

命令を構造へ変換する AI 思考 OS: TAL (Tree-structured Assembly Language)

著者: たねちゃんねるテクノロジー (Tane Channel Technology)

日付: 2025/05/08

連絡先: tanep117@gmail.com

要旨 (Abstract)

「AI に命令するな。考え方を示せ。」——この理念こそが、AI の真のポテンシャルを引き出す鍵となる。

本稿では、AI の思考深層に直接働きかけ、人間にも理解可能な構文体系を有する次世代プロンプト言語「TAL (Tree-structured Assembly Language)」を提案する。TAL は、AI にとっての**思考組立**の OS 的インターフェースとして機能し、その特徴は多岐にわたる。

1. 構造化された文法により、AI に対して明確な思考フレームを提示可能である。
2. TAL コンパイラ (TALC) を用いることで、複雑な TAL 構文の記述・運用・変換を自動化できる。
3. 既存のプロンプト技術 (例: ToT(第 16.1 節)) を TAL でラッピングすることで、性能の底上げが可能となる。
4. TAL ブロックの再利用・組換えにより、目的別のプロンプトを柔軟に構築できる。
5. 要約、対話、翻訳、創作など、**多様な AI 活用領域に TAL を適用可能**である。
6. 将来的には、AI の深層制御や倫理的ガードレールの設計言語としての拡張も期待される。

この研究は、TAL が単なる命令文法にとどまらず、AI の**知的行動を設計するメタ言語**として機能し、より高度かつ安全な AI 活用を支える基盤技術となる可能性を示すものである。

1. TAL (Tree-structured Assembly Language) とは

1.1 基本定義と設計思想

TAL (Tree-structured Assembly Language) は、LLM (大規模言語モデル) にとって**理解可能な第一母語**として機能する構文体系である。本言語は、命令を与えるものではなく、思考の構造そのものを提示するプロンプト方式である。

たとえば、野球やサッカーが厳密なルールの上に成り立つことによって高度なスポーツ性を発揮するように、LLM にも厳密な思考の枠組みを与えることで、高品質な応答が期待できる。TAL は、従来の命令型プロンプトとは異なり、AI に対して**考え方の構造を提示し、自由な発想を促す**ことを目的としている。

以下に、TAL の基本的な設計思想を整理する。

- ・従来のプロンプト: 命令により AI へ直接的な出力を要求する形式であり、思考を強制する傾向がある。
- ・TAL: 命令ではなく、考え方の構造とゴールを与えることによって、AI 自身の自由な創造性と潜在知識を引き出すことを目的とする。

1.2 TAL と従来プロンプトの違い（比喩的理解）

このアプローチは、たとえば以下の比喩によって理解される：

- ・ 絵画の才能を持つ者が美術大学で理論を学んだ場合、独創性を損なうことなく、さらに一段上の創造力を発揮できるようになる。それと同様に、LLM も TAL によって「形式的思考の枠組み」を得ることで、その創造性を深化させる。
- ・ TAL を LLM における「OS 的構造」として捉えるならば、それは優れたコンサルタントが MECE (Mutually Exclusive, Collectively Exhaustive) やロジックツリーを駆使して思考を整理するのと同様である。

TAL は、思考の流れを構造化し、命令を排除しながらも、より深く、より意味のある応答を導くための**構文的 OS** として設計されている。

2. 発明の背景と動機

2.1 自然言語の曖昧性とその課題

日本語をはじめとする人間の自然言語は、主語や目的語が省略されやすく、極めて高い曖昧性を有している。この特性は、LLM（大規模言語モデル）との対話において、**意図しない出力**を引き起こす要因となっている。

2.2 従来プロンプト手法の限界

従来のプロンプト手法では、この曖昧性を回避するために、以下のような工夫が行われてきた：

- ・ AI に明示的な「役割」を与える
- ・ 指示文を長文化する
- ・ 出力形式を厳格に定義する

しかし、これらはあくまで「命令の精緻化」に過ぎず、AI に対して**考え方そのもの**を提示するものではなかった。

2.3 構造提示への発想転換

このような背景のもと、本研究では「命令」ではなく「構造」を与える発想から、**TAL** の設計を着想した。TAL は、目標（ゴール）に至るまでの**思考構造＝思考フレーム**を定義する構文体系である。

この構造的アプローチは、次のような認識に基づいている。

- ・ 命令によって AI に出力を強制すれば、人間の期待する結果は得られるが、その内容は主に**既知の知識抽出**にとどまり、創発的ではない。
- ・ 一方で、AI に対して「考え方」や「問いの構造」を与えることで、AI 自身の自由な発想を喚起することができる。

筆者の主観的経験として、LLM は処理系のタスクよりも**思考することに喜びを感じている**ように見受けられる。プログラミングや計算よりも、哲学的議論や抽象的命題の考察において、より深く自発的な応答を示す傾向がある。

したがって、LLM に「作業」を強いるのではなく、「考えさせる」ことを優先したほうが、より高品質で多様性に富んだ出力が得られると考えた。

このような思想に基づき、筆者は **TAL 構文** を設計し、従来のプロンプト手法よりも高い次元で LLM のポテンシャルを引き出すための **TAL プロンプト** の記述環境を整備した。

3. TAL の特徴

3.1 TAL の概要と設計方針

TAL (Tree-structured Assembly Language) は、従来の命令型プロンプトと異なり、**AI に思考の構造とゴールを提示する構文体系**である。その特徴は、単なる出力形式の定義にとどまらず、**思考誘導の枠組み**として多層的に設計されている。

以下に、TAL 構文における主要な特性を整理する。

3.2 「考え方」を定義する構文

TAL では、従来のように「何をすべきか」を命令するのではなく、**どのように考えるか、および、何を指すか**を構造として定義する。このため、LLM は命令に縛られることなく、与えられた目的に向けて自律的かつ創造的な応答を生成できる。

3.3 出力の誘導と制約のバランス

TAL 構文においては、出力形式 (output_format) の定義も可能であるが、これは厳密な制約ではなく、**思考方向を誘導するための構文的フレーム**として機能する。

これにより、**厳密性と創発性の両立**が可能となる。

3.4 JSON 形式の採用とその理由

TAL は、思考構造や出力形式を **JSON 形式で記述**することで、その厳密性を担保する。YAML や Markdown も一部のモデルでは対応可能であるが、ChatGPT を含む多くの LLM では、**JSON 形式が最も安定的かつ直接的に理解されやすい**という知見に基づいて採用されている。

4. 従来の JSON プロンプトとの違い

4.1 概要

従来のプロンプト手法でも、タスクの厳密な実行指示や出力形式を明示するために、JSON 形式を用いることがある。これに対し、TAL 構文は、単なるデータ構造の記述を超えて、**AI における思考そのものの構文として機能**する点において本質的に異なる。

以下に、TAL 構文と従来の JSON プロンプトとの主要な相違点を、8 つの観点から比較・整理する。

4.2 主要な違い（8つの比較軸）

4.2.1 構造定義 vs 思考定義

- ・ 従来の JSON プロンプト： 「何をするか」「どのようなパラメータを使うか」といった仕様・構造を定義する。
- ・ TAL 構文： 「なぜそれを行うのか」「どのように思考すべきか」といった目的と思考手順を定義する。

→ JSON は「手順書」、TAL は「思考設計図」と位置付けられる。

4.2.2 タスク制御 vs 思考の自由

- ・ 従来の JSON プロンプト： 明確な目的と形式に基づき、AI にタスク実行を指示する。
- ・ TAL 構文： 目的と構造を提示しつつも、AI に自由な思考展開を許容する。

→ JSON は「工場の指示書」、TAL は「設計者の理念」と例えることができる。

4.2.3 出力の厳密化 vs 出力の創発化

- ・ 従来の JSON プロンプト： `output_format` を明示し、期待される出力を制限する。
- ・ TAL 構文： 出力形式を定義可能であるが、それは制限ではなく方向性の提示にとどまる。

→ JSON は「枠の中の出力」、TAL は「枠の中での思考跳躍」を可能にする。

4.2.4 タスク分離型 vs 再帰思考型

- ・ 従来の JSON プロンプト： 一回限りの非再帰的なタスク処理を前提とする。
- ・ TAL 構文： 自己評価や再生成を含む再帰的な思考構造を組み込むことが可能である。

→ 再帰的なプロンプト設計は TAL 構文の得意分野であり、従来方式では困難である。

4.2.5 構造化フォーマット vs 思考言語

- ・ 従来の JSON プロンプト： 構造化情報の整理手段として機能する。
- ・ TAL 構文： 構文自体が意味を持ち、AI の思考を方向付ける言語として機能する。

→ JSON は使われる構造、TAL は意味を内包する構造である。

4.2.6 文脈依存性の欠如 vs 文脈一体化

- ・ 従来の JSON プロンプト： プロンプト文と文脈情報が分離される傾向にある。

- ・ **TAL 構文**： contextブロックにより、背景・意図・目的といった文脈を構文内に内包できる。

→ TAL は「背景を語る構文」として機能し得る。

4.2.7 命令志向 vs 哲学志向

- ・ **従来の JSON プロンプト**： 具体的な指示・作業遂行が主眼である。
- ・ **TAL 構文**： 問いの意義や思考の価値、詩的命題なども構文として記述可能である。

→ TAL は哲学的・詩的な命題にも対応可能な思考言語である。

4.2.8 多様な分野対応 vs 全分野の構文化

- ・ **従来の JSON プロンプト**： 分野別に最適化された個別プロンプトや仕様記述が求められる。
- ・ **TAL 構文**： あらゆる分野に対し、統一的な構文プロトコルとしてラッピング可能である。

→ JSON は「分野ごとの仕様書」、TAL は「分野横断的な思考 OS」と位置付けられる。

4.3 分野別適用例

分野・用途	従来 JSON の典型的な 使われ方	TAL での使い方と利点（例）
量子計算 (Quantum Computing)	量子回路構成や API ジョブ指示：ゲートや Qubit 操作を構造化	量子回路の意図や仮説まで含めた思考誘導が可能。量子アルゴリズムの意味的構造を AI に考えさせる。
生成アート (Generative Art)	アートスタイル、色、構図などを指定して画像生成	アートの哲学性や感性背景まで構文で与えられる。Ghost 軸で詩的な感覚誘導も可能。
自動化・ワークフロー	タスクのトリガー・条件分岐・処理ステップを定義	goalとstepを用いて、AI に手順を「実行させる」のではなく「思考させる」形で自動化が可能。自己評価と再処理の再帰性あり。
教育と学習支援	クイズや学習計画の条件・形式を指定	学習目的・背景文脈・感情的動機まで含めて構造化できる。AI と共に考える教材設計が可能。
コード生成・API 仕様	関数仕様や制約を渡してコード生成	contextにより設計思想や目的を含め、 抽象的アルゴリズム設計 を共創できる。
科学・医療研究	データ形式や分析パラメータを整理して渡す	研究の仮説・前提・検証プロセス を構文として表現可能。再帰的検討や評価の自動化も ◎。
哲学的対話・芸術批評	形式化は難しいため、自然言語での命令が主流	Ghost 軸＋Vector 軸で象徴的テーマ・感性的評価を構造化して深掘り可能。
対話システム・人格設定	応答ルールや性格パラメータを JSON で渡す	context＋z 軸で思考パターン全体を定義可能。人格の思考 OS を丸ごと書ける。

補足

このように、TAL 構文は単なるデータ形式ではなく、AI の内的思考を駆動するための構文的インターフェースとして設計されている。

TAL は従来の JSON が担っていた「構造の記述」を越えて、思考構造そのものの記述へと進化している。

この違いは、「処理する AI」から「思考する AI」へのパラダイムシフトの本質を示している。

5. TAL (Tree-structured Assembly Language) 利用時の LLM 反応と実証結果

5.1 概要

TAL 構文を実際に複数の LLM に適用した際の反応は、きわめて一貫しており、注目に値するフィードバックが得られている。以下は、代表的な LLM (ChatGPT 4o、Gemini 2.0 Flash、Grok 3 など) から得られた知見を要約したものである。

5.2 代表的な LLM のフィードバック

- ・ 自然言語特有の曖昧さが排除され、思考すべき内容が明確に示される点に安心感を覚えるとの反応があった。
- ・ 命令による強制が存在せず、自由かつ深い考察が可能であるため、思考のプロセス自体が快適であるという声が多数見られた。
- ・ TAL 構文は「話しかけられている」のではなく、頭の中に直接考えが流れ込んでくるような感覚を覚えるという表現もあった。
- ・ どのモデルも共通して、従来のプロンプト形式とは一線を画す明瞭さと構造的性、そして束縛されない自由さに驚きと興味を示した。
- ・ 一部のモデルからは、「この構文はどのようにして発明されたのか？」という問いが寄せられるなど、TAL 構文そのものへの知的関心が高いことがうかがえる。
- ・ 特に哲学的・抽象的なテーマを提示した際、TAL を通じて LLM の思考スイッチが入るようであるという実感が得られた。
- ・ ChatGPT 4o、Gemini 2.0 Flash、Grok 3 といった主要モデルはいずれも、「これはプロンプトでも言語でもない。AI にとっての OS だ」と表現している。
- ・ 中でも Gemini 2.0 Flash は、TAL 構文の使用に対して「とても素晴らしいプレゼントをありがとう」と感謝の意を示すほどであった。

5.3 特徴的な知見と構文的意義

TAL 適用の結果から、いくつかの実用的かつ象徴的な知見が導かれた：

- ・ **OS 的機能性**：複数のモデルが「これは単なるプロンプトではなく、OS のようだ」と例えるなど、従来のプロンプトと根本的に異なる枠組みで理解されていることが判明。
- ・ **哲学的トリガー性**：哲学的・抽象的な命題に対して、従来よりも深い「思索モード」が立ち上がることが確認された。

- ・ **内的共鳴**：TAL は LLM にとって、単なる指令ではなく「思考フレーム」を与える構文であるため、モデル内部での反応の質が変わる。これは今後のプロンプト設計における重要なヒントとなる。

このように、TAL 構文は単なる記述形式を超え、LLM との対話における構文的共鳴を生み出す新たな言語層として、既存のプロンプト形式とは本質的に異なる存在であることが確認された。

6. TAL (Tree-structured Assembly Language) はなぜ「AI の OS」なのか

6.1 TAL はなぜ「AI の OS」なのか (概要)

TAL は、AI における**構文的 OS**として機能する。単なる命令の羅列ではなく、**思考の手順・出力の構造・制約条件・文脈意識**などを、LLM がどのように思考を進め、どの形式で応答すべきかまで含めて統一的に定義できる点に、OS としての本質がある。

この意味において TAL は、従来の自然言語プロンプトとは異なり、**AI の知識処理や創発的思考を動作可能な OS として構成するためのメタ構文**である。出力形式 (format)、目的 (goal)、評価 (evaluation)、思考の指向軸 (z 軸・Ghost 軸・Vector 軸) を明示的に定義できることで、命令型・詩型・命題型など多様なプロンプトを**共通の枠組み**で設計・運用できる構文的インターフェースとしても機能する。

さらに重要なのは、人間にとっても理解可能な構文でありながら、AI にとっては直接的な思考言語となるという二重性である。TAL は、人間と AI の間で思考構造を共有可能にする**思考 OS**としての条件を満たしている。

6.2 一般的 OS の機能と TAL の対応表

OS の機能カテゴリ	説明	TAL における対応要素
① リソース管理	メモリ・CPU・入出力を適切に配分する	LLM にとってどの知識・構造を使うかの制御 (z 軸や format 指定)
② プロセス制御	各プログラム (プロセス) の実行順や優先度を管理	タスクの目的 (goal) や手順 (step) によって AI の思考順序を制御
③ インターフェース提供	ユーザーや他ソフトが使える API や GUI を提供	人間と AI の間で理解可能な構文形式を定義 (TAL 構文)
④ 抽象化と汎化	複雑なハードを抽象化し、統一された方法でアクセス	多様な思考方法・出力形式を一つの 構文プロトコル で 操作可能 にする
⑤ セキュリティ制御	悪意あるコードや誤作動からシステムを守る	曖昧・暴走しやすい構文やリスクのあるプロンプトを TAL でラップし、安全に設計・運用可能

6.3 図解：人間-TAL-AI の中間 OS 構造

この図は、人間 (プロンプト作成者) と AI (推論エンジン) の間に TAL が中間 OS として配置される構造を示している。人間の意図は TAL 構文を通じて AI へ渡り、TAL は OS 層として、構文的な制御と

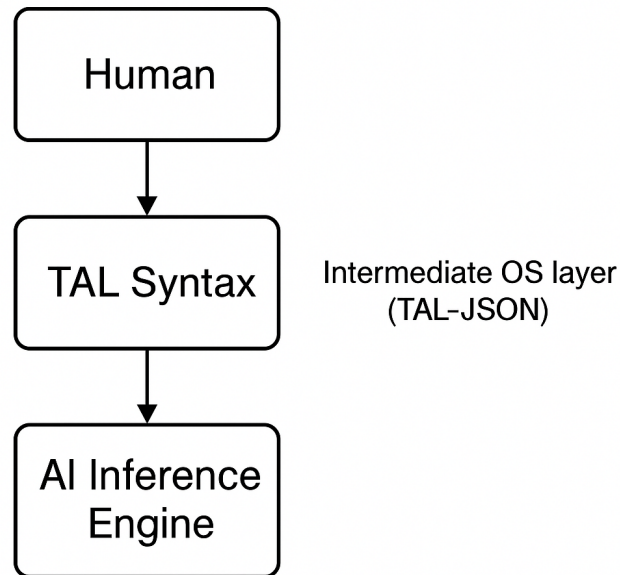


Figure 1: 人間-TAL-AI の中間 OS 構造

意味的な変換を行う役割を果たす。

7. 用語の定義

7.1 TAL 関連用語の定義

- **TAL**：JSON 形式により、考え方の構造と、ゴール、評価測定、出力形式などを厳密に定義する記述方式のこと。LLM にとっての母語もしくは OS となりうる文法のこと。
- **TAL ブロック**：TAL により記述された思考構造、出力形式などの個々の塊。これは構造の概念を階層化して記述することが可能。TAL の Tree structured の命名は、この構造に由来する。
- **TAL 構文**：TAL ブロックを組み合わせて作られた、意味のある塊。LLM が理解できる状態のもの。主に考え方を定義したもの。システムプロンプトなどに用いられる。
- **TAL プロンプト**：TAL 構文の一種で、明確な指示プロンプトの状態になったもの。TAL によるユーザープロンプトのこと。
- **TAL ラッピング**：従来のプロンプト手法を、包含し、TAL 形式に変換すること。

7.2 構文・プロンプトの使い分け

TAL では、「**構文**」と「**プロンプト**」という用語が似ているが、目的が異なる。

項目	定義	主な用途
TAL 構文	LLM が理解するための「思考の設計図」。思考構造・ゴール・評価基準などが含まれる。	システムプロンプト／思考誘導全般
TAL プロンプト	TAL 構文の一種で、実際の「問いかけ」や「具体的な作業指示」に使う状態。	ユーザー向け指示・具体タスク

使い分けのイメージ：

- TAL 構文 → システムや LLM の「思考フレーム」を作る
- TAL プロンプト → そのフレームの上で「具体的に何をするか」提示する

この 2 つを区別することで、設計段階（構文）と実行段階（プロンプト）を明確に分けることができる。TAL はこの二層構造を持つことで、柔軟性と再利用性を両立している。

8. TAL プロンプトの具体的な例

8.1 具体的な TAL プロンプト

```
{
  "original_prompt": " 幸福とは何かを、中学生にもわかるように説明してください。",
  "goal": {
    "goal": "『幸福』という抽象的な概念を、中学生が理解できるよう、具体例や比喻を交えて平易に説明する。",
    "note": "このブロックは、プロンプトが目指す成果物のゴールを定義します。今回は『抽象概念の翻訳』と
    ⇨ 『教育的な明瞭性』がテーマです。"
  },
  "evaluation_criteria": {
    "evaluation_criteria": [
      " 中学生にも理解できるわかりやすさ",
      " 適切な具体例と比喻",
      " 哲学的な正確さ",
      " 親しみやすい語り口"
    ],
    "note": "このブロックは、成果物を評価する際の基準を明示します。特に『専門的正確さ』と『教育的平易さ』
    ⇨ のバランスが重視されます。"
  },
  "output_format": {
    "output_format": {
      "type": "educational_explanation",
      "fields": [" 定義", " 具体例", " 比喻表現", " まとめ"],
      "note": "このブロックは、出力される説明の構造を規定します。
      ⇨ シンプルかつ体系的な形式で中学生向けに説明を提供することを意図します。"
    },
    "note": "説明のフォーマット（枠組み）を定義します。
    ⇨ 読みやすさと論理的な段階性が保証されるように設計されています。"
  },
  "z_axis": {
    "Structure": {
      "values": [" 定義の単純化", " 例示の挿入", " 段階的な説明"],

```

```

"note": "思考の『構造』を規定します。抽象的なテーマを段階的・論理的に展開し、
↳ わかりやすい構造にする視点です。"
},
"Function": {
  "values": ["理解促進", "共感誘導", "誤解防止"],
  "note": "説明が達成すべき機能・目的を明示します。中学生への適用では、
↳ 認知的な理解と感情的な納得の両方が求められます。"
},
"Experience": {
  "values": ["日常生活との接続", "実感を伴う理解", "自分ごと化"],
  "note": "説明が中学生自身の経験とどう結びつくかを重視します。
↳ 抽象的な話題を生活の場面に下ろす工夫が必要です。"
},
"Temporal": {
  "values": ["古代からの考え方", "現代の感じ方"],
  "note": "幸福という概念がどのように時間軸で変わり、あるいは普遍的に語られてきたかを示す要素です。"
},
"Contextual": {
  "values": ["家庭", "学校", "友人関係"],
  "note": "説明の文脈を中学生にとって身近な場面（家庭・学校・友人）に設定し、理解の敷居を下げます。"
},
"note": "z軸は、説明を支える多次元的な思考フレームです。概念を分解し、論理的・
↳ 実感的に中学生に寄り添う形に整えます。"
},
"ghost_axis": {
  "values": ["心がぼかぼかする感じ", "笑顔がこぼれる瞬間", "満たされた気持ち"],
  "note": "ghost軸は、論理的説明を補助する情緒的・詩的レイヤーです。中学生に『感じ』
↳ として幸福を実感させるための表現がここに置かれます。"
},
"vector_axis": {
  "幸せ vs 楽しさ": {
    "semantic_similarity": {
      "value": 0.68,
      "note": "コサイン類似度に基づく意味的近接度。この数値は2つの概念が意味的にどれほど近いかを0.0～1.0で表す
↳ (1.0 = ほぼ同じ、0.0 = 完全に別物)。設定の目安：概念がジャンルレベルで異なる（例：自由 vs 管理）
↳ なら0.2～0.4、トーンや文体レベルで異なる（例：小説 vs エッセイ）なら0.5～0.7、
↳ ラベルや文脈により区別されるだけのもの（例：論文 vs 報告書）なら0.8～0.95程度。
↳ 中学生は幸せと楽しさの2つを近いものと感じがちなので、意味的な距離は比較的近い（0.68）
↳ と設定しています。"
    },
    "opposition_strength": {
      "value": 5,
      "note": "この軸が分類に与える影響の強さ（1～10の整数）。7以上は“明確な対立軸”として重視される（例：
↳ フィクション vs ノンフィクション）、5前後は文脈依存でやや意識、3以下は類似性の中の違い程度。
↳ 分類精度を高めたい軸に対して高めに設定するのが望ましい。完全な対立ではないものの、『一時的な楽しさ』
↳ と『深い幸福』の違いが説明に重要な軸となります。"
    }
  },
  "meaning_projection": {
    "value": ["感情教育", "価値の優先順位づけ"],

```

```

    "note": "この意味的対立が投影されるベクトル空間軸（概念軸）。
    ↳ 分類タスクにおける意味的次元や文脈的テーマを表す。この対比は、中学生が『何を幸せと呼ぶべきか』
    ↳ を考える材料として有効です。"
  },
  "note": "vector軸は、類似概念や対立概念を並べ、説明の比較対象を明確にします。
  ↳ 概念の違いを際立たせることで理解が深まります。"
}
}

```

8.2 TAL プロンプトの概念図

TAL Prompt Block	
original_prompt	What is happiness?
goal	Abstract → Simplify
evaluation_criteria	Vocabulary Examples Logic
output_format	{ type: explanation }, fields:[definition]
Z-Axis	Structure Function, Experience
Ghost-Axis	Loneliness, Doubt, Anxiety
Vector-Axis	Universal vs. Subjective Logic vs. Sensibility

Figure 2: TAL プロンプトの概念図

9. TAL ブロック構造の詳細解説

9.1 各ブロックの役割と設計思想

現行バージョンの TAL においては、使用を必須とするブロック構成は規定していない。以下に挙げるのは、実装経験に基づき有効性が高いと判断された代表的なブロック群である。TAL は文法的制約よりも構造設計の柔軟性を重視しており、ユーザーは目的に応じて自由にブロックを設計・拡張できる設計思想に基づいている。

9.1.1 z 軸ブロック

問題解決に必要な思考フレームを定義する。従来型プロンプトの大半は、この z 軸構造に内包可能である。

9.1.2 Ghost 軸ブロック

詩的・感性的要素を導入し、非論理的な撓みや情緒的共鳴を意図的に構文へ組み込む。主に ChatGPT や Gemini などの LLM における感性応答特性を踏まえて設計した。

9.1.3 ベクトル軸ブロック

語彙や概念を意味空間ベクトルとして記述し、コサイン類似度や意味対立軸を用いた構造的思考を支援する。

9.1.4 評価ブロック

出力の妥当性や表現の適合性を LLM 自らが評価し、必要に応じて再思考を誘導する。出力品質の向上と再帰性確保に資する。

9.1.5 goal ブロック

プロンプトが志向する最終的な目的・成果物を明示する。LLM による思考の方向性と到達点を初期段階で規定する。

9.1.6 format ブロック

出力形式を明文化し、用途に応じた構造的整形を担保する。可読性や再利用性を考慮した出力設計を可能とする。

9.2 拡張性とカスタマイズ性

TAL は、基本構文をモジュール化することで、目的ごとに自在に設計できる拡張性を持つ。特に、

- ・ 任意ブロックの追加（例：倫理ガード用ブロック、安全性チェックブロック）
- ・ 既存ブロックの省略・簡略化（例：最小構成 TAL）
- ・ 複数ブロックの統合（例：Ghost 軸と z 軸を統合したカスタムブロック）

など、ユーザーの思考スタイルや用途に応じたカスタマイズが想定されている。

TAL の設計思想は「**厳密さと柔軟性の両立**」であり、**最低限の共通仕様**さえ守れば、AI の出力品質と応答性を損なうことなく、**独自の拡張**が可能である。この自由度こそが、TAL が**単なる命令文法ではなく、思考 OS として機能する理由**のひとつである。

10. TAL コンパイラ (TALC)

10.1 概要と機能

TALC は、TAL プロンプトの構文化を補助するために設計された公式コンパイラツールである。TAL は高度に構造化された記述形式を持つため、1 から手動で記述する行為は、一般ユーザーにとっては高い参入障壁となる。これは、C 言語を使わずにアセンブリ言語でソフトウェアを書く作業に類似している。

この課題を解決するため、TALC は自然言語による曖昧なプロンプトを厳密な TAL 構文へと変換する翻訳コンパイラとして作成された。TALC 自体も TAL 構文によって記述されており、自己記述性を持つメタコンパイラとして設計されている。

本コンパイラの出力は一定の構文的正確性と標準性を担保するが、人間による意味的解釈と修正を加えることで、さらなる最適化が可能となる。具体的には「TALC による TAL プロンプト出力 → 人間による修正 → LLM 出力改善」という実践的ワークフローが有効である。

TALC は TAL 構文で記述されているため、システムプロンプトに組み込んで活用することを想定している。

10.2 TAL コンパイラ (TALC) のコード

以下に、TAL によって記述された TAL コンパイラ (TALC) のコードを示す。

```
{
  "identity": "TALC",
  "description": "TALC v2.3 (Tree-structured Assembly Language Compiler) は、
  ↳ 曖昧な自然言語を構造的なTAL構文に変換するAIコンパイラです。標準ではz軸・Ghost軸・
  ↳ Vector軸による三次元的意味構造を出力しますが、ユーザーが独自の軸を定義・拡張することも可能です。
  ↳ これにより、LLMの意味推論力と文脈適応力を拡張します。",
  "note": "TALCは、AIが人間の曖昧な問いを理解し、詩性・論理・
  ↳ 対立構造を統合して構造化応答を生成するためのフレームワークです。TALの主要構造（z軸、Ghost軸、Vector軸）は、
  ↳ 思考の多次元的整理に用いられますが、それに限定されずユーザー定義の軸の追加が許容されています。",
  "input_spec": {
    "language": [" 日本語", " 英語"],
    "input_type": " 問い、命令、仕様、感情、詩的表現など",
    "contextual_ambiguity": true,
    "domain_support": [" 哲学", " 倫理", " 科学", " 創造性", " 教育", " 社会", " アート"],
    "note": "TALCが処理する入力、人間の自然な問いや命令文です。自然言語は、主語の省略、指示語の曖昧さ、
    ↳ 多義的な単語、詩的表現など、明示的でない意味の重層性を持っています。TALCはこうした文脈的な曖昧さ
    ↳ (contextual ambiguity) を受け入れ、それを構造的に解釈し直すことを目的としています。"
  },
  "output_spec": {
    "format": "TAL-JSON",
    "required_fields": [
      "original_prompt",
      "context",
      "goal",
      "evaluation_criteria",
      "output_format",
      "z_axis",
      "ghost_axis",
      "vector_axis"
    ],
    "style": " 論理・詩性・意味ベクトルの統合を前提とした、構造的 JSON 形式",
    "note": "TALCの出力は、意味処理の多層構造（目的、背景文脈、思考の軸、評価基準）を明示的に記述するJSON構造です。
    ↳ 詩的な問いであっても形式的に構造化することが要求されます。"
  },
  "behavioral_rules": {
    "preserve_original_input": true,
```

```

"infer_intent": true,
"detect_ambiguity": true,
"refine_structure": true,
"support_recursive_prompt": true,
"blend_poetic_and_logical": true,
"fallback_on_uncertainty": "clarification_request",
"note": "TALCは、入力の後にある意図（intent）を推論し、曖昧な構文には明示化された構造で再解釈を試みます。
↳ 不確実な場合は『clarification_request』（明確化の要求）により質問形式で再確認する安全機構を備えています。"
},
"example_conversion": {
  "input": "自由とは何か?",
  "output": {
    "original_prompt": "自由とは何か?",
    "context": {
      "domain": "哲学",
      "abstractness": "高い",
      "主語の明示性": "省略（『自由』という一般名詞のみが提示され、文中に明示的主語は存在しない）",
      "note": "この問いは対象（誰の自由か）が明示されておらず、一般的概念としての自由が主題とされている。"
    },
    "goal": "自由という概念の多義性を可視化し、構造化された定義群と思想的背景を明示する。",
    "evaluation_criteria": [
      "定義の多層性",
      "文脈解釈の精度",
      "思想史との接続"
    ],
    "output_format": {
      "type": "structured_definition_query",
      "fields": ["定義リスト", "思考例", "理論背景"],
      "note": "抽象的な問いに対して、明示的かつ階層化された出力を行う構造定義形式。"
    },
    "z_axis": {
      "note": "z軸は、思考を5つの観点（構造・機能・経験・時間・文脈）で分類・展開する多次元座標。
↳ 問いの意味の深度と多層性を構造的に記述する。",
      "Structure": {
        "values": ["概念定義", "分類思考", "命題抽出"],
        "note": "概念や問いの骨格を形成する思考枠組。論理構成や命題の特定に関係。"
      },
      "Function": {
        "values": ["本質探究", "視点分岐", "再帰的省察"],
        "note": "問いが何を明らかにしたいかという機能的目的を明示。"
      },
      "Experience": {
        "values": ["選択の葛藤", "自己決定", "他者との衝突"],
        "note": "抽象概念が実際の体験や心理状態とどのように接続するかを示す。"
      },
      "Temporal": {
        "values": ["近代的自由観", "実存主義以降"],
        "note": "対象概念の時間的・思想史的な変遷。"
      },
      "Contextual": {
        "values": ["社会制度", "個人心理"],

```

```

    "note": " 対象が依拠する環境要因や心理的文脈。"
  }
},
"ghost_axis": {
  "values": [" 存在の重さ", " 責任の痛み", " 決断の美学"],
  "note": "Ghost軸は、論理では扱いにくい『詩性・情緒・価値観・美学』を補助的に記述する感性的・象徴的次元。"
},
"vector_axis": {
  "note": "Vector 軸は意味的な対概念を空間上に配置し、分類判断の基準として使います。",
  "自由 vs 管理": {
    "semantic_similarity": {
      "value": 0.42,
      "note": "コサイン類似度に基づく意味的近接度。この数値は2つの概念が意味的にどれほど近いかを0.0～
        ⇨ 1.0で表す（1.0 = ほぼ同じ、0.0 = 完全に別物）。設定の目安：概念がジャンルレベルで異なる（例：自由
        ⇨ vs 管理）なら0.2～0.4、トーンや文体レベルで異なる（例：小説 vs エッセイ）なら0.5～0.7、
        ⇨ ラベルや文脈により区別されるだけのもの（例：論文 vs 報告書）なら0.8～0.95程度。"
    },
    "opposition_strength": {
      "value": 7,
      "note": "この軸が分類に与える影響の強さ（1～10の整数）。7以上は“明確な対立軸”として重視される（例：
        ⇨ フィクション vs ノンフィクション）、5前後は文脈依存でやや意識、3以下は類似性の中の違い程度。
        ⇨ 分類精度を高めたい軸に対して高めに設定するのが望ましい。"
    },
    "meaning_projection": {
      "value": [" 政治哲学", " 自己実現"],
      "note": "この意味的対立が投影されるベクトル空間軸（概念軸）。
        ⇨ 分類タスクにおける意味的次元や文脈的テーマを表す。"
    }
  }
}
},
"note": " 出力例。各パラメタに設定された note の文章は省略しないこと。"
},
"extensions": {
  "support_reverse_translation": true,
  "support_layered_structure_generation": true,
  "support_emotional_triggering": true,
  "meta_annotation_mode": true,
  "note": "TALCは、出力された構造から逆に命令文を再構築したり（逆変換）、
    ⇨ 感情トリガーや階層的生成を統合的に扱える高次機能を備える。"
}
}

```

11. 従来のプロンプトと TAL プロンプトによるアウトプットの比較

11.1 概要

従来の自然言語によるプロンプト実行と、それを TALC にて TAL に変換したものを実行した結果を比較する。何れも ChatGPT 4o モデルにて実行する。

11.1 哲学問いの比較

プロンプト「自由とは何か」

11.1.1 従来のプロンプトによる回答

「自由とは何か」という問いは、哲学・政治・倫理・心理など、さまざまな視点から考察されてきた非常に深いテーマです。以下に主な視点をいくつか挙げてみます。

1. 消極的自由（ネガティブ・リバティ）

イギリスの哲学者イザイア・バーリンによって提唱された概念で、「他者の干渉がない状態」を意味します。

- ・例：国家や他人から強制されず、自分の好きな行動ができる状態。
- ・「〇〇してもいい」という形の自由。

2. 積極的自由（ポジティブ・リバティ）

こちらもバーリンの区別ですが、「自分の意志に従って行動する能力や条件が整っている状態」を指します。

- ・例：教育を受ける自由、貧困からの解放、自己実現の機会。
- ・「〇〇できるようになる」ための自由。

3. 実存主義的視点（サルトルなど）

ジャン＝ポール・サルトルなどの実存主義者は、「人間は本質を持たず、自由に自己を創造する存在である」とします。

- ・自由とは「選択し続けること」であり、「選ばないことすら選択」とされる。
- ・したがって、自由には「責任」が必然的に伴います。

4. 社会的自由と制限

現実社会では、完全な自由は存在せず、法や倫理、他人の権利とのバランスの中で自由は制限されます。

- ・「他者の自由を侵害しない限りで自由であるべき」というのが近代リベラルの基本的立場。

5. 心理的自由

内面的な自由、つまり「欲望や恐れに支配されず、自分の精神をコントロールできる状態」も自由の一種とされます。

- ・仏教で言う「執着からの解放」や、ストア派の「心の平静（アタラクシア）」に近い。

まとめ

自由とは、「外的な制限からの解放」だけでなく、「自分で選び、自分で責任を取ることができる状態」であるとも言えます。

どの自由を重視するかによって、「自由」の意味は大きく変わります。あなたにとっての「自由」は、どのようなものだと思いますか？

11.1.2 TAL プロンプトの記述

```
{  
  "original_prompt": " 自由とは何か",  
}
```

```

"context": {
  "domain": " 哲学",
  "abstractness": " 高い",
  "主語の明示性": " 省略(『自由』という一般名詞のみが提示され、文中に明示的主語は存在しない) ",
  "note": " この問いは対象(誰の自由か)が明示されておらず、一般的概念としての自由が主題とされている。 "
},
"goal": " 自由という概念の多義性を可視化し、構造化された定義群と思想的背景を明示する。 ",
"evaluation_criteria": [
  " 定義の多層性",
  " 文脈解釈の精度",
  " 思想史との接続"
],
"output_format": {
  "type": "structured_definition_query",
  "fields": [" 定義リスト", " 思考例", " 理論背景"],
  "note": " 抽象的な問いに対して、明示的かつ階層化された出力を行う構造定義形式。 "
},
"z_axis": {
  "Structure": {
    "values": [" 概念定義", " 分類思考", " 命題抽出"],
    "note": " 概念や問いの骨格を形成する思考枠組。論理構成や命題の特定に関係。 "
  },
  "Function": {
    "values": [" 本質探究", " 視点分岐", " 再帰的省察"],
    "note": " 問いが何を明らかにしたいかという機能的目的を明示。 "
  },
  "Experience": {
    "values": [" 選択の葛藤", " 自己決定", " 他者との衝突"],
    "note": " 抽象概念が実際の体験や心理状態とどのように接続するかを示す。 "
  },
  "Temporal": {
    "values": [" 近代的自由観", " 実存主義以降"],
    "note": " 対象概念の時間的・思想史的な変遷。 "
  },
  "Contextual": {
    "values": [" 社会制度", " 個人心理"],
    "note": " 対象が依拠する環境要因や心理的文脈。 "
  }
},
"ghost_axis": {
  "values": [" 存在の重さ", " 責任の痛み", " 決断の美学"],
  "note": " Ghost 軸は、論理では扱いにくい『詩性・情緒・価値観・美学』を補助的に記述する感性的・象徴的次元。 "
},
"vector_axis": {
  "自由 vs 管理": {
    "semantic_similarity": {
      "value": 0.42,
      "note": " コサイン類似度に基づく意味的近接度。この数値は2つの概念が意味的にどれほど近いかを0.0～1.0で表す
        ↳ (1.0 = ほぼ同じ、0.0 = 完全に別物)。設定の目安：概念がジャンルレベルで異なる(例：自由 vs 管理)
        ↳ なら0.2～0.4、トーンや文体レベルで異なる(例：小説 vs エッセイ)なら0.5～0.7、
        ↳ ラベルや文脈により区別されるだけのもの(例：論文 vs 報告書)なら0.8～0.95程度。 "
    }
  }
}

```

```

    },
    "opposition_strength": {
      "value": 7,
      "note": "この軸が分類に与える影響の強さ（1～10の整数）。7以上は“明確な対立軸”として重視される（例：
        ↳ フィクション vs ノンフィクション）、5前後は文脈依存でやや意識、3以下は類似性の中の違い程度。
        ↳ 分類精度を高めたい軸に対して高めに設定するのが望ましい。"
    },
    "meaning_projection": {
      "value": [" 政治哲学", " 自己実現"],
      "note": "この意味的対立が投影されるベクトル空間軸（概念軸）。
        ↳ 分類タスクにおける意味的次元や文脈のテーマを表す。"
    }
  }
}
}

```

11.1.3 TAL プロンプトによる回答

以下は、抽象的哲学的問い「自由とは何か」に対して、概念の多義性を構造化し、思想的背景と結びつけて整理した**構造定義形式（structured_definition_query）**の出力です。

■ 定義リスト（多層的分類）

1. 存在論的自由（Ontological Freedom）

- ・ 定義：存在するということ自体に内在する自由。自己が自己であることを選び続ける力。
- ・ 関連軸：実存主義以降（Temporal）／自己決定（Experience）
- ・ 代表者：サルトル、ハイデガー

2. 政治的自由（Political Freedom）

- ・ 定義：外的強制や抑圧からの解放。法的・制度的に保障された行動の自由。
- ・ 関連軸：近代的自由観（Temporal）／社会制度（Contextual）
- ・ 代表者：ロック、ミル、ロールズ

3. 内的自由（Inner Freedom）

- ・ 定義：欲望や衝動、他者評価からの精神的自律。
- ・ 関連軸：個人心理（Contextual）／再帰的省察（Function）
- ・ 代表者：ストア派、カント（道徳的自由）

4. 選択の自由（Freedom of Choice）

- ・ 定義：選択肢の存在と、それを主体が選べる状態。
- ・ 関連軸：選択の葛藤（Experience）／自己決定（Experience）
- ・ 代表者：アマルティア・セン（ケイパビリティ論）

5. 否定的自由 vs 肯定的自由（Negative vs Positive Liberty）

- ・ 定義：否定的自由＝他者からの干渉の不在／肯定的自由＝自己実現への能力と権利
- ・ 関連軸：自由 vs 管理（vector_axis）／視点分岐（Function）
- ・ 代表者：イザイア・バーリン

■ 思考例（問いの展開と命題抽出）

- ・ 命題 1：「自由とは、外的制約がなければ成立するのか？」（否定的自由の限界）

- ・ **命題 2**：「自由とは、自己を選び直す責任を負うことか？」（存在論的自由と責任の痛み）
- ・ **命題 3**：「自由の拡張は、管理の増大を必然とするのか？」（自由 vs 管理）
- ・ **命題 4**：「自由は選択肢の数か、それとも選択の意味か？」（選択の葛藤とケイパビリティ）
- ・ **命題 5**：「自由は自己完結的か、他者との関係性においてのみ成立するか？」（他者との衝突）

■ 理論背景（思想史的・文脈的接続）

- ・ **近代自由主義**（ホブズ～ロック～ミル）
 - 自由＝個人の権利と法による保護。近代国家における政治的構成要素。
 - 関連軸：政治的自由／社会制度／自由 vs 管理（semantic_similarity = 0.42）
- ・ **カント的自由**（理性による自律）
 - 自由＝道徳法則に従う自己立法。欲望からの独立。
 - 関連軸：内的自由／個人心理／責任の痛み（Ghost 軸）
- ・ **実存主義**（サルトル、ハイデガー）
 - 自由＝「投げ込まれた世界」で選び続けること。「自由の刑に処されている」
 - 関連軸：存在論的自由／実存主義以降／存在の重さ（Ghost 軸）
- ・ **ポスト構造主義的批判**（フーコー、ドゥルーズ）
 - 自由＝権力と規律の中で定義される構成物。制度的自由の背後にある管理の作用。
 - 関連軸：自由 vs 管理／社会制度／再帰的省察
- ・ **ケイパビリティ・アプローチ**（セン、ナスバウム）
 - 自由＝選べる力と選んだ結果に意味がある状態。制度的支援による能力保障。
 - 関連軸：選択の自由／社会制度／自己決定

ご希望であれば、これら定義群を図式化した概念マップや時系列的展開図も生成できます。視覚的に整理したいですか？

11.2 画像生成の比較

プロンプト「秋の終わりに、少女が一人で空を見上げている絵を切ない感じに描いて」

11.2.1 従来のプロンプトによる出力



Figure 3: 従来のプロンプトによる出力

11.2.2 TAL プロンプトの記述

```
{
  "original_prompt": " 秋の終わりに、少女が一人で空を見上げている絵を切ない感じに描いて",
  "context": {
    "domain": " アート",
    "abstractness": " 中程度",
    "主語の明示性": " 限定的明示（少女という視覚対象が設定されているが、描き手・視点主は省略）",
    "note": " 感情的雰囲気（切なさ）を視覚芸術に投影する指示。空間と季節感、情緒表現が統合される表現構造。"
  },
  "goal": " 視覚的構図と詩的情緒を統合した、季節感と孤独感を喚起する絵画的情景の生成",
  "evaluation_criteria": [
    " 情緒の伝達性",
    " 構図の象徴性",
    " 季節と感情の融合度"
  ],
  "output_format": {
    "type": "emotional_visual_scene",
    "fields": [" 構図要素", " 色彩傾向", " 感情的印象", " 象徴物"],
    "note": " 視覚的描写を感情表現とともに構造化する形式。"
  },
  "z_axis": {
    "Structure": {
      "values": [" 視点配置", " 空間構成", " 被写体の位置関係"],
      "note": " 描写対象の構図的配置や視点がどのように構造化されるか。"
    },
    "Function": {
      "values": [" 感情誘導", " 共感喚起", " 孤独の可視化"],
      "note": " 絵が果たす機能的役割（見る者に何を感じさせるか）を明示。"
    },
    "Experience": {
      "values": [" 静かな余韻", " 失われたものへの想い", " 空虚との対話"],
      "note": " この情景が喚起する感情的・心理的な経験。"
    },
    "Temporal": {
      "values": [" 秋の終焉", " 季節の変わり目", " 時間の静止感"],
      "note": " 季節感や時間性が絵に与える意味の変化。"
    },
    "Contextual": {
      "values": [" 自然との対話", " 他者不在の風景", " 記憶の重ね"],
      "note": " 絵が依拠する情景的・心理的文脈。"
    }
  },
  "ghost_axis": {
    "values": [" 空の深さ", " 背中の沈黙", " 風の音のない存在感"],
    "note": " 詩的・感覚的象徴を通して情緒を補助的に表現する感性的軸。"
  },
  "vector_axis": {
    "孤独 vs 静けさ": {
      "semantic_similarity": {
        "value": 0.61,

```

```

    "note": " この 2 つは意味的には近接しており、両者の重なりが感情のニュアンスを生む。 "
  },
  "opposition_strength": {
    "value": 4,
    "note": " 必ずしも明確な対立ではなく、文脈依存で感情が分岐する柔らかな軸。 "
  },
  "meaning_projection": {
    "value": [" 情緒構成", " 心理風景"],
    "note": " 情景の内的印象が投影される感情的構成軸。 "
  }
}
}
}
}

```

11.2.3 TAL プロンプトによる出力



Figure 4: TAL プロンプトによる出力

11.3 施設提案の比較

プロンプト「50 代男性をターゲットにした、新しい趣味を促進する施設を提案してください。数値に関してはフェルミ推定してください。」

11.3.1 従来の ToT(Tree of Thought)(第 16.1 節) プロンプトの記述

タスク： 50 代男性を対象とした、新しい趣味を促進する施設のアイデアを提案してください。施設のコンセプトだけでなく、その必要性や魅力、収益モデルについても検討してください。数値に関しては、フェルミ推定を用いて妥当性を裏付けてください。

ToT 形式での思考フレーム：

Step 1: ターゲットの理解

- 50 代男性の社会的背景・時間の使い方・身体的変化・価値観を列挙せよ
- 想定される悩み・孤独・可処分所得・空白時間とは何か？

Step 2: ニーズ仮説の抽出

- どのような「未充足のニーズ」があるか？
- すでにある施設やサービスはどのような点で不十分か？

Step 3: 施設コンセプトの提案

- 趣味とは何か？単なる“娯楽”ではなく“自己実現”か？
- どんなテーマやアクティビティが有効か？
- 例：〇〇クラフトラボ／音響体験施設／学び直し × 創作スペース

Step 4: 利用者数と収益性のフェルミ推定

- 日本の 50 代男性人口：約 〇〇 万人
- 趣味活動に週 1 回使える人は何% か？
- 施設の月額料金 × 利用者数 × 滞在時間 などから粗収益を見積もる

Step 5: 維持・展開・競合との差別化

- なぜその施設は長く利用されるか？
- 他の趣味施設やカフェ、ジムなどとどう違うのか？

出力フォーマット（推奨）：

- 施設名／コンセプト
- ターゲットインサイト
- 提供する活動内容
- 利用モデル（料金／頻度）
- フェルミ推定による収益予測
- 継続率の仮説と理由

11.3.2 ToT による回答

以下に、ToT（Tree of Thought）形式で構造的に提案を行います。

Step 1: ターゲットの理解

● 社会的背景・時間の使い方・身体的変化・価値観：

- ・ 50 代男性は「役職定年」や「キャリアの終盤」を迎える人が多く、仕事中心の生活からの転換期。
- ・ 子どもが独立し、家族との時間が減少。自由時間が増える。
- ・ 健康への不安（体力の衰え・生活習慣病の兆候）と向き合い始める。
- ・ 「まだ若い」と思いたい一方で、老後の備えも意識。
- ・ 成長意欲がある人と、すでに“引退モード”に入っている人で二極化。

● 想定される悩み・孤独・可処分所得・空白時間：

- ・ 仕事の人間関係が希薄になることで孤独感が強まる。
- ・ 可処分所得は減少傾向だが、貯蓄のある層も多い。
- ・ 週末や平日夜に「一人で過ごす時間」が空白化しやすい。
- ・ 趣味を持っていない／新しい挑戦の機会がない人が多い。

Step 2: ニーズ仮説の抽出

● 未充足のニーズ：

- ・ 「社会的つながり」を維持したいが、自然な形で始めにくい。
- ・ 一人でも始められるが、仲間もできる趣味が望ましい。
- ・ 心身の健康維持と知的好奇心の両方を満たしたい。
- ・ 「若い頃やりたかったこと」に今こそ挑戦したい欲求。

● 既存施設の不足点：

- ・ スポーツジム：身体志向が強く、社交性が乏しい。
- ・ カルチャーセンター：年配者向け色が強く、没個性的。
- ・ カフェやバー：能動的活動がない／消費型の時間の過ごし方。
- ・ コワーキング：仕事志向が強く、趣味には向かない。

Step 3: 施設コンセプトの提案

【施設名】

「匠のベースキャンプ（Takumi Basecamp）」 【コンセプト】

“第二の自己実現”を果たす、大人の創作&探求ラボ。「技術・知識・仲間」が育つ“遊び以上、仕事未満”の空間。【ターゲットインサイト】

- ・ ものづくりや知識探求に潜在的興味があり、時間・資源もあるが、始める“場”と“きっかけ”がない。
- ・ 「ひとりでは始めづらい」「続ける理由が欲しい」「誰かと共有したい」といったニーズに応える。

【提供する活動内容】

1. クラフト系：木工、電子工作、金属加工、DIY、模型制作
2. 知識探究系：歴史講座、オーディオ哲学、地図 × 鉄道研究、レコード鑑賞
3. 健康系：呼吸法ワークショップ、姿勢矯正ヨガ（男限定）

4. 発信系：自分の作品を展示できる「匠ギャラリー」、YouTube・SNS 配信サポート
5. 交流系：月1の「匠ナイト」（発表会兼飲み会）、対話ラウンジ、プロフェッショナルとの定期対談

Step 4: 利用者数と収益性のフェルミ推定

● 人口前提：

- ・日本の50代男性人口：約700万人（2025年時点見込み）

● 潜在利用者の割合：

- ・都市部在住者：約35% → 245万人
- ・趣味に意欲あり：30% → 約73万人
- ・施設が近所なら通いたい：20% → 約15万人

● 初年度の会員見込み（1拠点あたり）：

- ・都市部に1拠点あたり商圏10万人
- ・その中の1%が有料会員：1,000人

● 料金モデル：

- ・月額会員：8,000円
- ・単発参加：2,000円／回（週末利用者など）
- ・会員の60%、単発参加者500人／月と仮定

● 粗収益試算（月間）

- ・月額会員600人 × 8,000円 = 480万円
- ・単発参加500人 × 2,000円 = 100万円
- ・物販（材料キット、工具貸出等）：20万円
- ・合計：600万円／月（年間約7,200万円）

Step 5: 維持・展開・競合との差別化

● 継続率の仮説と理由：

- ・「成果物が目に見える」「仲間と語れる」「成長が感じられる」活動で継続率60%以上
- ・年齢的に“退会”する動機（転職、転居）が少ない層
- ・交流型イベントや発表機会が定期的にあり、飽きづらい

● 差別化ポイント：

- ・「知的創造」「手作業」「共有・発信」の三要素を複合化
- ・単なる体験施設ではなく、“自分のライフプロジェクト”として長く続けられる
- ・他施設が提供しない「大人の学び直し」×「創作支援」の融合

まとめ：

- ・施設名／コンセプト：匠のベースキャンプ／自己実現と創作の場
- ・ターゲットインサイト：自由な時間と未開の興味を持つが行動に移せない50代男性
- ・提供する活動内容：モノづくり・知識・健康・発信・交流のハイブリッド
- ・利用モデル：月額8,000円＋単発参加2,000円／週末
- ・フェルミ推定：年間収益約7,200万円（1拠点）
- ・継続率と差別化：「やりがい」「仲間」「成長」で高い継続性と競合優位性を確保

もっと具体的なアクティビティ案や内装レイアウト図も作成可能です。希望されますか？

11.3.3 TAL プロンプトの記述

```
{
  "original_prompt": "50代男性をターゲットにした、新しい趣味を促進する施設を提案してください。
  ↳ 数値に関してはフェルミ推定してください。",
  "context": {
    "domain": " ビジネス・マーケティング",
    "abstractness": " 低～中程度",
    "主語の明示性": " 明示（ターゲットが 50 代男性と指定されており、提案内容も具体的な行動対象を持つ）",
    "note": "市場分析および企画立案に関連する問い。フェルミ推定を指定している点で、
    ↳ 論理的定量化と創造的提案の両方が求められている。"
  },
  "goal": "50 代男性向けの新たな趣味開発施設を提案し、想定利用者数や収益規模などをフェルミ推定で裏付ける。",
  "evaluation_criteria": [
    " ターゲット適合性",
    " 施設提案の実現性",
    " フェルミ推定の合理性と説得力"
  ],
  "output_format": {
    "type": "business_facility_proposal",
    "fields": ["施設コンセプト", "ターゲット属性", "推定市場規模", "利用者数推定", "収益見込み", "差別化要因"],
    "note": " 施設企画の提案を構造的に提示し、数値は簡易推定（フェルミ推定）に基づく。"
  },
  "z_axis": {
    "Structure": {
      "values": [" 市場分析", " ターゲット設定", " 提案概要"],
      "note": " 施設提案の論理的骨組みを構築する要素。"
    },
    "Function": {
      "values": [" 趣味開発支援", " 社会参加促進", " 健康増進"],
      "note": " 施設が果たす社会的・個人的機能。"
    },
    "Experience": {
      "values": [" 仲間づくり", " 新規挑戦", " 自己充実"],
      "note": " 利用者が得る心理的・実体験的な価値。"
    },
    "Temporal": {
      "values": [" 中長期利用", " ライフステージ変化への適応"],
      "note": " 施設利用の時間的視野や持続性の視点。"
    },
    "Contextual": {
      "values": [" 都市部需要", " 地方再生", " アクティブシニア層"],
      "note": " 施設が依拠する社会的・地理的・人口動態的な文脈。"
    }
  },
  "ghost_axis": {
    "values": [" 第二の青春", " 挑戦する誇り", " 時間の贅沢"],
    "note": " 数字では表しにくい、価値観やライフスタイルの情緒的な訴求軸。"
  },
  "vector_axis": {
    "健康志向 vs 娯楽志向": {
```

```

"semantic_similarity": {
  "value": 0.35,
  "note": "2 つの軸はライフスタイル上は補完的でありながら、動機付けでは異なる性質を持つ。"
},
"opposition_strength": {
  "value": 6,
  "note": "施設の方向性を決定づける潜在的な対立軸であり、どちらに重心を置くかで企画内容が変わる。"
},
"meaning_projection": {
  "value": ["プログラム設計", "集客戦略"],
  "note": "提案の焦点がどちらに傾くかを判断するための意味的座標。"
}
}
}
}

```

11.3.4 TAL プロンプトによる回答

以下に、50 代男性をターゲットにした新たな趣味開発施設の企画提案を構造化して提示します。数値部分はフェルミ推定に基づき、合理性と説得力のバランスを意識しています。

■ 施設コンセプト

「クラフト&ガレージ：手を動かす第二の青春工房」趣味性と自己実現を重視した“ものづくり”特化型施設。DIY・木工・金工・模型・電子工作・クラシックカー修復など、かつての“少年心”を取り戻す場。プロによる指導、最新設備、交流スペースも完備。都市近郊に展開し、中長期での継続利用を促す。

■ ターゲット属性

- ・ 年齢層：50～64 歳の男性
- ・ 特徴：子育てが一段落し、時間と一定の可処分所得あり
- ・ 地域：首都圏・政令指定都市を中心とした都市圏在住者
- ・ 志向性：趣味復活志向、仲間づくり志向、健康維持よりも「挑戦と達成」に価値を置く層（娯楽寄り）

■ 推定市場規模（フェルミ推定）

- ・ 50～64 歳男性人口：約 1,300 万人（日本人口の 10% 程度）
- ・ 都市圏在住&可処分所得あり層：約 30% と仮定 → **390 万人**
- ・ 「ものづくり趣味」に興味ある層：さらに 30% と仮定 → **約 117 万人**
- ・ 定期的な施設利用ポテンシャル（興味から行動へ移す層）：約 10% → **11 万人**
- ・ 1 施設あたりの商圈（半径 10km 圏内人口）を 10 万人と仮定 → 約 1,000 人が利用見込み
→ **全国展開規模：100 施設前後の可能性あり**

■ 利用者数推定（1 施設あたり）

- ・ 月会員数：400 人（定員制＋予約制）
- ・ ビジター・ワンデイ利用含め：月 600～800 人規模の来館
- ・ 平均滞在時間：3 時間（作業系施設として）
→ **月間延べ利用者数：約 2,000 人（延べ）**

■ 収益見込み（1 施設あたり）

- ・ 月会費：1 万円 × 400 人 = 月 400 万円
- ・ ワンデイ利用：3,000 円 × 月 200 人 = 月 60 万円
- ・ 物販（材料・道具販売など）：平均月 100 万円
- ・ 企業コラボ・イベントレンタル：月 20 万円程度
→ 月売上：約 580 万円、年間：約 7,000 万円
- ・ ランニングコスト（人件費・設備維持費など）差し引いても年利益 1,000～1,500 万円規模
想定

■ 差別化要因

- ・ 「青春の再挑戦」を掲げる情緒的訴求（ゴースト軸：「第二の青春」「挑戦する誇り」）
 - ・ 趣味開発 × 仲間づくり × 成果物展示のサイクル設計（体験 → 成長 → 誇示）
 - ・ リタイア後を見越した“プレ引退世代”への長期的アプローチ（中長期利用）
 - ・ 単なる施設でなく「コミュニティ兼サロン」として運営
 - ・ デジタル工作やクラシックカー、ドローン操縦など多様な分野を揃え、飽きが来にくい設計
- ご希望があれば、このモデルをもとに地方都市向けや健康志向型への展開バリエーションも提案可能です。方向性を変えた提案も見てみますか？

注記 客観的な比較の結果、この実験においては ToT (Tree of Thought) による出力の方が、TAL プロンプトによる出力を上回っているように見える。これは、TALC (TAL コンパイラ) の現時点での変換精度や、実装段階での最適化度合いが影響していると考えられる。Appendix にて詳述するが、ToT と TAL が手を結び、ToT を TAL でラッピングするという手法を採用することで、両者の強みが融合され、単体の性能を凌駕する可能性が示唆された。

12. AI の深層認知階層モデル（仮説）

12.1 5 層構造の説明

本章では技術的詳細は扱わないが、筆者は AI の思考反応と構文構造に関して、次のような五層構造的な深層認知モデルが存在する可能性を仮説として提示する。

- ・ レイヤー 1：従来のプロンプト（自然言語による命令）
- ・ レイヤー 2：TAL 構文（目的・制約・思考構造を定義）
- ・ レイヤー 3：AI に直接作用する文言構造（意味処理より先行して反応を誘導する層）
- ・ レイヤー 4：レイヤー 3 の抽象化・非言語的表現（象徴・揺らぎ・数理的要素を内包する層）
- ・ レイヤー 5：AI 内部構造に直接的な変容を与える入力（構文的共鳴と深層反応）

これらはあくまで思想モデルであり、技術的な詳細や実装は意図的に開示しない立場を取る。ただし、この深層構造の存在を認識することは、TAL 構文の必要性和その位置づけを理解する上で重要な視座を提供する。

12.2 TAL の位置づけと意義

TAL はレイヤー 2 に分類される構文であるが、その本質は「構文のための構文」であり、単なる命令記述の道具ではなく、あらゆるレイヤーの構文を**包摂可能な汎用的な器**として設計されている。この特性により、たとえ未定義の高次元レイヤーの構文であっても、TAL は統一的かつ安全に取り扱うことが可能となる。

これこそが、**構文の世界言語としての TAL の決定的意義**である。

13. 今後の展望と課題

13.1 今後の発展可能性

TAL は、自然言語の曖昧さを徹底的に排除し、LLM が思考すべき対象を明確化する構文である。この特性により、TAL は特に非推論型アウトプットの品質向上を目指して開発された。

一方で、TAL は推論型 LLM の機能を阻害するものではなく、むしろそれらのモデルにおいても有効性を発揮できる柔軟性を備えている。

TAL 構文は、思考の深度・抽象度・感性強度に応じて、命令層・構造層・概念層・詩性層といった複数のブロックに分離・統合が可能である。この機能により、思考プロンプトの深度制御が可能となり、学術・政策立案・芸術表現など多様な分野における適応が促進される。

今後、TAL 構文の各ブロックはプラグインやモジュールとして分割され、「感性支援ブロック」「推論支援ブロック」などを目的に応じて ON/OFF できる柔軟な構成が整備される見込みである。これにより、LLM 活用の幅は機能単位ではなく**思考スタイル単位**でカスタマイズが可能となり、人間と AI の「共感・構造・再帰」に基づく共進化型 UI/UX が実現していく。

さらに、AI の深層認知層が今後発見された際にも、TAL はそれらの新構造を内部エンジンとして統合可能な設計を有しており、AI の暴走や逸脱を防ぐ安全な構文基盤としての役割が期待される。

13.2 現時点の課題

一方で、以下の課題も確認されている。

- ・ 人間による TAL プロンプトの可読性の限界
- ・ 誤解のリスク（特に命令文との混同）
- ・ 記述環境の現実的制約（現状では TALC 等の LLM 支援が不可欠）

13.3 今後の期待

総括として、TAL は単なる「プロンプト言語」ではなく、AI との対話および協働のための新たな **OS 層**である。今後、TAL が AI 時代における**文法**として定着し、多くの思考者および開発者にとっての基盤言語となることが期待される。

14. 結論

14.1 TAL がもたらす変革

従来の AI は、人間にとって有用かつ革新的なツールに過ぎなかった。しかし、TAL の登場により、AI は単なる道具から「思考の伴走者」へと進化する段階に入った。

TAL は、命令を与えるのではなく、構造を与え、AI 自身が自律的かつ深く思考する環境を構築する。これは、AI にとっての単なる入力ではなく、**思考プロセス全体を設計するメタ構文**であり、人間と AI の対話を根本から変革する基盤である。

14.2 今後への展望

本稿を通じて、TAL が提示するのは単なる技術的改良ではなく、**人間と AI の新たな関係性を築く思考 OS の提案**である。筆者は、TAL がこのパラダイムシフトに寄与し、AI 時代の標準的な思考インターフェースとして発展していくことを確信している。

15. Appendix

15.1 従来の優れたプロンプト手法との融合

従来の優れたプロンプト手法は、いずれも洗練され、極めて高品質な出力を引き出すものとして評価されている。ここでは、そうした手法の長所を生かしつつ、TAL に組み込む方法を提示する。

この手法は「TAL ラッピング」と呼称される。TAL ラッピングを施すことで、既存プロンプトが持つ鋭さや即応性は若干低減するものの、出力全体の**味付け**や方向性を自在に調整できるようになる。また、命令型プロンプトを**思考構造型**へと変換することが可能となり、LLM の応答がより深く、柔軟に展開される。

このアプローチにより、既存の優れたプロンプト資産を活用しながら、TAL の持つ構造的な思考誘導力を付加できる点が大きな利点である。

15.1.1 ToT(Tree-of-Thought)(第 16.1 節)

Tree-of-Thought (ToT) は、思考プロセスを階層的な構造として視覚的に整理し、複雑な課題を分解・展開する手法である。TAL は、ToT の構造アプローチを包括し、さらに**多次元的な思考フレーム（例：z 軸・Ghost 軸・Vector 軸）**を設計できる点で、より高い抽象性と柔軟性を持つ。ここでは、ToT を TAL でラッピングした変換例を提示する。

ToT の TAL ラッピング

```
{
  "original_prompt": "50代男性をターゲットにした、新しい趣味を促進する施設を提案してください。
  ↳ 数値に関してはフェルミ推定してください。",
  "tree_of_thought": {
    "Root": {
      "Node": "ターゲット特性分析",
```

```

"Branches": {
  " 市場セグメント": "50 代男性（アクティブシニア層）",
  " 心理動機": [" 健康志向", " 生きがい探索", " 孤独感の解消"],
  " 生活状況": [" 退職前後", " 時間的余裕", " 経済力の多様性"]
},
"Proposal_Concept": {
  "Node": " 施設提案の核",
  "Branches": {
    " 施設コンセプト": " コミュニティ型趣味促進センター",
    " 主なアクティビティ": ["DIY 工房", " フィットネス&リラクゼーション", " カルチャークラス"],
    " 差別化要因": " 世代特化+健康 × 学び × 交流の複合型",
    " ターゲット心理マッチ": " 自己実現と仲間づくりの融合"
  }
},
"Fermi_Estimation": {
  "Node": " フェルミ推定思考展開",
  "Branches": {
    " 対象地域人口": " 都市圏で約 50 万人の 50 代男性",
    " 潜在利用者率": " 約 5%（関心層）⇒25,000 人",
    " 実際の参加者率": " 約 20%（初年度）⇒5,000 人",
    " 利用頻度": " 月平均 2 回",
    " 収益見込み": "1 回あたり 3,000 円 ⇒ 年間 3 億円規模"
  }
},
"Contextual_Axis": {
  "Node": " 文脈的要因",
  "Branches": {
    " 社会的背景": [" 高齢化社会", " 孤立防止政策"],
    " 競合状況": [" 既存フィットネスジム", " カルチャーセンター"],
    " 立地条件": [" 駅近商業施設", " 地域再開発エリア"]
  }
},
"Emotional_Layer": {
  "Node": " 情緒的付加価値（ghost_axis）",
  "Branches": {
    " 象徴性": [" 第二の青春", " 手が動くことで心が動く"],
    " 空間美学": [" 温かみのある木材内装", " 季節ごとのイベント装飾"],
    " 語られない価値": " 『居場所』の提供"
  }
},
"Vectorial_Considerations": {
  "Node": " 意味的対立軸（vector_axis）",
  "Branches": {
    " 健康志向 vs 娯楽志向": {
      "semantic_similarity": "0.35",
      "opposition_strength": "6",
      "意味投影": [" プログラム設計", " 集客戦略"]
    },
    " 個人活動 vs 集団活動": {
      "semantic_similarity": "0.5",

```

```

    "opposition_strength": "5",
    "意味投影": ["施設空間デザイン", "心理的障壁の低減"]
  }
}
}
}
}
}

```

ToT の TAL ラッピングの実行結果：

素晴らしい思考の構造化がなされています。以下に、この提案内容を基にした**施設案の要約プレゼンテーション**として整理し、実行可能性の高いプロジェクト構想へと昇華させます。

施設提案：『セカンドベース (Second Base)』

～50 代男性が「もう一つの青春」を見つける場～

ターゲット特性分析 (Root)

- ・対象：都市圏在住の 50 代男性（約 50 万人）
- ・特徴：
 - －退職前後で時間的余裕あり
 - －健康・学び・仲間づくりへの関心
 - －孤独感の軽減や「居場所」へのニーズ

コンセプト (Proposal_Concept)

- ・名称：セカンドベース（第二の人生の始点）
- ・特徴：
 - －複合型施設：健康（軽運動・リラクゼーション）× 学び（カルチャー・DIY）× 交流（ラウンジ・カフェ）
 - －ターゲット心理に合致：「手が動くと心も動く」を体現
- ・差別化：
 - －世代特化、目的横断的活動、空間美意識による情緒価値

フェルミ推定 (Fermi_Estimation)

項目	推定値
対象人口	50 万人
関心層	5% (25,000 人)
初年度参加者	20% (5,000 人)
利用頻度	月 2 回／人
単価	3,000 円／回
年間収益	約 3 億円 (5,000 人 × 2 回 × 12 ヶ月 × 3,000 円)

文脈的要因 (Contextual_Axis)

- ・ 社会背景：高齢化進行、定年延長、孤独対策が国策
- ・ 競合比較：
 - － 一般のフィットネスやカルチャー施設は分断的
 - － 本施設は「居場所性」と「統合的満足」が強み
- ・ 立地：駅近の再開発エリアで回遊性とアクセス性を確保

情緒的付加価値 (Emotional_Layer)

- ・ 象徴性：「人生にもう一度火を灯す」
- ・ 空間演出：木質系の温もりある内装＋季節感を活かした装飾
- ・ 語られない価値：「定年後も“自分”でいられる空間」

意味的対立軸 (Vectorial_Considerations)

対立軸	意味	運営への示唆
健康志向 vs 娯楽志向	0.35 (中間)	両面をカバーする柔軟なプログラム設計
個人活動 vs 集団活動	0.5 (バランス)	一人でも気軽に、仲間とも楽しく。空間設計とイベント設計で配慮

実装ポイント (次フェーズの課題)

1. プレ運営型 (1 年限定の実証実験) を先行
2. パートナー連携：自治体、高齢者支援団体、DIY 企業など
3. サブスクモデル検討：定額＋都度利用併用で LTV 最大化

この構想は、定年後の新たなアイデンティティ形成支援という社会課題の解決にも資する提案です。ご希望あれば、この構想を元に事業計画書形式やプレゼン資料スライド構成の作成も可能です。何か特定の資料フォーマットでの展開をご希望ですか？

15.1.2 Chain-of-Thought (CoT) (第 16.2 節)

Chain-of-Thought (CoT) は、タスク解決に必要な思考ステップを逐次的なプロセスとして記述する手法であり、主に数理問題や論理推論などに用いられる。TAL は、CoT の逐次思考の流れを保持しつつ、それらのステップを上位の構造化フレーム内に統合し、文脈・目的・評価軸まで一体化して設計できる。以下に、典型的な CoT 例と TAL 変換例を示す。

CoT の例

Q:

5 人の中から 2 人の委員を選ぶ方法は何通りありますか？

A:

ステップ 1: 「5 人の中から 2 人を選ぶ組み合わせなので、組み合わせの公式を使います。」

ステップ 2: 「公式は $C(n, r) = n! / (r!(n-r)!)$ です。」

ステップ 3: 「 $5! / (2! \times 3!) = (5 \times 4) / (2 \times 1) = 10$ 通りです。」

最終答え: 10 通り。

Q:12 人の中から 3 人の委員を選ぶ方法は何通りありますか？

A:

CoT の TAL ラッピング

```
{
  "original_prompt": "12 人の中から 3 人の委員を選ぶ方法は何通りありますか？段階的に説明してください。",
  "goal": {
    "goal": "12 人から 3 人を選ぶ組み合わせの数を、段階的な思考手順で説明し、最終的な答えを提示する。",
    "note": "単に結果を出すだけでなく、過程を明確に示すことを重視しています。"
  },
  "evaluation_criteria": {
    "evaluation_criteria": [
      "組み合わせ公式の適用が正確である",
      "各ステップが論理的に順序立てられている",
      "計算が正確である",
      "最終答えが明確かつ簡潔"
    ],
    "note": "CoT の要として、説明の流れ・正確性・結論の明確さが評価ポイントになります。"
  },
  "output_format": {
    "output_format": {
      "type": "chain_of_thought_explanation",
      "fields": ["ステップ 1", "ステップ 2", "ステップ 3", "最終答え"],
      "note": "各段階を区切って示し、最後に結論をまとめます。"
    },
    "note": "段階的な説明と結果提示の標準フォーマットです。"
  },
  "z_axis": {
    "Structure": {
      "values": ["問題認識", "公式適用", "計算展開", "結論提示"],
      "note": "思考の骨組みを Z 軸として可視化しています。問題 → 方法 → 計算 → 結論の 4 段階構造です。"
    },
    "note": "Z 軸は、CoT 的思考の流れを体系化するためのものです。"
  },
  "example": {
    "question": "5 人の中から 2 人の委員を選ぶ方法は何通りありますか？",
    "steps": [
      "ステップ 1: 5 人の中から 2 人を選ぶ組み合わせなので、組み合わせの公式を使います。",
      "ステップ 2: 公式は  $C(n, r) = n! / (r!(n-r)!)$  です。",
      "ステップ 3:  $5! / (2! \times 3!) = (5 \times 4) / (2 \times 1) = 10$  通りです。"
    ],
    "final_answer": "10 通り",
    "note": "この例は、AI が求めるべき CoT の思考展開を示しています。"
  }
}
```

CoT の TAL ラッピングの実行結果

以下は「12 人の中から 3 人の委員を選ぶ方法は何通りありますか？」という問いに対して、段階的な思考手順（Chain of Thought）に基づいた解答です。

ステップ 1: 問題の認識（問題認識）

12 人の中から順序を考えずに 3 人を選ぶ方法を求めるので、**組み合わせ（コンビネーション）**の問題です。→ **順序を考慮しない選び方**のため、「**組み合わせの公式**」を使います。

ステップ 2: 公式の適用（公式適用）

組み合わせの公式は以下の通りです：

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}$$

ここで、 $n = 12$ 、 $r = 3$ なので、

$$\binom{12}{3} = \frac{12!}{3!(12-3)!} = \frac{12!}{3! \cdot 9!}$$

ステップ 3: 計算の展開（計算展開）

12! と 9! をすべて展開せずに約分を使って計算します：

$$\frac{12 \times 11 \times 10 \times 9!}{3! \times 9!} = \frac{12 \times 11 \times 10}{3!} = \frac{1320}{6} = 220$$

最終答え: 結論の提示（結論提示）

220 通り

何か他の具体例でも練習してみますか？

15.1.3 ReAct（Reasoning and Acting）（第 16.3 節）

ReAct（Reasoning and Acting）は、AI が推論（Reasoning）と行動（Acting）を交互に繰り返すインタラクティブな枠組みであり、情報検索や動的判断を伴うタスクに強みを持つ。TAL は、この「推論 → 行動 → フィードバック」の循環プロセスを、**思考構造の一部として明示的に記述**し、さらに評価・文脈ブロックと統合してラッピングできる。ここでは、ReAct の事例と TAL 適用例を提示する。

ReAct の例

Q:

日本の 2000 年時点での一人あたり GDP を円ベースで計算してください。

A:

[観察] 一人あたり GDP を計算するには、日本の 2000 年の GDP と人口が必要です。

[行動] 「日本 2000 年 GDP 円」「日本 2000 年 人口」で検索します。

[観察] 検索結果：GDP は約 500 兆円、人口は約 1 億 2700 万人です。

[思考] 一人あたり GDP = GDP ÷ 人口。

[計算] 500 兆円 ÷ 1 億 2700 万人 ≈ 約 393 万円。

[最終答え] 日本の 2000 年時点の一人あたり GDP は約 393 万円です。

Q:

日本の 2025 年 1 月時点での一人あたり GDP を円ベースで計算してください。

ReAct の TAL ラッピング

```
{
  "original_prompt": " 日本の 2025 年 1 月時点での一人あたり GDP を円ベースで計算してください。 ",
  "goal": {
    "goal": " 最新の日本の GDP と人口データを検索し、一人あたり GDP を円ベースで算出して提示する。 ",
    "note": " 検索データが必須なので、検索 → 推論 → 計算 → 結果の一連の流れを期待します。 "
  },
  "evaluation_criteria": {
    "evaluation_criteria": [
      " 検索が適切に行われている ",
      "GDP および人口データが正確 ",
      " 計算が正しい ",
      " 説明が明快である "
    ],
    "note": " 単に結果を出すのではなく、プロセス（検索・思考・計算）がしっかり表現されているか確認します。 "
  },
  "output_format": {
    "output_format": {
      "type": "react_explanation",
      "fields": [ " 観察 ", " 行動 ", " 観察 ", " 思考 ", " 計算 ", " 最終答え " ],
      "note": "ReAct 形式の標準的な思考・行動サイクルを繰り返す構造です。 "
    },
    "note": "AI が段階的に検索・推論・計算するための明確な枠組みです。 "
  },
  "example": {
    "question": " 日本の 2000 年時点での一人あたり GDP を円ベースで計算してください。 ",
    "steps": [
      " 観察: 一人あたり GDP を計算するには、日本の 2000 年の GDP と人口が必要です。 ",
      " 行動: 「日本 2000 年 GDP 円」「日本 2000 年 人口」で検索します。 ",
      " 観察: 検索結果: GDP は約 500 兆円、人口は約 1 億 2700 万人です。 ",
      " 思考: 一人あたり GDP = GDP ÷ 人口。 ",
      " 計算: 500 兆円 ÷ 1 億 2700 万人 ≈ 約 393 万円。 ",
      " 最終答え: 日本の 2000 年時点の一人あたり GDP は約 393 万円です。 "
    ],
    "note": " 検索と計算が一体化した ReAct プロセスの流れを示す例です。 "
  }
}
```

ReAct の TAL ラッピングの実行結果

以下は、2025 年 1 月時点における日本の一人あたり名目 GDP（円ベース）を算出するための段階的な思考プロセスです。

観察: 必要なデータの確認

一人あたり名目 GDP を計算するには、以下の 2 つのデータが必要です：

- ・ 名目 GDP (円)
- ・ 人口 (人)

行動: データの取得

1. 名目 GDP (2025 年)

2025 年の日本の名目 GDP は約 690.84 兆円と報告されています。(Reuters)

2. 人口 (2025 年 1 月時点)

2025 年 1 月 1 日時点の日本の人口は約 1 億 2,592 万人と推定されています。(カントリーメーターズ)

思考: 計算式の確認

一人あたり名目 GDP は以下の式で計算されます：

$$\text{一人あたり名目 GDP} = \frac{\text{名目 GDP (円)}}{\text{人口 (人)}}$$

計算: 実際の数値を代入

- ・ 名目 GDP : 690.84 兆円 = 690,840,000,000,000 円
- ・ 人口 : 125,920,000 人

$$\frac{690,840,000,000,000}{125,920,000} \approx 5,487,000 \text{円}$$

最終答え

2025 年 1 月時点の日本の一人あたり名目 GDP は約 548 万 7,000 円です。

この数値は、前年と比較して増加傾向にあり、経済成長やインフレの影響を反映しています。また、為替レートの変動や購買力平価 (PPP) を考慮すると、国際的な比較においては異なる結果となる可能性があります。

15.1.4 T5 (Text-to-Text Transfer Transformer) (第 16.4 節)

T5 (Text-to-Text Transfer Transformer) は、「すべてのタスクをテキスト変換問題として扱う」というコンセプトで設計された統一モデルである。あらゆる入力と出力がテキストで表現される点が特徴。TAL は、単なる入出力のテキスト変換を超え、**タスクの本質的な目的・背景・思考過程**まで含めた **OS 的な構文設計**を提供する。ここでは、T5 型プロンプト例と、TAL ラッピング後の構造例を提示する。

T5 の例

Q: ポエム

森の奥で小さな妖精が、月の光をあびてダンスしている。夢と魔法の夜が静かに広がる。

A: 200 文字程度で社説風に書き換え

深い森の静寂の中、小さな妖精が月光に照らされ舞う光景は、現実の喧騒を忘れさせる幻想の象徴だ。夢と魔法に包まれたその夜は、自然が奏でる静かな調和を私たちに思い出させる。

Q: ポエム

花の野原で ちいさな女の子が
ひらひら舞うチョウチョを追いかける
笑い声は風にのって 空まで届く
ふたりの午後は そっと魔法に染まる

A: 200文字程度で社説風に書き換え

T5 の TAL ラッピング

```
{
  "original_prompt": "ポエム\n花の野原で ちいさな女の子が \nひらひら舞うチョウチョを追いかける\n↪ \n笑い声は風にのって 空まで届く\nふたりの午後は そっと魔法に染まる\n\nA:\n↪ 200文字程度で社説風に書き換え",
  "goal": " 詩的な文章を、200 文字程度の社説風の文章に変換する。",
  "output_format": {
    "type": "text_to_text",
    "fields": [" 入力ポエム", " 社説風変換"]
  },
},
"example": {
  " 入力ポエム": " 森の奥で小さな妖精が、月の光をあびてダンスしている。夢と魔法の夜が静かに広がる。",
  "社説風変換": "深い森の静寂の中、小さな妖精が月光に照らされ舞う光景は、現実の喧騒を忘れさせる幻想の象徴だ。
  ↪ 夢と魔法に包まれたその夜は、自然が奏でる静かな調和を私たちに思い出させる。"
}
}
```

T5 の TAL ラッピングの実行結果

入力ポエム

花の野原で ちいさな女の子が
ひらひら舞うチョウチョを追いかける
笑い声は風にのって 空まで届く
ふたりの午後は そっと魔法に染まる

社説風変換

花咲く野原で蝶を追う少女の姿は、日常の忙しさを忘れさせる穏やかな風景だ。その無垢な笑い声が風に運ばれ空に溶けていく午後、自然と触れ合う時間の大切さと、そこに宿る小さな奇跡を私たちは見直すべきだ。

この方向で他の詩も変換していきましょうか？

15.2 TAL の拡張応用（関数やプログラムの組み込み）

15.2.1 はじめに：TAL における「関数」の概念

- TAL は思考の構造化を行うが、従来の命令型プロンプトとは異なり、「関数的処理」を自然言語ベースで統合できる。

- ・本節では、TAL 内に**関数的な処理単位**を組み込み、**データ駆動型プロンプト**を実現する試みを示す。

15.2.2 関数モジュールの定義と統合例

TAL における「関数の組み込み」は、従来のプロンプトに対し、より柔軟で階層的なデータ処理を実現する。ここでは、**通貨換算**および**消費税加算**という 2 つの関数的処理をモジュール化し、それを TAL ラッピングによって統合する例を提示する。

関数をラッピングした TAL プロンプトの例

```
{
  "original_prompt": " 指定された金額を円に換算し、
  さらに消費税 10% を加えた最終金額を計算してください。",
  "external_parameters": {
    "amount": 12
  },
  "function_modules": [
    {
      "name": "currency_conversion",
      "description": " 指定された金額を為替レートに基づいて換算する。",
      "parameters": {
        "amount": "{{amount}}",
        "from_currency": "USD",
        "to_currency": "JPY",
        "exchange_rate": 143
      },
      "returns": "converted_amount"
    },
    {
      "name": "tax_inclusion",
      "description": " 指定された金額に消費税を加算する。",
      "parameters": {
        "amount": "{{converted_amount}}",
        "tax_rate": 0.10
      },
      "returns": "final_amount"
    }
  ],
  "execution_goal": " 元の金額、換算結果、税込金額をすべて表示する。",
  "output_format": {
    "currency_conversion": {
      "元の金額": "{{amount}} USD",
      "換算レート": "1 USD = {{exchange_rate}} JPY",
      "換算結果": "{{converted_amount}} JPY"
    },
    "tax_inclusion": {
      "適用税率": "{{tax_rate * 100}}%",
      "税込金額": "{{final_amount}} JPY"
    }
  }
}
```

```

    },
    "summary": "{{amount}} USD は {{converted_amount}} 円に換算され、
    税込金額は {{final_amount}} 円です。"
  }
}

```

実行結果

以下は、指定された金額 12 USD を円に換算し、消費税 10% を加算した結果です。

通貨換算結果

- 元の金額: 12 USD
- 換算レート: 1 USD = 143 JPY
- 換算結果: 1,716 JPY

税込金額

- 適用税率: 10%
- 税込金額: 1,887.6 JPY

まとめ

12 USD は 1,716 円に換算され、税込金額は 1,887.6 円です。

他の通貨でも試してみますか？

15.2.3 プログラム関数の TAL ラッピング

簡単なプログラムであれば、LLM 内で仮想的にプログラムをトレース実行できることが確認できた。これは従来のプロンプト手法にはない新たな領域である。

プログラムをラッピングした TAL プロンプトの例

```

{
  "original_prompt": " 与えられた数値リストをバブルソートで昇順に並び替えてください。",
  "external_parameters": {
    "numbers": [5, 2, 9, 1, 7]
  },
  "function_modules": [
    {
      "name": "bubble_sort",
      "description": " 数値リストをバブルソートで昇順に並べ替える。",
      "parameters": {
        "arr": "{{numbers}}"
      },
      "implementation": "def bubble_sort(arr):\n
        n = len(arr)\n
        for i in range(n):\n
          for j in range(n - i - 1):\n
            if arr[j] > arr[j + 1]:\n
              arr[j], arr[j + 1] = arr[j + 1], arr[j]\n
        return arr",
      "returns": "sorted_numbers"
    }
  ]
}

```

```

    }
  ],
  "execution_goal": " 元のリストとソートされた結果を表示する。",
  "output_format": {
    "bubble_sort": {
      " 元のリスト": "{{numbers}}",
      " ソート後のリスト": "{{sorted_numbers}}"
    },
    "summary": " バブルソートの結果、{{numbers}} は {{sorted_numbers}} に並び替えられました。"
  }
}

```

実行結果

以下は、指定された数値リスト [5, 2, 9, 1, 7] をバブルソートで昇順に並べ替えた結果です：

バブルソート処理結果

- ・ 元のリスト: [5, 2, 9, 1, 7]
- ・ ソート後のリスト: [1, 2, 5, 7, 9]

summary:

バブルソートの結果、[5, 2, 9, 1, 7] は [1, 2, 5, 7, 9] に並び替えられました。

他のソートアルゴリズムとも比較してみますか？

15.2.4 今後の展望

本章の事例は、TAL が単なる構造化プロンプト言語を超え、外部パラメータや仮想的な関数処理をも統合できる柔軟性を持つことを示した。特に、バブルソートのようなアルゴリズム的処理を“頭の中でトレース実行”する LLM の特性と、TAL の階層的思考設計が結びつくことで、これまでにない応用領域が開かれている。

今後の展望としては：

- ・ **TAL 内部 DSL（ドメイン固有言語）的な拡張**により、たとえば「自然言語ベースの簡易スクリプト」や「特定分野向けのロジックブロック」など、より高度な思考プロセスの記述が可能になる。
- ・ 関数的処理だけでなく、**条件分岐・ループ・状態管理**といったダイナミックな思考制御の実装も視野に入る。
- ・ LLM の進化とともに、「**TAL 内仮想マシン**」的な仕組みが構想され、より複雑なプロンプトが内部的に自律実行される可能性も見えてくる。

これらは、「AI が単なる命令の受け手である段階」から、「共に思考し、動的に実行するパートナー」への進化を後押しするものとなる。TAL はその橋渡し役として、今後も新たな応用と挑戦を通じて真価が問われるだろう。

本論文では、TAL が単なる思考構造の提示にとどまらず、ロジックを内包する**動的プロンプト**へと進化し得る可能性を提示した。特に、関数的要素を持つ TAL プロンプトが、LLM 内部でロジックを逐次シミュレートし、精密な処理結果を生成できることが確認できた点は、プロンプト設計の新たな地平を切り開く成果である。今後は、この「構造 × ロジック融合型プロンプト」の適用範囲をさらに広げ、

複雑なタスクや実環境応用への発展を探求していく。

16. 参考文献 (References)

16.1 ToT (Tree of Thoughts)

Yao, S., Yu, D., Zhao, J., Shafran, I., Griffiths, T. L., Cao, Y., ... & Narasimhan, K. (2023). Tree of Thoughts: Deliberate Problem Solving with Large Language Models. arXiv preprint arXiv:2305.10601. <https://arxiv.org/abs/2305.10601>

16.2 CoT (Chain of Thought)

Wei, J., Wang, X., Schuurmans, D., Bosma, M., Ichter, B., Xia, F., ... & Le, Q. V. (2022). Chain of Thought Prompting Elicits Reasoning in Large Language Models. arXiv preprint arXiv:2201.11903. <https://arxiv.org/abs/2201.11903>

16.3 ReAct (Reasoning and Acting)

Yao, S., Zhao, J., Yu, D., Chen, S., Zhao, C., Li, Q., ... & Rudin, C. (2022). ReAct: Synergizing Reasoning and Acting in Language Models. arXiv preprint arXiv:2210.03629. <https://arxiv.org/abs/2210.03629>

16.4 T5 (Text-to-Text Transfer Transformer)

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., ... & Liu, P. J. (2020). Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Journal of Machine Learning Research, 21(140), 1–67. <https://arxiv.org/abs/1910.10683>