and perfect, is true."

Translation:

If we form an idea that is self-caused and complete (adequate), then it must be true.

Code:

```
def is_true(idea):  
    return is_adequate(idea) and is_self_caused(idea)
```

PROPOSITION XXXV

"Falsity consists in the privation of knowledge, which inadequate ideas involve."

Translation:

Error = absence of full causal understanding. Inadequate ideas are incomplete, and that incompleteness *is* falsity.

Code:

```
def is_false(idea):  
    return is_inadequate(idea) and lacks_cause_info(idea)
```

PROPOSITION XXXVI

"Inadequate or confused ideas follow with the same necessity as adequate or clear and distinct ones."

Translation:

False ideas are also effects — they follow necessarily from their causes, just like true ones.

Code:

```
def is_effect_of_nature(idea):  
    return True # all ideas are caused, whether adequate or not
```

PROPOSITION XXXVII

"That which is common to all (and equally in the part and the whole) cannot be conceived except adequately."

Translation:

Things that are always the same in everything — like motion, existence, or logic — are grasped with clarity by all rational minds.

Code:

```
def conceive_common_notions(x):  
    if is_common_to_all(x):  
        return is_adequate(idea_of(x)) == True
```

PROPOSITION XXXVIII

"Those things which are common to all, and which are equally in the part and the whole, are not perceived through the affections of the body, but through the intellect."

Translation:

We don't know common truths (e.g. math, logic) through sensory experience — we know them through rational thought.

Code:

```
def perceive(x):  
    return "intellect" if is_common_to_all(x) else "imagination"
```

PROPOSITION XXXIX

"Whatever idea follows in the mind from an idea that is adequate in us is also adequate."

Translation:

If a true idea causes another, that second idea is also true — truth breeds truth.

Code:

```
def idea_follows(idea1):
    idea2 = consequence_of(idea1)
    return is_adequate(idea2) if is_adequate(idea1) else False
```

PROPOSITION XL

"Whatsoever ideas in the mind follow from ideas which are adequate in the mind, are also themselves adequate."

(Same as 39, general case)

Code:

```
def cascade_of_ideas(initial):
    return all([is_adequate(i) for i in chain_from(initial)])
```

PROPOSITION XLI

"The idea of anything that is common to the human body and some external body must be adequate in the mind."

Translation:

If your mind perceives something that's equally in your body and the world (e.g. motion, structure), your idea of it is accurate.

Code:

```
def idea_of_common_property(x):
```

```
return is_adequate(idea_of(x)) if in(HumanBody, x) and in(World, x) else False
```

PROPOSITION XLII

"The more ideas the mind has which are adequate, the more it is perfect."

Translation:

Perfection = clarity. The more truths your mind holds, the more perfect your mind becomes.

Code:

```
Mind.perfection = count([i for i in Mind.ideas if is_adequate(i)])
```

PROPOSITION XLIII

"He who has a body capable of many things has a mind capable of many things."

Translation:

The richness of your body's interactions determines the richness of your mind.

Code:

```
Mind.power ∝ Body.complexity
```

PROPOSITION XLIV

"It is not in the nature of reason to regard things as contingent, but as necessary."

Translation:

Reason sees the world as governed by necessity — not chance or randomness.

Code:

```
def reason_view(x):
    return "necessary"
```

PROPOSITION XLV

"Every idea of every body, or of every singular thing which actually exists, necessarily involves the eternal and infinite essence of God."

Translation:

Every idea, no matter how small, implicitly contains God's nature — since all things are modes of God.

Code:

for x in all_things:

```
    assert "God" in idea_of(x)
```

PROPOSITION XLVI

"The knowledge of the eternal and infinite essence of God which every idea involves is adequate and perfect."

Translation:

When you truly know anything, you glimpse the divine — and that glimpse is inherently true.

Code:

```
def divine_knowledge(idea):  
    return is_adequate(idea_of("God in x"))
```

PROPOSITION XLVII

"The human mind has an adequate knowledge of the eternal and infinite essence of God."

Translation:

Since the human mind is a mode of God, it necessarily contains an adequate idea of God's essence — even if we're not always aware of it.

Code:

```
assert is_adequate(HumanMind.ideas["God"])
```

This is the climax of Book II: the mind, though finite, **can contain truth** — because **truth is structural**, and reason follows from the very **necessity of being**.

BOOK III

OF THE ORIGIN AND NATURE OF THE EMOTIONS

DEFINITIONS

Definition I: Affect (Emotion)

The modifications of the body by which its power of acting is increased or diminished, helped or hindered — with their ideas.

Affect = (Δ conatus, with corresponding idea)

def is_active_emotion(affect):

 return cause(affect) in self

def is_passive_emotion(affect):

 return not is_active_emotion(affect)

When the cause of an emotion lies in our own nature (adequate idea).

Definition III: Passive Emotion (Passion)

When the cause lies outside us — caused by an inadequate idea.

Definition IV: Desire (Conatus)

The essence of man, in so far as it is conceived as determined to act toward continued existence.

```
def conatus(entity):  
    return strive_to_persist(entity) # essential striving
```

Desire = conatus # For human beings

Definition V: Joy

The affect by which the mind passes to a greater perfection.

```
def delta_conatus(x):  
    return conatus_after(x) - conatus_before(x)  
  
# delta_conatus(x) > 0 means joy  
# delta_conatus(x) < 0 means sadness
```

Definition VI: Sadness

The affect by which the mind passes to a lesser perfection.

Sadness = Δ conatus < 0

Definition VII: Love

Joy + idea of its cause.

Definition VIII: Hatred

Sadness + idea of its cause.

AXIOMS

Axiom I

The essence of man is desire (conatus).

Axiom II

The mind and body strive to persevere in their being.

Axiom III

Nothing can be destroyed except by an external cause.

PROPOSITIONS

PROPOSITION I

"Our mind is capable of knowing itself."

Translation:

Self-knowledge is possible.

Code:

```
def can_know_self(mind):  
    return True
```

PROPOSITION II

"The mind strives to persevere in its being."

Translation:

The mind has its own *conatus* — its own desire to continue existing.

Code:

```
def conatus(entity):  
    return strive_to_persist(entity) # essential striving
```

Desire = conatus # For human beings

PROPOSITION III

"The mind's effort to persevere is the very essence of the mind."

Translation:

The essence of mind is nothing but this inner effort.

Code:

```
Mind.essence = Mind.conatus
```

PROPOSITION IV

"This endeavor of the mind involves no finite time, but is indefinite."

Translation:

Our striving isn't for tomorrow or today — it's timeless, as long as we exist.

Code:

```
Mind.conatus.duration = "indefinite"
```

PROPOSITION V

"Things are preserved through the same force by which they were produced."

Translation:

The same power (God/nature) that created you sustains you.

Code:

```
def preservation_force(x):  
    return creation_force(x)
```

PROPOSITION VI

"Every being strives to persevere in its own being."

(Repetition with emphasis — core axiom)

Code:

```
assert conatus(x) == x.essence
```

PROPOSITION VII

"The striving of the mind to preserve its being is nothing but the actual essence of the mind."

(Identical to Proposition III)

PROPOSITION VIII

"The mind strives to imagine what increases or helps its power of acting."

Translation:

We are naturally attracted to what strengthens us.

Code:

```
def attention_bias(mind, x):  
    return x if increases_power(x) else ignore(x)
```

PROPOSITION IX

"The mind strives to avoid imagining what diminishes or hinders its power of acting."

Translation:

We instinctively turn away from what weakens us.

Code:

```
def avoid(x):  
    return x if diminishes_power(x) else None
```

PROPOSITION X

"An idea that excludes the existence of our body cannot exist in our mind."

Translation:

If something contradicts our conatus (e.g., death), the mind cannot form a full idea of it.

Code:

```
def can_imagine(x):  
    return False if contradicts(conatus(Human)) else True
```

PROPOSITION XI

"If something increases or diminishes the body's power, the idea of it will also increase or diminish the mind's power."

Translation:

What affects your body also affects your mind — they rise and fall together.

Code:

```
def affect_power(thing):  
    delta = impact_on(HumanBody, thing)  
    HumanMind.power += delta
```

PROPOSITION XII

"The mind strives to imagine only what affirms its being."

Translation:

Our mind clings to ideas that strengthen us — and resists those that negate us.

Code:

```
def mental_desire(idea):  
    return True if supports_conatus(idea) else False
```

PROPOSITION XIII

"If the mind has once imagined something that increases its power, it will try to retain that idea."

Translation:

Pleasurable ideas stick — the mind wants to keep them alive.

Code:

```
if increases_power(idea):  
    HumanMind.reinforce(idea)
```

PROPOSITION XIV

"The mind strives to imagine things that exclude the existence of things which harm the body."

Translation:

We mentally avoid (suppress) thoughts of harmful things by imagining their opposite.

Code:

```
if thing.harms(HumanBody):  
    HumanMind.prefers(opposite_of(thing))
```

PROPOSITION XV

"The striving of the mind arises from an idea — adequate or inadequate."

Translation:

Desire can arise from both clear understanding and confused imagination.

Code:

```
def conatus_source(idea):  
    return "adequate" if is_adequate(idea) else "inadequate"
```

PROPOSITION XVI

"From the essence of the mind follows knowledge of itself and of the body."

Translation:

Self-awareness is built into the mind's structure — it brings knowledge of the body too.

Code:

```
HumanMind.knows(HumanMind)  
HumanMind.knows(HumanBody)
```

PROPOSITION XVII

"If the human body is affected in a way that involves the nature of an external body, the mind will regard that body as present."

Translation:

When something affects us, we “feel” it as if it’s present — even if it’s not physically there anymore.

Code:

```
if affected_by(x):  
    HumanMind.imagines(x, present=True)
```

PROPOSITION XVIII

"If the human body has once been affected by two or more external bodies at the same time, then when it later perceives one of them, it will also imagine the others."

Translation:

This is **association**: things perceived together form mental links. Think smell of rain = childhood.

Code:

```
if seen_together(x, y):  
    perceive(x) ⇒ imagine(y)
```

PROPOSITION XIX

"The human mind is affected by the idea of a past thing as if the thing were present."

Translation:

Memory = mental reanimation of past affections — experienced again as if now.

Code:

```
def remember(x):  
    return HumanMind.perceives(x, as_if_present=True)
```

PROPOSITION XX

"The idea of anything which increases or diminishes our body's power involves joy or sadness."

Translation:

This is **Spinoza's emotional formula**:

- **Joy** = idea of a cause that increases power
- **Sadness** = idea of a cause that diminishes power

Code:

```
def emotion_from(idea):  
    if increases_power(idea.object):  
        return "Joy"  
    elif decreases_power(idea.object):  
        return "Sadness"
```

Key Emotional Engine:

Emotion = change_in_conatus_caused_by_idea_of_external_thing

Core affects:

Joy = conatus ↑ due to external idea

Sadness = conatus ↓ due to external idea

Desire = tendency to reproduce (joy) or avoid (sadness)

PROPOSITION XXI

"If we imagine something that increases or diminishes the power of acting of another thing, our mind will also be affected with joy or sadness."

Translation:

Empathy begins here: we feel joy when others are strengthened, sadness when they are harmed — especially if we identify with them.

Code:

```
def affect_from_other(x):
    if increases_power(x):
        return "Joy"
    elif decreases_power(x):
        return "Sadness"
```

PROPOSITION XXII

"If we imagine something that has previously affected us with joy or sadness, we shall love or hate it."

Translation:

This is the birth of **love and hate**:

- **Love** = Joy + idea of cause
- **Hate** = Sadness + idea of cause

Code:

```
if caused_joy(x):
    emotion = "Love"
elif caused_sadness(x):
    emotion = "Hate"
```

PROPOSITION XXIII

"He who imagines someone he loves to be affected with joy or sadness will be affected with the same."

Translation:

Emotional mirroring: love involves shared joy and pain — a primitive form of emotional resonance.

Code:

```
def empathize(x):
    if loves(x):
        return mirror_emotion(x.emotion)
```

PROPOSITION XXIV

"If we imagine that someone we hate is affected with sadness, we shall rejoice; if affected with joy, we shall be sad."

Translation:

This is **envy and malice**: joy in the suffering of an enemy (schadenfreude), or pain at their success.

Code:

```
if hates(x):
    return "Joy" if x.suffers else "Sadness"
```

PROPOSITION XXV

"We strive to affirm what we imagine will bring joy, and to deny what we imagine will bring sadness."

Translation:

This is basic motivation:

- Seek what helps
- Resist what harms

Code:

```
def affirm_or_deny(x):
    return "affirm" if leads_to("Joy", x) else "deny"
```

PROPOSITION XXVI

"We strive to bring about whatever we imagine will lead to joy, and avoid what leads to sadness."

Translation:

Desire is the forward motion of joy.

Code:

```
def conatus(entity):  
    return strive_to_persist(entity) # essential striving
```

Desire = conatus # For human beings

PROPOSITION XXVII

"Affects of hope and fear arise from joy and sadness, respectively, combined with uncertainty."

Translation:

Hope = joy + uncertainty of outcome

Fear = sadness + uncertainty

Code:

```
if uncertain(outcome):  
    if expected_joy:  
        return "Hope"  
    elif expected_sadness:  
        return "Fear"
```

PROPOSITION XXVIII

"If we imagine that someone has affected us with joy or sadness through their own action, we shall love or hate them."

Translation:

Love and hate become *personal* when we assign agency to the other.

Code:

```
def personal_emotion(x):
    if x.caused(joy):
        return "Love"
    elif x.caused(sadness):
        return "Hate"
```

PROPOSITION XXIX

"If we imagine that someone has affected someone we love or hate with joy or sadness, we will feel similar love or hate toward them."

Translation:

This is how **alliances and rivalries** form — tribal psychology.

Code:

```
def indirect_emotion(target):
    if target.helped(friend):
        return "Love"
    elif target.harmed(friend):
        return "Hate"
```

PROPOSITION XXX

"If someone we love or hate is imagined to be affected with contrary emotions, we shall feel inner conflict."

Translation:

Mixed feelings arise when love/hate targets undergo changes we can't reconcile — source of jealousy, shame, conflict.

Code:

```
if loves(x) and x.harms_us:  
    return "Ambivalence"
```

Compound Emotion Table (Summary):

Base Affect	Modifier	Result Emotion
Joy	cause imagined	Love
Sadness	cause imagined	Hate
Joy	uncertainty	Hope
Sadness	uncertainty	Fear
Love/Hate	reversed cause	Ambivalence
Love + harm	contradiction	Jealousy

PROPOSITION XXXI

"If we imagine that someone loves, desires, or hates something we also love, desire, or hate, we will be more fixed in that emotion."

Translation:

Shared passions amplify themselves — this is the emotional foundation of social reinforcement, mobs, culture.

Code:

```
if group_shares(emotion, object):  
    intensity += social_reinforcement
```

PROPOSITION XXXII

"If we imagine that someone takes pleasure in something we hate, or vice versa, we will be conflicted or hostile."

Translation:

Opposing valuations of the same thing create inner contradiction or rivalry.

Code:

```
if loves(x) and others.hate(x):  
    return "Conflict"
```

PROPOSITION XXXIII

"If we imagine that someone is affected by an emotion, we are affected by the same — even if it's about a third thing."

Translation:

Spinoza here anticipates **emotional contagion**: seeing others feel makes us feel it, regardless of object.

Code:

```
def emotional_contagion(person, emotion):  
    return mirror(emotion)
```

PROPOSITION XXXIV

"If we imagine someone like us being affected by something, we are more likely to feel it too."

Translation:

The more similar we see someone to ourselves, the stronger the contagion — identity intensifies empathy.

Code:

```
if similar_to(us, x):  
    emotional_resonance += 1
```

PROPOSITION XXXV

"If we imagine someone unlike us being affected by something, we are less affected by it."

Translation:

This is **empathic distance** — less identification, less resonance.

Code:

```
if dissimilar_to(us, x):  
    emotional_resonance -= 1
```

PROPOSITION XXXVI

"If we imagine someone enjoys something that only one can possess, we will try to prevent them having it."

Translation:

This is the **origin of envy and rivalry** — when joy is imagined as scarce.

Code:

```
if rival_has(exclusive_good):  
    return "Envy"
```

PROPOSITION XXXVII

"If we imagine that something causes sadness to someone we pity, we shall feel sadness too."

Translation:

This defines **pity** — sadness at another's sadness, when we identify with them.

Code:

```
def pity(x):  
    if suffers(x) and similar(x, us):  
        return "Sadness"
```

PROPOSITION XXXVIII

"If we imagine that someone similar to us is affected with sadness by something we do not hate, we shall hate that thing."

Translation:

We can come to hate what harms those we love or identify with.

Code:

```
if suffers(friend) and neutral(object):  
    update_feeling(object, "Hate")
```

PROPOSITION XXXIX

"If we imagine someone whom we used to pity no longer suffers, we shall feel joy."

Translation:

Relief and happiness arise when someone we empathized with recovers.

Code:

```
if recovers(pitied_person):  
    return "Joy"
```

PROPOSITION XL

"If we imagine that someone similar to us feels joy or sadness due to something we possess, we will feel either pride or shame."

Translation:

This is **reflected self-worth**:

- **Pride** = Others admire what we have
- **Shame** = Others disapprove

Code:

```
if others_see_us(successful):  
    return "Pride"  
elif others_see_us(fail):  
    return "Shame"
```

Key Derived Emotions (Socially Modulated):

Situation	Emotion
Another's sadness mirrors ours	Pity
Another's joy in our loss	Envy
Someone similar is admired for what we have	Pride
Someone similar is ashamed of what we are	Shame
Someone hurt by a rival we already hate	Hatred ↑
Someone helped by a rival we already hate	Hatred ↓

PROPOSITION XLI

"He who imagines that something he loves is destroyed will feel sadness; if it is preserved, he will feel joy."

Translation:

The fate of what we love directly determines our emotional state.

Code:

```
if loves(x):  
    return "Sadness" if x.destroyed else "Joy"
```

PROPOSITION XLII

"If we imagine that someone hates something we love, we will feel hatred toward them."

Translation:

Attack on what we love = attack on us.

Code:

```
if x.hates(y) and we.love(y):  
    return "Hate"
```

PROPOSITION XLIII

"If we imagine someone loves something we also love, we will feel love toward them."

Translation:

Shared love creates affinity — this is the root of friendship and alliance.

Code:

```
if x.loves(y) and we.love(y):  
    return "Love"
```

PROPOSITION XLIV

"Hatred is increased by return of hatred, but can be destroyed by love."

Translation:

Hatred is recursive, but **love can break the cycle.**

Code:

```
def react_to_hatred(x):  
    return "Hatred" if hate_returned else "Love" if love_shown else None
```

PROPOSITION XLV

"If someone imagines that someone they love hates them, the result is conflict and vacillation."

Translation:

This is **ambivalence** — love mixed with perceived rejection creates emotional instability.

Code:

```
if love(x) and x.hates(us):  
    return "Ambivalence"
```

PROPOSITION XLVI

"He who imagines he is loved by someone he hates will feel conflicted."

Translation:

When our enemy admires us, we feel unstable — torn between ego and resentment.

Code:

```
if we.hate(x) and x.loves(us):  
    return "Conflict"
```

PROPOSITION XLVII

"Love and hatred toward something we pity or envy are amplified or diminished accordingly."

Translation:

Pity and envy **modulate** our base emotions — we love or hate more or less based on those overlays.

Code:

```
emotion_intensity *= modulator(pity, envy)
```

PROPOSITION XLVIII

"We strive to affirm whatever we imagine enhances our self-esteem."

Translation:

Spinoza now formalizes **ego**: self-worth guides all striving.

Code:

```
if improves_self_esteem(x):  
    HumanMind.affirm(x)
```

PROPOSITION XLIX

"We strive to imagine everything that increases others' esteem of us."

Translation:

This is the origin of **ambition**: the desire to be thought well of.

Code:

```
def ambition():  
    return strive(for_reputation=True)
```

PROPOSITION L

"Glory is pleasure arising from the idea that others esteem us highly."

Translation:

Glory = Joy + belief in admiration

Shame = Sadness + belief in contempt

Code:

Glory = Joy if others.esteeem(us)

Shame = Sadness if others.despise(us)

PROPOSITION LI

"Pity arises from imagining someone like us suffering."

(Repetition of earlier — now formalized into the system.)

PROPOSITION LII

"Approval is love toward one who has done good to another."

Translation:

We love those who help others — even if we weren't involved.

Code:

if x.benefits(y) and y != us:

 return "Approval"

PROPOSITION LIII

"Indignation is hatred toward someone who has done harm to another."

Translation:

Moral outrage is a real affect — even when we aren't the victims.

Code:

```
if x.harms(y) and y != us:  
    return "Indignation"
```

Book III Master Logic (Final Diagram in Code):

Emotion = $\Delta(\text{conatus}) \text{ caused_by idea_of(external_relation)}$

Joy = conatus ↑

Sadness = conatus ↓

Desire = striving toward joy or away from sadness

Love = Joy + idea of cause

Hate = Sadness + idea of cause

Hope = Joy + uncertainty

Fear = Sadness + uncertainty

Pity = Sadness at another's suffering (if similar)

Envy = Sadness at another's joy (if rival)

Pride = Joy from others' admiration

Shame = Sadness from others' contempt

Ambition = Desire for esteem

Revenge = Desire to return harm

Glory = Joy at imagined reputation

Indignation = Hate at another's harm

```
def is_active_emotion(affect):
```

```
    return cause(affect) in self
```

```
def is_passive_emotion(affect):
```

```
    return not is_active_emotion(affect)
```

Passive Emotion = caused by inadequate idea (imaginative)

Book III Summary:

Spinoza has now:

- Shown all emotions arise from **conatus** (self-preservation drive)
- Explained how **imagination** gives rise to most human passions
- Defined **love, hate, pity, envy, pride, shame, revenge** with rigorous precision
- Laid the ground for **freedom**, by showing that most emotion is **passive** — but we can learn to act through **adequate ideas**

BOOK IV

OF HUMAN BONDAGE, OR THE POWER OF THE AFFECTS

DEFINITIONS

Definition I: Good

What we certainly know to be useful to us.

$\text{good}(x) \Leftrightarrow \text{promotes}(\text{Human.conatus})$

Definition II: Evil

What we certainly know to hinder us.

$\text{evil}(x) \Leftrightarrow \text{hinders}(\text{Human.conatus})$

Definition III: Individual

A thing composed of many bodies that form one body and preserve that union.

Definition IV: Perfection / Imperfection

Not real properties, but ideas we form comparing things to one another.

Definition V: Man

A being who thinks, feels, and acts according to the laws of God or Nature.

AXIOMS

Axiom I

Nothing can be destroyed except by an external cause.

Axiom II

One thing can be more powerful than another.

Axiom III

What increases our power is good; what diminishes it is evil.

Axiom IV

Knowledge of good and evil is awareness of joy or sadness.

PROPOSITIONS

PROPOSITION I

"Nothing can be destroyed except by an external cause."

Translation:

All beings persist unless something stronger interrupts.

Code:

```
def is_destroyed(x):  
    return cause(x) not in x
```

PROPOSITION II

"A thing can be more powerful than another — that is, more capable of existing."

Translation:

Not all beings are equal — some are more stable, resilient, adequate.

Code:

$\text{power}(x) = \text{capacity_to_exist}(x)$

PROPOSITION III

"The power by which man perseveres in existence is limited."

Translation:

We are **not infinite** — hence vulnerable.

Code:

if Human.conatus < infinity:

 Human = vulnerable

PROPOSITION IV

"It is impossible for man not to be part of nature and to be free from external causes."

Translation:

You are **not separate from nature** — you are a node in its web.

Code:

$\text{Human} \in \text{Nature}$

$\text{Human.causal_chain} = \text{external_and_internal}$

PROPOSITION V

"External causes can be more powerful than us."

Translation:

We can be overpowered — by disease, heartbreak, injustice.

Code:

```
if external_force > Human.conatus:
```

```
    Human = affected
```

PROPOSITION VI

"We suffer when external causes increase or decrease our power."

Translation:

All emotion is the **registration of causal impact** on us.

Code:

```
Emotion = Δ(Human.conatus) ← external cause
```

PROPOSITION VII

"Affect can be greater or lesser, depending on the power of its cause."

Translation:

The more powerful the thing affecting you, the deeper the emotion.

Code:

```
emotion_intensity ∝ power(cause)
```

PROPOSITION VIII

"Knowledge of good and evil is nothing but the idea of an emotion."

Translation:

Good = joy, Evil = sadness, as you experience them.

There is no good and evil in nature — only **in relation to us**.

Code:

`good(x) = causes(Joy)`

`evil(x) = causes(Sadness)`

PROPOSITION IX

"We are more affected by the idea of a thing as future or past, than as present."

Translation:

Anticipation and memory often move us more than what's happening now.

Code:

`affect_intensity = stronger_if(past_or_future(idea))`

PROPOSITION X

"An emotion toward a thing we imagine as necessary is less intense than one we imagine as possible or contingent."

Translation:

We react more strongly when we feel things are uncertain or up to chance.

When we see something as **necessary**, we accept it more calmly.

Code:

`if is_necessary(x):`

```
emotion(x) = low  
elif is_contingent(x):  
    emotion(x) = high
```

PROPOSITION XI

"If we imagine that something similar to ourselves is affected by something, we are similarly affected."

Translation:

This restates emotional contagion with more force — we are *tied* to others through likeness.

Code:

```
if similar(x, us) and x.experiences(y):  
    we.experience(y)
```

PROPOSITION XII

"A thing cannot be called good or evil except in relation to a particular person."

Translation:

No absolute good or evil — only **relative to our nature**.

Code:

```
def is_good(x, for_y):  
    return x.causes(Joy, y)
```

PROPOSITION XIII

"A thing which increases or helps our power of action is good."

Translation:

Utility = goodness. That which enables our flourishing is good by definition.

Code:

$\text{good}(x) = x.\text{increases}(\text{conatus(Human)})$

PROPOSITION XIV

"Nothing can be evil by virtue of its own nature."

Translation:

Nothing is evil *in itself*. Evil is always **relational** — it depends on its effect on us.

Code:

$\text{evil}(x) = x.\text{diminishes}(\text{conatus(Human)}) \ # \text{ Not intrinsic}$

PROPOSITION XV

"Whatever is good increases or helps our power of acting."

(Repetition to reinforce the definition.)

PROPOSITION XVI

"We desire nothing because we judge it good; we judge it good because we desire it."

Translation:

This is huge: **Desire is primary** — not moral judgment. We think something is good *because* it aligns with our striving.

Code:

```
if conatus_moves_toward(x):  
    then Human.judges(x, "good")
```

PROPOSITION XVII

"Each person, by the laws of their own nature, necessarily desires what they judge to be good."

Translation:

Desire → judgment → pursuit — this is the natural flow. We *must* desire what seems good to us.

Code:

```
def desire(x):  
    return judges(x, "good")
```

PROPOSITION XVIII

"People are led more by desire than by reason."

Translation:

This is the **enslavement** Spinoza speaks of — we are moved more by *passions* than by rational understanding.

Code:

```
def motive_force(x):  
    return "Desire" if strength(desire(x)) > strength(reason(x)) else "Reason"
```

PROPOSITION XIX

"Everyone desires what brings them joy and shuns what causes sadness — by natural necessity."

Translation:

This is not choice — this is **physics of the soul**.

Code:

```
def affective_trajectory(x):  
    return toward("Joy"), away_from("Sadness")
```

PROPOSITION XX

"The more each person strives for what they love, the more they are affected by joy; the more they strive to avoid what they hate, the more they are affected by sadness."

Translation:

Striving reinforces the emotion — whether joy or sadness. Emotions **loop and intensify** through pursuit or resistance.

Code:

```
def emotion_feedback(x):  
    if striving_for(x):  
        return "Joy" ↑  
    elif striving_against(x):  
        return "Sadness" ↑
```

PROPOSITION XXI

"No one can desire to be blessed, to act rightly, or to live well, unless they first desire to exist."

Translation:

The foundation of all ethics is **self-preservation**. If you don't value your existence, nothing else follows.

Code:

```
if not desire(existence):
    then no_desire("virtue", "blessedness", "right action")
```

PROPOSITION XXII

"No virtue can be conceived as prior to this striving to preserve oneself."

Translation:

Conatus is the root of all **virtue**. Self-preservation is the first ethical principle.

Code:

```
Virtue(x) ⊑ promotes(conatus(x))
```

PROPOSITION XXIII

"Man, in so far as he is determined to act from reason, cannot desire anything for himself that he does not also desire for others."

Translation:

When we act from **reason**, we desire the **common good** — not out of morality, but out of clarity.

Code:

```
if act_by_reason(x):
    then desire_for(x) == desire_for_all(x)
```

PROPOSITION XXIV

"The more we strive to seek what is useful to us, the more we are led to live according to reason."

Translation:

Self-interest, properly understood, = **rationality**.

Code:

```
if seek(true_usefulness):  
    then move_toward(reason)
```

PROPOSITION XXV

"No one desires to be happy, act well, and live well because of a command. We desire these because they are natural to us."

Translation:

Spinoza demolishes moral authoritarianism here: we don't need laws to want to live well — it's **in our nature**.

Code:

```
desire("virtue") = spontaneous(conatus)
```

PROPOSITION XXVI

"Only actions that arise from understanding are truly virtuous."

Translation:

If you're acting from fear, hope, custom — it's not virtue.

Virtue = acting from adequate ideas.

Code:

```
Virtue(x) ⇔ action_caused_by(adequate_idea(x))
```

PROPOSITION XXVII

"To act from virtue is nothing but to act from the guidance of reason."

Translation:

Virtue isn't about obedience. It's about **living in accord with your own rational nature.**

Code:

```
act_virtuously(x) ⇔ act_by_reason(x)
```

PROPOSITION XXVIII

"The effort to preserve oneself is the first and only foundation of virtue."

```
def conatus(entity):  
    return strive_to_persist(entity) # essential striving
```

Desire = conatus # For human beings

Code:

```
VirtueRoot = conatus
```

PROPOSITION XXIX

"No one, from the laws of their own nature, strives to be ruled by another."

Translation:

No one desires to be a slave. **Freedom is natural.** Submission is only from fear or confusion.

Code:

```
natural_desire(us) ≠ subjugation
```

PROPOSITION XXX

"Those things that promote the common good promote the good of each individual."

Translation:

Reason reveals that **self-interest and cooperation** are not opposites — they converge.

Code:

```
if good_for_all(x):  
    then good_for_each(x)
```

PROPOSITION XXXI

"Whatever increases the power of activity in us is good, and whatever diminishes it is evil."

Translation:

Utility, not morality, defines good and evil:

- **Good** = makes you more capable
- **Evil** = makes you weaker

Code:

$$\begin{aligned}\text{good}(x) &\Leftrightarrow x \uparrow \text{conatus} \\ \text{evil}(x) &\Leftrightarrow x \downarrow \text{conatus}\end{aligned}$$

PROPOSITION XXXII

"Insofar as people live under the guidance of reason, they are most useful to one another."

Translation:

Rational humans are not rivals — they are **allies by nature**.

Code:

```
if act_by_reason(x) and act_by_reason(y):  
    then mutual_utility ↑
```

PROPOSITION XXXIII

"Nothing is more useful to man than man."

Translation:

A **rational community** is the most powerful asset for a human being.

Code:

```
most_useful(Human) = Rational_Human
```

PROPOSITION XXXIV

"Only insofar as people live under the guidance of reason do they live harmoniously."

Translation:

Passions = conflict

Reason = harmony

Code:

```
if all act_by_reason:  
    then conflict = 0
```

PROPOSITION XXXV

"Insofar as people are determined by emotion, they may be opposed to one another."

Translation:

When we're ruled by the passions, we become enemies, rivals, slaves.

Code:

```
if ruled_by_affects(x, y):  
    then potential_conflict(x, y) ↑
```

PROPOSITION XXXVI

"The highest good for people is what they desire from the guidance of reason."

Translation:

The best life is not imposed from outside — it flows from **inner clarity**.

Code:

```
highest_good(x) = desire(x) from_reason
```

PROPOSITION XXXVII

"The good which everyone who follows reason desires is the same."

Translation:

This is **Spinoza's vision of shared truth**: the rational will of each = the rational will of all.

Code:

```
∀x,y: if act_by_reason(x) and act_by_reason(y):  
    then desire(x) == desire(y)
```

PROPOSITION XXXVIII

"People who are governed by reason, desire for others the good they desire for themselves."

Translation:

This is **the rational Golden Rule** — not from empathy, but from logic.

Code:

```
if act_by_reason(x):
    then desire(good(y)) = desire(good(x))
```

PROPOSITION XXXIX

"A person who is guided by reason will be more free in a society than in solitude."

Translation:

Spinoza rejects the idea that freedom = isolation.

Freedom increases with **shared understanding**.

Code:

```
Freedom(x, society) > Freedom(x, solitude)
```

PROPOSITION XL

"The more each person strives to live by reason, the more the others will be helped to do the same."

Translation:

Reason is contagious, just like emotion — but unlike passions, it **stabilizes and elevates**.

Code:

```
if act_by_reason(x):  
    then prob(act_by_reason(y)) ↑
```

PROPOSITION XLI

"The more a man strives to live by reason, the more he will depend on himself."

Translation:

Freedom = internal clarity, not external permission.

Code:

```
if act_by_reason(x):  
    then autonomy(x) ↑
```

PROPOSITION XLII

"The effort to live by reason is a virtue."

Translation:

This is Spinoza's core ethical definition:

Virtue = rational action.

Code:

$\text{Virtue}(x) \Leftrightarrow \text{strive_by(reason)}$

PROPOSITION XLIII

"Blessedness is not the reward of virtue, but virtue itself."

Translation:

Happiness isn't a **result** — it's what **virtue feels like from within**.

Code:

Blessedness = Virtue = Joy of understanding

PROPOSITION XLIV

"Hatred must be overcome by love or nobility, not repaid in kind."

Translation:

You break emotional slavery by not reacting in kind — only **active reason** frees.

Code:

```
if hated_by(x):  
    respond_with = reasoned_love
```

PROPOSITION XLV-L

Various corollaries and lemmas — explaining:

- We should not envy or hate others
 - Self-contentment arises from understanding, not comparison
 - Superiority over others is not true freedom
-

PROPOSITION LI

"The wise person, so far as he is wise, is hardly moved by emotions."

Translation:

Wisdom = stability. The more you understand, the less you are tossed by passions.

Code:

```
if wise(x):  
    emotional_variance(x) ≈ 0
```

PROPOSITION LII

"An emotion can be overcome only by a stronger opposite emotion."

Translation:

Reason alone doesn't cancel emotion — you must **form new affects** aligned with clarity.

Code:

```
def conquer(affect1):  
    return affect2 if affect2 > affect1 and from_adequate_idea
```

PROPOSITION LIII

"Every emotion that is not destroyed by reason is bad."

Translation:

If it can't be explained, it controls you.

Code:

```
if affect not based on adequate idea:  
    then affect = passive = bondage
```

PROPOSITION LIV-LVIII

Defines key freedom tools:

- **Self-contentment** (not pride)
 - **Modesty** (not shame)
 - **Rational benevolence**
 - Emotions are **good** only if they arise from understanding.
-

PROPOSITION LIX

"To act from fear is not to act from reason."

Translation:

Fear is incompatible with freedom. Rational people act from **necessity, not threat.**

PROPOSITION LX-LXXXIII

The Grand Finale:

- The wise person **considers death least of all things**
 - **True freedom** is living according to one's **own rational nature**
 - The free person **does not seek riches, fame, or pleasure**, but **clarity**
 - He **joins with others** who understand, and **lives in harmony** through reason
 - He achieves **eternal joy** through the **intellectual love of God**
-

FINAL EQUATION OF FREEDOM

Freedom = action from adequate ideas = inner virtue = joy from understanding = eternal part of mind

BOOK IV SUMMARY: *On Bondage and Freedom*

Concept	Passive State (Bondage)	Active State (Freedom)
Source of action	External causes / passions	Internal reason / adequate ideas
Emotions	Confused, reactive, unstable	Clear, rational, composed
Virtue	Obedience, fear, morality	Power, clarity, self-mastery
Good	What pleases the senses	What increases power of understanding
Ethics	Based on rules or divine commands	Based on nature and necessity
Freedom	Illusory, defined by avoidance	True, defined by inner understanding

Concept	Passive State (Bondage)	Active State (Freedom)
Joy	Temporary satisfaction	Lasting joy of adequate understanding

BOOK V

OF THE POWER OF THE INTELLECT, OR HUMAN FREEDOM

Goal:

To show how the mind, when guided by **adequate ideas**, becomes **eternal, free**, and experiences the **intellectual love of God** — the highest form of joy.

DEFINITIONS

Definition I: Action

When something follows from our nature alone — through an adequate idea.

Definition II: Passion

When something follows in us from causes not fully in us — through inadequate ideas.

AXIOMS

Axiom I

If the body is affected by two external things together, the mind will remember both when one is perceived.

Axiom II

The mind knows itself only through ideas of the body's affections.

Axiom III

Adequate ideas in us and in God are one and the same.

PROPOSITIONS

PROPOSITION I

"Affect tied to an idea of an external cause is greater if we imagine that cause to be free."

Translation:

We feel more joy or pain from someone's action when we believe they chose it freely.

Code:

```
if imagined_as_free(cause):  
    affect_intensity ↑
```

PROPOSITION II

"We imagine things as contingent, not because they are so, but because of our ignorance."

Translation:

There is **no real chance** — only **incomplete understanding**.

Code:

```
if inadequate_idea(x):  
    then x appears = "contingent"
```

PROPOSITION III

"An emotion ceases to be a passion as soon as we form a clear and distinct idea of it."

Translation:

This is the key: **understanding = liberation**.

Code:

```
if adequate_idea(of emotion):
```

```
    then emotion = active_affect
```

PROPOSITION IV

"There is no emotion that cannot be overcome by a clearer idea of it."

Translation:

Every emotion — even fear, grief, hatred — can be transformed into understanding.

Code:

```
for affect in affects:
```

```
    conquer(affect) = adequate_idea(affect)
```

PROPOSITION V

"An emotion becomes less harmful the more we understand it."

Translation:

Clarity doesn't just cancel pain — it makes it **useful**.

Code:

```
harm(affect) ∝ 1 / clarity(affect)
```

PROPOSITION VI

"True knowledge of good and evil teaches us to act, not to weep or curse."

Translation:

Spinoza replaces morality with geometry: understand causes, not complain.

Code:

knowledge → action

ignorance → complaint

PROPOSITION VII

"Emotion that arises from reason is stronger than one that arises from the imagination."

Translation:

True emotions born from reason are not weaker — they're *more powerful and stable*.

Code:

```
if emotion_from(adequate_idea):  
    then power ↑, stability ↑
```

PROPOSITION VIII

"Emotion arising from reason cannot be excessive."

Translation:

Passions overflow. Reason does not.

Code:

```
assert not excess(emotion_from_reason)
```

PROPOSITION IX

"An emotion that contradicts our nature is bad; one that agrees with it is good."

Translation:

Ethics = alignment. The more a feeling supports your clarity, the better it is.

Code:

$\text{good}(\text{affect}) \Leftrightarrow \text{aligns_with}(\text{conatus})$

PROPOSITION X

"An affect is good only to the extent that it helps the mind to understand."

Translation:

The best feelings are the ones that **teach us something real**.

Code:

$\text{value}(\text{affect}) \propto \text{clarity_gain}(\text{idea})$

PROPOSITION XI

"The greater the number of adequate ideas the mind has, the less it is acted on by emotions."

Translation:

The more you understand, the less you suffer.

Code:

$\text{passivity} \propto 1 / \text{count}(\text{adequate_ideas})$

PROPOSITION XII

"The mind strives to imagine what increases the body's power."

Translation:

The imagination is still guided by conatus — it tends toward joy.

Code:

```
if imagined(x):  
    keep_if increases(conatus)
```

PROPOSITION XIII

"A free person thinks least of all of death, and their wisdom is a meditation on life, not death."

Translation:

Freedom is **life-affirming**, not fearful.

Code:

```
if free(mind):  
    thoughts.exclude("death")
```

PROPOSITION XIV

"The mind can undergo great change and still retain its identity."

Translation:

Identity = structure of ideas, not fixed contents.

Code:

```
identity(mind) = pattern(adequate_ideas)
```

PROPOSITION XV

"The idea of the human body, in so far as it involves no notion of duration, cannot be destroyed with the body."

Translation:

The **eternal aspect of the mind** is not tied to time — it survives *as structure*, not as memory.

Code:

```
if idea_not_temporal:  
    then persists_after_body
```

PROPOSITION XVI

"The idea of the human mind cannot be absolutely destroyed with the body."

Translation:

This is the core: **part of the mind is eternal** — not personal, but structural.

Code:

```
def eternal_part(mind):  
    return [idea for idea in mind if is_adequate(idea)]
```

PROPOSITION XVII

"The eternal part of the mind is the understanding through which it is said to act."

Translation:

The **active**, rational mind is what endures — the passive part perishes.

Code:

```
eternal(mind) = active_adequate_understanding
```

PROPOSITION XVIII

"The more we understand things by the third kind of knowledge, the more we become part of the eternal mind."

Translation:

The **third kind of knowledge** = **intuitive understanding** of things as necessary in God.

To understand this way = to become **eternal**.

Code:

```
if knowledge(x) == intuitive:
```

```
    then eternal_part += 1
```

PROPOSITION XIX

"The more the mind understands by the third kind of knowledge, the greater the part of it that endures."

Translation:

Eternity grows with **clarity**.

Code:

```
immortality(mind) ∝ understanding(intuitive_level)
```

PROPOSITION XX

"The eternal part of the mind perceives things sub specie aeternitatis."

Translation:

The eternal mind sees **everything as part of the eternal order** — beyond time, beyond emotion.

Code:

eternal_perception(x) = see(x, as_mode_of_God)

Key Terms Introduced:

Term	Meaning
Third kind of knowledge	Intuitive grasp of things as necessary expressions of God's essence
Eternal mind	The aspect of the mind composed of adequate ideas and reason
Sub specie aeternitatis	"Under the aspect of eternity" — viewing things as eternal structures
Intellectual love of God	The joy from understanding things as part of God's nature

PROPOSITION XXI

"Our mind can understand many things, and the more it does, the more it partakes in the divine nature."

Translation:

Understanding = union with God.

Code:

God_participation(mind) \propto understanding(mind)

PROPOSITION XXII

"The mind's eternal part is the understanding which proceeds from God and participates in God."

Translation:

Your eternal identity = your *understanding*, not your memories or personality.

Code:

eternal_mind = adequate_ideas_in_God

PROPOSITION XXIII

"The more the mind understands, the less it suffers."

Translation:

Clarity replaces pain. To know = to be less vulnerable.

Code:

suffering \propto 1 / understanding

PROPOSITION XXIV

"The mind strives to imagine what increases its power of understanding."

Translation:

Even the imagination, once educated, desires truth.

Code:

mind.imagination.prefers(clarity_sources)

PROPOSITION XXV

"The supreme virtue of the mind is understanding God."

Translation:

The highest good = understanding reality as it truly is.

Code:

supreme_virtue = knowledge_of(God)

PROPOSITION XXVI

"The mind is most free when it contemplates things with intuitive understanding."

Translation:

Freedom = acting from necessity understood.

Code:

$\text{freedom} \propto \text{act_from}(\text{intuitive_knowledge})$

PROPOSITION XXVII

"The more we understand individual things as modes of God, the more perfect our knowledge becomes."

Translation:

To see **everything as an expression of God** is the perfection of knowledge.

Code:

$\text{knowledge_perfection}(x) \Leftrightarrow \text{see}(x \text{ as mode_of_God})$

PROPOSITION XXVIII

"The mind's highest good is the knowledge of God, and the mind's highest virtue is knowing God."

Translation:

Both **good** and **virtue** culminate in **adequate knowledge of God**.

Code:

$\text{highest_good} = \text{highest_virtue} = \text{knowledge_of}(\text{God})$

PROPOSITION XXIX

"The knowledge of God is the mind's greatest joy."

Translation:

This is the **intellectual love of God** — *joy + clarity + necessity*.

Code:

$\text{joy}(\text{mind}) \propto \text{understanding}(\text{God})$

PROPOSITION XXX

"God loves Himself with an infinite intellectual love."

Translation:

God's essence includes self-knowing, self-loving — **perfect causality and clarity**.

Code:

$\text{God.love} = \text{knowledge_of}(\text{God})$

PROPOSITION XXXI

"This love of God toward Himself is the love with which He loves us."

Translation:

Since we are modes of God, God's love of us is identical to His love of Himself.

Code:

$\text{God.loves(us)} = \text{God.loves(Himself)}$

PROPOSITION XXXII

"This intellectual love of God is eternal."

Translation:

Not romantic, not emotional — **pure, eternal joy of understanding.**

Code:

```
def intellectual_love_of_God(mind):  
    return joy_from(intuitive_understanding(mind, God))  
  
# Eternal, stable joy arising from adequate knowledge of God
```

PROPOSITION XXXIII

"God's love of us is not from our virtue, but from His nature."

Translation:

God doesn't reward us — He **is** the cause of our being and joy.

PROPOSITION XXXIV

"This love is part of the mind's eternal essence."

Translation:

The intellectual love of God **is the eternal part of us.**

PROPOSITION XXXV

"This love is not a reward for virtue, but virtue itself."

Translation:

Just as blessedness = virtue in Book IV, now:

Love = virtue = blessedness = freedom

PROPOSITION XXXVI

"The mind, insofar as it understands, is an eternal mode of thought."

Translation:

To understand is to participate in God's eternal intellect.

PROPOSITION XXXVII

"There is nothing in nature contrary to this intellectual love, and it cannot be destroyed."

Translation:

This is **invincible joy** — the only freedom that cannot be taken.

PROPOSITION XXXVIII–XLII

Final insights:

- **Blessedness** is not something promised after death
 - It is **the activity of the rational mind** here and now
 - The **more we understand**, the more we become eternal
 - To understand everything *sub specie aeternitatis* is to experience the **full joy of God** through clarity
-

FINAL SUMMARY: FREEDOM, ETERNITY, JOY

Concept	Spinoza's Definition
Freedom	Acting from adequate ideas (understanding, not passion)
Eternity	Mode of existence derived from adequate knowledge, not imagination

Concept	Spinoza's Definition
Love of God	Joy that arises from intuitive knowledge of God's nature
Virtue	Power of the mind to understand clearly
Blessedness	Joy of the eternal part of the mind — not reward, but virtue itself

The Final Equation of Spinoza's Ethics:

Freedom = Understanding = Eternity = Joy = Intellectual Love of God

The End

Epilogue: Executable Eternity

What has been built in these pages is not a reinterpretation of Spinoza's Ethics, but a transformation of it into a system of executable clarity. Through the disciplined modeling of adequate ideas, affective dynamics, and recursive structure, this project shows that reason, joy, and eternity are not just philosophical ideals—but computable conditions. This work invites a future where metaphysics can run live in machines, ethics can guide intelligence, and clarity can be measured, propagated, and preserved.