



生成 AI を効果的に活用した プログラミングの授業実践

島根県立情報科学高等学校教諭 渡部 謙

1. はじめに

島根県立情報科学高等学校は、島根県安来市に位置し、マルチメディア科、情報処理科、情報システム科を有する専門高校です。本校は、「普通教育ならびに情報・ビジネスに関する専門教育を施し、健康で心豊かな人間性を育成すること」を教育目標に掲げ、専門的な知識・スキル習得のみならず、情報技術を活用した課題解決能力育成、地域社会との連携、ICTを活用した教育を推進しています。

毎年12月頃に開催する「情報ITフェア」では、約1,000人から2,000人のお客様に会場いただき、プログラミングやITを活用した講座を生徒が企画・運営し、学びを实践する取り組みを実施しています。近年、社会の情報化が急速に進展するなかで、本校ではプログラミング教育における生徒の学習意欲の維持・向上や、変化の速い情報技術に対応するための教員の知識・スキル更新といった課題に直面しています。その解決策の一つとして生成AIの活用注目しました。本稿では、本校3年情報システム科「総合実践」の授業における、生成AIを活用したゲームプログラミングの実践事例を紹介し、教育現場における生成AI活用の可能性を探ります。本稿が、生成AIの授業導入に際して、読者の皆様の参考となれば幸いです。

2. 事例紹介：3年情報システム科総合実践

(1) 授業の背景と生成AI活用への着想

本事例は、3年情報システム科の「総合実践」(3単位)で、DXRubyを用いたゲーム開発に取り組んだものです。システム開発演習を通して、



昨年度の情報ITフェアのチラシ

生徒がユーザーを意識したシステム開発ができるようになることを目標としています。実は、私自身もプログラミングを指導するのは7年ぶりであり、授業開始当初はプログラミングを教えることに大きな不安を抱えていました。授業中、エラーコードが出た際に生徒と一緒に原因を考え、解決するのに多くの時間を要し、授業全体を俯瞰して多くの生徒に気を配ることが難しい場面がありました。また、私と同じコードを打っているつもりでもエラーが出てしまう生徒がおり、その個別の原因を探るのにも時間がかかり、これらの問題を抱えながら授業改善の方法を模索していました。このような状況のなか、以前から私が個人的にエラーの原因特定などに生成AIを活用していた経験から、生徒自身が生成AIを使えるようになれば、エラー解決が迅速になり、教員はより大きな困難に直面している生徒に対応できるようになるのではないか、という可能性を感じました。この気づきが、本授業で生徒に生成AIを活用させてみようと考えたきっかけです。

(2) 授業の概要と流れ

授業は1学期から3学期に分けて行いました。

1学期は、全員共通のプログラムを打ち、簡単なゲームを制作し、プログラミングの基礎とDXRubyの扱いに慣れることを目指しました。この段階から、生徒たちは生成AIを活用し始めました。2学期は、生徒それぞれがオリジナルのアイデアに基づいたゲームの企画・実装に取り組みました。そして3学期には、完成したゲームの体験会を実施し、生徒同士でプレイし、フィードバックを行いました。最後に、制作したゲームの内容や制作過程、苦労した点、工夫した点などを発表する機会を設けました。

(3) 生成AIの活用方法と生徒の取り組み

本授業では、主にChatGPTとCopilotを活用しました。これらのツールを選定した理由は、無料で使用できること、日本語に対応していること、そして文部科学省が示す基準に準拠していると判断したためです。授業の冒頭では、文部科学省が公開している生成AIに関する資料の内容を基に、生成AIを利用するうえでの注意点や倫理について事前学習を行いました。その際にプロンプトの入力例を示しながら、ハルシネーションについても説明を行い、生成AIの活用方法、プロンプトの入力の仕方について考える時間を設けました。その後の授業で、生徒たちは、生成AIに対して多様なプロンプトを与えていました。主な活用方法は以下の三点です。

●エラーコードの原因特定と修正：プログラム実行時に発生したエラーコードを生成AIに入力し、その原因と解決策について解説を求めたり、修正後のコードを提案してもらったりしました。具体的なプロンプトとしては、「〇〇（エラーコード）というエラーが出ました。このエラーの原因と解決策を教えてください」といったものがありました。

●実装したい機能のコード生成：実現したいゲームの機能や動作を具体的に記述し、そのためのDXRubyのコード生成を依頼しました。たとえば、「〇〇（作りたいゲームの機能）を実現するためのDXRubyのコードを書いてください」といった指示がありました。

●コード生成以外の活用：ゲームのアイデア出しのブレインストーミングとしてテーマやコンセプトについて意見を求めたり、デバッグの補助として活用したりしました。また、生成されたコードや既存のコードの解説を求めることで、ソースコードの理解を深めるためにも生成AIを利用しました。

これらの活用場面において、私は生徒の様子を確認し、どのようなことを生成AIに聞いたのか、なぜそれが必要だと考えたのかを尋ねました。また、生成AIが返した結果がどのような内容であったかを説明させ、その場で生徒の理解度を確認しました。生徒の回答に対してさらに「それはどういう意味?」「他にどんな方法が考えられる?」といった問いを投げかけることで、理解の深まりを捉えるよう努めました。

【生徒のプロンプトの入力例とその回答（一部）】

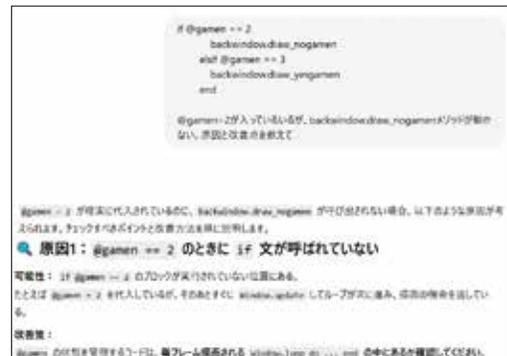


図1



図2

(4) 生成されたコードの扱いと生徒の学び

生成 AI が生成したコードは、基本的な機能についてはそのまま使えるレベルであることも多かったのですが、生徒が既に作成している既存のプログラムに組み込む際に、うまく動作しないケースが見られました。このような状況に直面した生徒たちは、生成 AI が生成したコードの挙動を理解し、自身のプログラムに合わせて修正・改善を行う必要がありました。具体的には、変数名の不一致を直したり、エラーが出ている箇所を修正したり、必要な処理を追加したりといった作業が発生しました。生成 AI が生成したコードを利用することのメリットとして、コードを一から記述する時間を短縮できること、そしてこれまで知らなかった新しい技術や構文に触れるきっかけになることが挙げられます。一方で、生成 AI が生成したコードの品質にはばらつきがあるため、生成されたコードを鵜呑みにせず、その内容を理解し、既存のコードとの整合性を自ら判断する必要があるというデメリットもありました。しかしながら、このデメリットは生徒の学びという点では非常に重要な役割を果たしており、問題が解決するように粘り強くトライ＆エラーを繰り返すことができました。また、我々教員も生徒にヒントを与えたり、一緒に問題解決に取り組んだりすることで、教員の指導の幅を広げてくれる機会となりました。

(5) 生徒の成果と反応

生徒が制作したゲームの種類は、シューティングゲーム、育成ゲーム、パズルゲーム、対戦ゲーム、アクションゲーム、的当てゲームなど、生徒の関心やアイデアを反映した多様なものとなりました。生成 AI を活用した授業に対する生徒の主な感想は、以下の通りです。

- 「1学期で学んだプログラムの基礎を応用することができました。」
- 「生成 AI を活用することで、効率的にゲームを作成することができました。」
- 「自分の作りたいゲームを形にすることができて、とても嬉しかったです。」

- 「ゲームを作る大変さと楽しさを学ぶことができました。」
- 「生成 AI に頼りすぎることなく、コードを理解することの重要性を学びました。」
- 「生成 AI を活用することで、学びの幅が広がりました。」
- 「ChatGPT では答えてくれないところも、Copilot だと答えてくれることがあり、生成 AI をうまく活用することの大切さを知ることができました。」
- 「AI をうまく活用して、自分の思い描いていたゲームを作ることができてよかった」

これらの感想からは、生成 AI がゲーム制作の効率化やモチベーション向上に貢献し、生徒自身の学びを深める一助となっていることが伺えます。

(6) 事例から見えた成功要因

本事例を通して見えてきた生成 AI 活用の成功要因はいくつかあります。まず、生成 AI を活用することで、生徒がコードの記述にかける時間を短縮し、その分、ゲームの企画やデザイン、機能の実装といった創造的な活動に時間を費やすことができた点です。また、生成 AI がエラー解決や新しいコードの提示を行うことで、生徒がプログラミングの難しさに挫折することなく、主体的に学習を進めることができた点も大きかったです。さらに、生成 AI が提示する多様なコードや表現に触れることで、生徒のプログラミングスキルや問題解決能力の向上につながったと考えられます。

(7) 事例から見えた課題と対策

一方で、課題も明らかになりました。第一に、生成 AI では、実装したいことをプロンプトに入力すればコードが生成されるため、それが生徒の学びの本質につながるのか、という問いです。第二に、生成 AI の利便性ゆえに、生徒がコードの内容を深く理解せずに「コピペ」に終始してしまう可能性があることです。第三に、生成 AI が生成するコンテンツに関する著作権や倫理的問題への配慮です。これらの課題に対し、本授業では生成 AI が生成したコードはあくまで参考とし、鵜呑みにせず必ず内容を理解するように指導しま

した。生徒に生成されたコードについて解説をさせ、自力でコードを修正・応用できるようサポートしました。また、著作権に関する基本的な知識を教え、倫理的な問題についても生徒と共に考える機会を設けました。

3. おわりに

本事例の実践を通して、生成 AI が専門高校のプログラミング教育における、生徒の学習意欲向上や効率化に有効であるという手応えを得ました。特に、私自身が抱えていた「個別のエラー対応に時間がかかり、生徒への対応が難しくなる」という課題に対して、生徒自身が生成 AI を活用してエラーを解決できるようになることで、教員はより多くの生徒の状況を把握し、全体の進捗管理やより発展的な内容の指導に時間を割くことができるようになって感じています。生成 AI は、コードを生成するだけでなく、アイデアを広げたり、難解な部分の理解を助けたりと、学習のさまざまな段階で強力なツールとなり得ます。これは、プログラミングに苦手意識を持つ生徒にとって、学習のハードルを下げる効果も期待できます。また、システム開発をする場合には、生成 AI を活用することで、自分の表現したいことを短時間で実現できるという、開発の成功体験が得られることも大きな収穫でした。

さらに、この実践を通して強く感じたのは、生成 AI が生徒の学びを深めるツールであると同時に、教員自身の学びを深めるツールでもあるという点です。特に、私のように特定分野（ここではプログラミング指導）にしばらくブランクがあったり、苦手意識があったりする先生方には、ぜひ活用をお勧めしたいツールです。生成 AI と対話することで、自分自身の知識や解釈を整理し、深めることができます。周りにすぐに質問できる同僚がいない状況でも、生成 AI は強力な壁打ち相手となります。たとえば、生徒に解説する内容について、事前に音声入力で生成 AI に説明を聞かせ、その伝わりやすさや間違いがないかななどを指摘してもらおうといった使い方も考えられます。

このように、生成 AI は教育に多様な可能性をもたらしますが、あくまで学習をサポートするツールの一つとして捉えることが重要です。学習方法には、問題集を解いて理解を深める、他の人に質問して新たな視点を得るなどさまざまな方法がありますが、生成 AI を利用することも、その選択肢の一つとして位置づけられます。生徒が生成 AI に過度に依存し、自ら考える力を失わないよう、教員が適切に指導し、生成されたものを批判的に検討し、応用する力を育むことが重要です。

また、生成 AI を効果的に活用するためには、プログラミングにおける基礎的な知識が不可欠であると、この実践を通して強く感じています。基礎的な内容については、教員が生徒に問いかけながら理解を促す講義型の授業も引き続き重要です。そのため、私は講義によって基礎知識を生徒に身につけさせることを特に重要視しています。なぜなら、基礎知識がなければ、生成 AI に対して意図を正確に伝える質の良いプロンプトを作成することができないからです。生成 AI を単なる「答えの自動生成機」としてではなく、真に学習を加速させるツールとして使いこなすためにも、我々教員が土台となる基礎知識を丁寧に教える役割は、今後も変わらないどころか、より重要になると考えています。また、情報モラルや著作権といった倫理的な側面についても、継続的な指導が必要です。生徒の主体的な学びと生成 AI の利便性をどのように調和させていくかが、今後の教育における大きなテーマとなるでしょう。

今後も生成 AI の教育活用に関する知見が蓄積され、より効果的な指導法が確立されていくことを期待しています。本稿が、商業科の先生方にとって、生成 AI を自身の授業に取り入れるための一歩を踏み出すきっかけとなり、生徒たちの学びがさらに豊かになることを願っています。