

特集 大学授業における生成AIの利活用と教育評価を考える

# ChatGPTを活用した情報システム構築型演習の 支援と課題



大阪電気通信大学 情報通信工学部教授 竹内 和広

## 1. はじめに

本学の情報通信工学部情報工学科(1学年の定員は160名。以下、単に学科と呼びます)では、2023年度から学科にChatGPTを導入しました。具体的には、学科全体として学科教員と学科学生全員の情報交換のためのコミュニケーションツールとしてSlackを導入していることを背景として、学部長が主導してSlackにChatGPTボットのチャンネルを設置しました。著者が担当する授業では、左記のSlack設置のChatGPTの活用について、2023年度前期の講義科目(2年生配当の1科目および3年生配当の1科目)と演習科目(3年生配当の1科目)で紹介し、積極的な利用を呼びかけました。その結果、授業や授業課題で疑問に思った点をChatGPTに問い合わせることも見られ、講義内容に関するFAQをリアルタイムの共有できる環境となっています。

他方、学生の多くはChatGPTを個人のアカウントで自分専用のアシスタントとして利用し、疑問点を他者に知られることを嫌う傾向が見られるようになってきました。本稿では、このChatGPTの個人活用について記述します。具体的には、2023年度前期および2024年度前期に実施した演習科目にて、後者ではChatGPTの個人活用を積極的に取り入れて、2年間で対照性のある実施ができました。その知見は第2節で記述します。また、2023年度後期では、プレゼминаール(卒業研究の準備活動として3年生を卒業研究実施予定の研究室に仮配属して実施)で、ChatGPTを活用した活動を行いました。その知見については、第3節で記述します。

## 2. 情報システム構築型演習へのChatGPT導入

学科では、図1のように1年生から2年生まで、週2回のC言語を基本としたプログラミング演習を継続的に実施するカリキュラムを提供しています。3年前期にはプログラミング演習を総括する形で、テーマを選択した演習を行います。著者が担当する情報システム設計演習は、その選択肢の一つであり、受講生が自ら構築したい情報システムを自由に決定し、構築の実践を目指すグループワークを伴う演習科目です。具体的には、部分モジュールの実装を通じて実験的にシステム設計を行い、可能であれば、試作システムの完成を目指します。

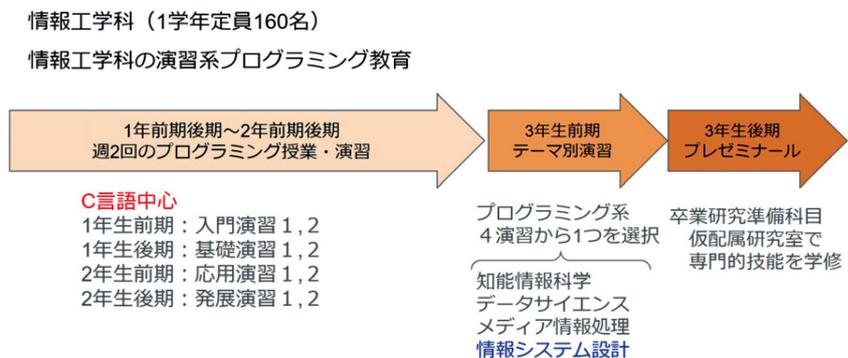


図1 学科におけるプログラミング演習科目

2023年度の実施では実質受講者は17名であり、2024年度の実施では23名でした。両年とも5班が形成され、次ページ表1のような情報システムの設計がなされました。なお、表1中に\*で示したシステムは実装の実践的テンプレートとなる書籍を参考にシステムを実装した班であり、Bは部分モジュールの試作・実験とそれに基づくシステム設計に留まった班、Aは左記だけではなくシステムの試作まで達成できた班です。表1が示すよ

表1 情報システム設計演習で各班が設計した情報システム

2023年度実施		2024年度実施	
Web掲示板*	A	Web掲示板*	A
IoT端末通信システム*	B	OSの実装*	A
機械学習演習支援システム*	B	全文検索システム	A
図書管理システム	B	時間割配信システム	A
冷蔵庫余り物レシピ提案システム	B	アルバイトシフト管理システム	B

うに、2024年度は大幅にシステム試作まで達成した班が増える結果となりました。

2024年度から90分15回授業から105分13回授業と授業体制が変化しましたが、教員と正規TA(修士学生)とサポート学生(筆者の研究室の4年生2名)の計4名のスタッフで各班を支援する体制は両年で同じでした。

当該科目では、自分たちの課題とするシステムの概要を決定するために、班相互にプレゼンテーションを繰り返しながら、適切な参考文献を調査する作業に授業期間中の約4割を費やすスケジュールになっています。例年、学生が希望するシステム構築の実践的テンプレートとなる書籍を見つけることは難しく、仮に見つかったとしても、それを読みこなし実装するにはスタッフの多大なサポートが必要となることが問題でした。

2024年度は、スタッフが第3節で述べるような課題によりChatGPTの使い方を理解した上で、受講者にChatGPTの利用法を適宜解説するサポートを行いました。そのことで受講者が自立的にChatGPTを活用できるようになり、その結果、多くの班がシステムの試作まで達成できたと考えています。

実際に受講者にChatGPTの有用性を5段階評価させたアンケートを実施したところ、回答者22名中19名(評価4が11名、評価5が8名)が有用であったと回答しました。同時に実施した自由記述アンケートでは、ChatGPTのポジティブな点に関して、システム構築に必要な知識や理解を深めることができる点があげられました。また、プログラミング・実装の観点からは、コード生成や改善提案によりプログラムの効率化や質の向上したことや、作業の行き詰まりの解消、コーディングの時間短縮といった点があげられました。

他方、自由記述アンケートに記載された内容をChatGPT-4oによって分類した結果、22名分の自

由記述から計26件分の問題点の指摘が抽出されました。その分類結果を図2に示します。問題点は、(a)不正確さと信頼性の問題、(b)意図の伝達と理解の問題、(c)学習と理解の阻害、(d)デバッグや環境構築の難しさ、(e)ループと非効率性の問題、(f)ユーザーのスキルに依存、として整理しています。これらの区分名は「ループ」といった自由記述のままの表現も残っているが、意見を大きく分けると、意図を正確に伝えること及び回答理解の難しさ(b, c)と、提案プログラムの適用に際して修正やデバッグに高度なスキルが必要となる点(a, d, e, f)に集約されます。

以上のように、スタッフから個別アドバイスを提供しながらChatGPTの個人活用を推進することにより、受講生がChatGPTを効果的に活用するスキルを向上させたと考えられます。特に、必要な技術文献を読む知識をChatGPTに個別に問い合わせることで、システムの構造的理解を深める学習負担を軽減したと捉えています。また、システム構築の部分要素に対するプログラム自動生成もシステム完成に有効に働いたものと考えています。他方、このようなChatGPTの活用には、提案されたプログラムに対する修正・デバッグ作業が必要であり、何らかのプログラミング言語でのプログラミング力を持っていることが前提となることも明らかになりました。

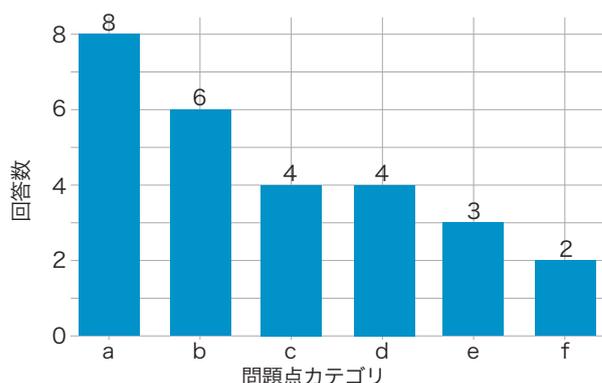


図2 自由記述アンケートの指摘する問題点

### 3. プロンプト記述の習熟課題

2023年度後期に実施したプレゼминаールでは、発表者の研究室に仮配属された11名の学生が、発表者の専門である自然言語処理と人工知能の基礎を学習します。また、人工知能システムを実装する上で重要となるWebシステムの構築を例年演習しています。2023年度後期は、ChatGPTを活用する課題を実施しました。この経験が第2節で



図3 ビジュアルブロックの組み合わせ例 (左) 対応するWebコンテンツ (右)

説明したスタッフによるChatGPT個人活用サポートの実施力の源泉になったと考えています。

課題内容は、研究室で開発したビジュアルプログラミングブロック<sup>[1]</sup>を基に、学生が新たなブロックを作成するというものです。これらのブロックはBlockly<sup>[2]</sup>を使って作成しています。例えば、図4のようにブロックを組み合わせることで、ChatGPTでプログラムを生成するためのプロンプトを出力します。ブロックの設計では、WebコンテンツのHTML、CSS、JavaScriptを生成するために、適切なプログラムを促すプロンプトを試行錯誤する必要があります。一方、Webコンテンツの基本構造であるテンプレートは既にブロック化されて提供されているため、学生は特定のWebコンテンツやモジュールに集中して、ブロックやプロンプトの設計に取り組むことができます。

このような課題実践の結果、自己紹介ページ作成ブロック、アンケートフォーム作成ブロック、データ検索機能ブロックなどが提案され、全員がWebコンテンツ生成のためのプロンプト記述を学ぶことができました。

今回紹介した実践はWebコンテンツを例としていますが、多くの情報システム構築においても、そのシステムに対応する構造的なテンプレートに相当する知識が存在すると考えられます。そのようなテンプレート構造をあらかじめブロックとして整備しておくことができれば、全体構造の中の部分的なモジュールの生成からプロンプト記述の学習を始められる教材を開発できると考えています。

#### 4. まとめ

本稿では、情報システム構築型演習におけるChatGPTの個人活用の利点と課題について報告しました。また、ChatGPTのプロンプト記述スキルを向上させるための方向性の一つとして、具体的な課題実践例も紹介しました。

#### 参考文献および関連URL

- [1] 宿谷海斗, 竹内和広: WEBデザインに向けたビジュアル開発ブロックの設計, 情報処理学会コンピュータと教育研究会172回研究発表会, 2023
- [2] Blockly <https://developers.google.com/blockly>

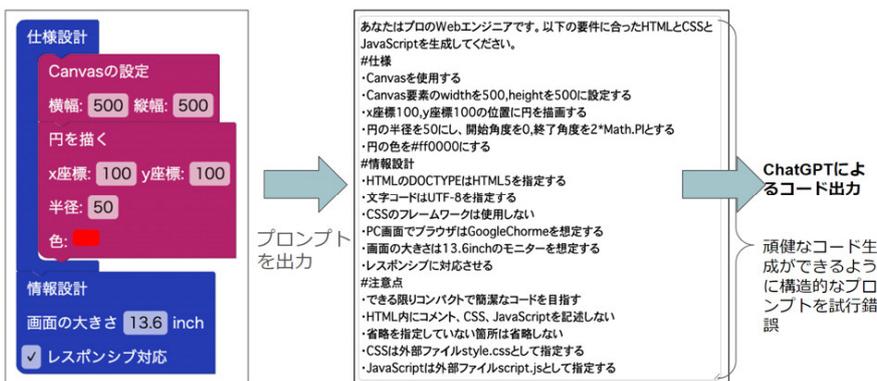


図4 ブロックからのプロンプト出力