**ガイダンス：プログラミング入門**

プログラミング入門の授業を始めたいと思います。最初に、このコースを履修するにあたってのガイダンスを行いたいと思います。このコースは、生成AIの利用を前提としたプログラミング講座です。

このプログラミング講座は、合計七つの講座で構成されています。

初級コースとしては、プログラミング入門、アルゴリズムとデータ構造、プログラミングを通じた問題解決という三つの講座があります。担当者は、熊本高専の村上ゆき先生、有明高専の加藤直子先生、そして私の三人です。

基本的には、プログラミング入門、アルゴリズムとデータ構造、プログラミングを通じた問題解決を順番に勉強してもらいたいと考えています。しかし、すでにプログラミングやアルゴリズムとデータ構造を勉強している人には、比較的優しい内容が多いと思われます。その場合、プログラミング入門とアルゴリズムデータ構造の内容は、それぞれ2週間程度で済むように分量を調整しています。

プログラミングを通じた問題解決には演習課題がありますので、それに取り組んでもらうため、3週間程度の時間がかかると思います。つまり、初級コースの三つの講座は、合計で7週間程度の時間を想定しています。

一方、中級コースは、オブジェクト指向プログラミング、テスト駆動開発、リファクタリングとコードの最適化、実践的なプロジェクト開発の四つの講座で構成されています。ただし、今学期はリファクタリングとコードの最適化を除いた三つの講座を勉強してもらいたいと考えています。

それぞれの担当者は、私と佐賀大学の大槻みか先生が分担しています。オブジェクト指向プログラミングとテスト駆動開発は、それぞれ2週間程度で勉強できるように分量を調整しています。

実践的なプロジェクト開発は、3週間程度の時間がかかると思います。これにより、中級の三つの講座は、リファクタリングとコードの最適化を除いた残りの三つで、14週のうち7週間程度かかることを想定しています。全体を通して、14週程度の時間がかかると考えています。

このビデオは、授業の最初にある「How to Build an App in Eighteen Minutes」というビデオです。これは、GitHubのCEOであるトーマス・ダムケさんによる、GitHub Copilotを用いたアプリケーション開発のライブデモンストレーションです。

講演は英語で行われますが、日本語字幕を表示することができます。皆さんも日本語字幕を表示して視聴していただければと思います。なお、日本語字幕の設定方法は解説ビデオに載っています。

このビデオを見ると、AIを活用したプログラミングや、AIとプログラマーによるペアプログラミングが実践されていることがわかります。AIを使ってコード生成やコードレビュー、テストコード生成などの様々な作業が支援されます。

このような支援を活用しながら、プログラミングとソフトウェア開発を進めることで、これまでのソフトウェア開発とは大きく異なる方法で進められることがわかります。これまでのプログラミング手順では、プログラマーがコーディングを行い、コンパイラーでコンパイルエラーを修正し、実行して動作確認を行っていましたが、しばしばエラーが発生します。

デバッグ作業を開始して、どこが間違っているかを見つけ、そこを修正するという大変な作業が発生します。これらすべては人間が行っていたわけですが、AIを使ってプログラミングをすると、人間はまず最初にコードの生成指示をAIに出します。

このコード生成指示に基づいて、AIがコードを生成します。人間はそのAIが生成したコードをコンパイルし、何かしらのコンパイルエラーが出た場合は、そのエラーをAIに修正指示として伝えます。AIはコンパイルが成功するようにコードを修正していきます。

修正が済んだら実行結果を確認し、うまくいけば問題ありませんが、うまくいかなかった場合には、その問題をAIに伝え、AIが高度な修正を行います。このように、プログラミング手順が根本的に変わっていきます。

これまでの従来のプログラミング手順とAIを活用したプログラミング手順では、人間が行うべき役割が根本的に変化します。人間に求められるのは、AIに対して適切な指示を出すことです。

また、AIが生成したコードや修正したコードの間違いを見抜く能力も求められます。その代わりに、自分でプログラムを書いたりコードを修正したりする必要はなくなります。

このように、生成AIを活用したプログラミングは、これまでのプログラミングとは大きく異なります。このコースを通じて、皆さんにAIを活用したプログラミング技術を学んでもらうことが目的です。

初級コースと中級コースの二つが構成されていますが、初級コースは、科研費「生成AIによるプログラミング教育のパラダイム転換と教育支援ツールの研究開発」の支援を受けて取り組んでいます。

**IP-guidance01**

プログラミング入門のガイダンスについて説明します。プログラミング入門の目的は、プログラミングの基礎概念を理解し、簡単なプログラムが作成できることです。テキストとしては、東京大学数理情報教育センターが提供しているPythonプログラミング入門を使用します。学習内容は、データ型と変数、基本的な制御構造、入出力操作、関数の定義と使用、基本的なデータ構造、ライブラリーの活用、デバッグといった内容です。

学習の進め方についてですが、Pythonプログラミング入門のテキストの解説を読み、実際にGoogle Colaboratoryを利用して練習を進めます。学習内容とテキストの対応を示した表がありますが、順番が異なる場合もあります。テキスト順に進めていくことが望ましいです。

また、三角が書かれた項目や、上記以外の内容もあります。これらは必須ではありませんが、プログラミングの習得のためには学習することを推奨します。以上でガイダンスの説明を終わります。

**IP-guidance02**

このビデオは、GitHub CEOによるGitHub Copilotを用いたアプリケーション開発のデモンストレーションです。英語ですが、日本語字幕を表示できます。日本語字幕を表示する方法をこれから説明します。リンクをクリックしてYouTubeを開きます。

ビデオの再生を一旦停止し、日本語字幕の設定を行います。まず、字幕ボタンをクリックします。この時、英語で字幕が表示されます。それを日本語に変換するために、隣にある設定ボタンをクリックします。

次に、字幕をクリックし、自動翻訳で日本語を選択します。再生してみると、以下のように日本語の字幕が表示されます。

これで日本語字幕を表示する方法についての説明を終わります。

**IP-aboutColab01**

グーグルコラボラトリーについて説明します。これ以降はコラボと呼びます。まずコラボの概要を説明し、その後、コラボの始め方を説明します。コラボでは生成AIジムニーを利用することができます。その利用方法についても説明します。

コラボは、ブラウザ上で実行可能なPythonプログラミング環境です。コラボの特徴として、環境構築が不要で、GPUに料金なしでアクセスでき、簡単に共有できる点があります。コラボはノートブックという、コードを記述して実行できるインタラクティブな環境を提供します。プログラムはコードセルと呼ばれる枠内に記述し、セルごとに実行できます。

コードは各アカウント専用の仮想マシンで実行されますが、90分間アイドル状態にある仮想マシンは削除されます。また、仮想マシンには12時間という利用制限がありますが、それを超えて使用することはできません。ただし、コラボノートブックはグーグルドライブに保存されるため、コード自体が消えることはありません。

ノートブックはIpynd形式でダウンロードすることができ、Jupyterノートブックでも利用することが可能です。参考として、グーグルが用意している「コラボへようこそ」というページに詳細な説明が書かれています。また、「よくある質問」ページのリンクも掲載されているので、そちらも適宜参照してください。

**IP-aboutColab02**

このビデオでは、コラボの始め方について説明します。コラボを使うためには、Googleアカウントが必要です。アカウントを持っていない場合は、Googleアカウントを取得しましょう。コラボにはブラウザからログインします。このURLにアクセスし、右上のログインボタンをクリックしてログインします。これはEdgeでURLを開いた場合の手順です。

ログインボタンが表示されているので、ここをクリックしてログインします。Chromeを利用する場合には、先にChromeにログインしておくこともできます。このブラウザはすでにChromeにログインしています。この状態でURLにアクセスすると、コラボが開きます。コラボへようこそをクリックすると、このようなページが表示されます。

このページでは、コラボに関する説明が書かれています。ノートブックを開く方法には三つの方法があります。まず、新しいノートブックを開く方法、Googleドライブに保存されたノートブックを開く方法、ローカルPCに保存されているノートブックを開く方法です。まず、新しいノートブックを開く方法について説明します。

メニューの「ファイル」を選び、「ドライブの新しいノートブック」を選びます。新しいノートブックが開きました。次に、Googleドライブのファイルを開く方法です。ファイルから「ノートブックを開く」を選び、「Googleドライブ」をクリックします。

保存されているノートブックの中から必要なものを選んでクリックすると、ノートブックが開きます。次に、ローカルPCにあるファイルを開く方法です。ノートブックをアップロードするには、「ノートブックをアップロードする」を選びます。参照ボタンをクリックするか、ここにファイルをドラッグします。

ここでは参照ボタンを押し、ノートブックを選んで「開く」を押します。アップロードが始まり、正常に終了するとノートブックが読み込まれます。以上がノートブックを開く方法でした。次に、ノートブックを保存する方法を説明します。

ノートブックは適当なタイミングでオートセーブされます。保存先はGoogleドライブです。ノートブックはGoogleドライブに保存することも、ローカルPCにダウンロードすることもできます。Googleドライブに保存するには、「ファイル」から「保存」を選びます。ダウンロードする場合には、「ファイル」から「ダウンロード」を選びます。

ダウンロードできる形式は、ipynbノートブック形式またはpy（Python）形式です。ここではノートブック形式で保存してみます。保存されたファイルはダウンロードファイルに保存されます。ノートブックを閉じるときには、コラボのタブを閉じます。

コードの入力と実行について説明します。コラボでは、コードやテキストを記述できるセルによって構成されています。セルは、コードボタンやテキストボタンで挿入できます。コードを記述するセルを「コードセル」、テキストを記述するセルを「テキストセル」と呼びます。コードセルにはPythonのコードを書きます。

テキストセルは、ノートブックの説明や見出しなどのノートを記述できます。テキストはマークダウンフォーマットで記述します。コードを実行するときは、実行するセルを選択して、コントロールキーとエンターキーを同時に押すか、プレイボタンを押します。これがプレイボタンです。コードが実行され、その結果がセルの下に表示されます。

実際に入力してみましょう。こちらがコードセルになります。コードセルに簡単なコードを書きます。実行ボタンを押すと、このように結果が表示されます。コードの実行が終了しない場合には、ストップボタンをクリックします。

試してみます。コードセルを追加してコードを書きます。実行してみると、このコードは継続条件が「True」となっているため、終了しないコードになります。このように、実行ボタンがぐるぐると回っている状態になります。中断するには、このプレイボタンをクリックします。終了しました。以上でコラボの始め方についての説明を終わります。

**IP-aboutColab03**

コラボでは生成AI Geminiを利用することができます。このビデオでは、生成AI Geminiの利用について説明します。Geminiを利用したコードの作成方法には、コードの補完、コードセルからの利用、チャットポットの利用の3種類があります。実際にコードを作成しながら説明を進めます。

まずはコードの補完についてです。コードの補完では、コードを入力していくと、Geminiが自動的に補完してくれます。例えば、「一から百までの和を求める」というプログラムを作ってみたいと思います。コメントにプログラムの内容を書いてエンターを押すと、提案されたコードがグレーで表示されます。Tabキーを押すとコードが確定します。実行してみましょう。

正しい計算結果が出てきました。次に、コードセルからの利用を試してみます。コードをクリックすると、「コーディングを開始するか、AIで生成します」というメッセージが表示されるので、生成をクリックします。プロンプトを入力するためのボックスが表示されます。ここで、先ほどと同じようにコメントを入力し、生成ボタンを押します。コードが生成されました。

生成されたコードは一つですが、複数のコードが提案されている場合には、この部分で生成されたコードを切り替えることができます。実行してみましょう。正しい結果が出てきました。

次に、チャットポットを利用してコードを生成していきたいと思います。ウィンドウの右上にあるGeminiのボタンをクリックします。Geminiのウィンドウが開きます。ここにプロンプトを書きます。「一から百までの和を求めるプログラムを作成してください」と入力し、送信ボタンを押すと、Geminiが処理を始め、コードを生成します。

さらに、生成されたコードについての説明も記述されます。生成されたコードはコピーするか、コードセルに挿入することができます。ここではコードセルを追加します。コードセルが追加され、実行ボタンを押すと実行されました。コラボでは、このように生成AI Geminiを使ってコードを作成することができます。

生成AI Geminiを使うと、エラーの説明やエラーの修正ができます。実際に試してみましょう。このプログラムは、整数Nを入力し、一からNまでの和を求めるプログラムです。解説のため、エラーが含まれているものになっています。実行してみましょう。エラーが表示されました。エラーメッセージを見ると、四行目にエラーがあることがわかります。

エラーメッセージには詳細が記述されています。このエラーの説明のボタンを押すと、Geminiがエラーの解説をしてくれます。エラーの解説と修正案のコード提案、それから修正案のコードの解説が表示されます。Geminiの解説は英語なので、日本語に翻訳してもらうため、プロンプトに「日本語で解説してください」と入力して送信します。

そうすると、Geminiが解説を日本語に訳してくれます。生成されたコードはコピーするか、コードセルに追加することができます。コードセルに追加してみましょう。追加されたコードセルを実行してみます。

今度は正しく実行されました。次に、エラー修正を試してみます。エラー修正は、文法的な誤りがある場合に利用できます。新しいコードを追加し、先ほどのコードをコピーしました。その後、このコロンを削除して文法的に誤りのあるコードにします。実行してみましょう。

エラーメッセージが表示され、エラーの修正ボタンが表示されます。エラーの修正をクリックすると、修正案が表示されます。オレンジの部分はコードの最初にマイナスがついている部分で、これは削除される部分です。グリーンの部分はコードの先頭にプラスがついている部分で、これは追加されるコードです。

修正案を承認する場合にはチェックボタンを押し、拒否する場合にはバツボタンを押します。ここではチェックボタンを押して承認します。実行してみましょう。正しく実行されました。以上で生成AI Geminiの利用についての説明を終わります。

**IP-text01**

このビデオでは、Pythonプログラミング入門コラボ版の利用方法について説明します。まず、教材のオープンについてですが、教材のインデックスページで開いたノートブックには編集権限がありません。実行した結果を保存できないので、ドライブにコピーしたものを使って演習を進めます。コピーの手順はここに書いています。

実際に教材をオープンする手順を見てみましょう。テキストのページから「Pythonプログラミング入門コラボ版」をクリックし、リンクをクリックします。すると、このように目次が表示されます。コラボの使い方については別のコンテンツですでに説明しています。

コラボラトリーの使い方では、コラボの立ち上げからノートブックのダウンロードまでが説明されています。適宜参考にしてください。それでは、1-1数値演算のセクションを始めます。セクションのリンクをクリックすると、ノートブックが読み込まれます。ここで開かれたファイルは編集ができませんので、ドライブにコピーします。

コピーを作成し、正常に開くと、このようにノートブックが開かれます。最初に目次から開いたタブは閉じましょう。ノートブックの名前を変更するには、ノートブック名の部分をクリックし、拡張子以降を削除します。

これで演習の準備が整いました。一度学習したセクションを再度学習する場合は、Googleドライブにコピーしたノートブックを開きます。そうすると、前回の学習の続きから始めることができます。

次に、教材の利用について説明します。テキストの説明に従いながら、コードセルへの入力や実行を確認していきましょう。どうしてそのような実行結果になるのかを考えながら確認していきます。エラーが生じる場合もありますので、どうしてエラーとなったのかを考えましょう。

エラーの解説に生成AIを利用することも有効です。コードセルやテキストセルは適宜追加して構いません。コードを修正したり、新しいコードを入力して試してみましょう。また、気づいた点などをテキストセルに残しておくと復習に役立ちます。

実際に確認してみましょう。ノートブックには説明の内容や参考となるリンクが挙げられています。適宜、これらのリンクも確認していきましょう。続いて説明を読んでいきます。説明の中には、このように実際にセルに入力して実行を試すものもあります。

やってみましょう。ここでは「一足す一」と入力し、「Shift」を押しながら「Enter」を押すように指示がありますので、その通りにしてみます。プログラムが実行され、結果が表示されました。また、あらかじめコードセルの中にコードが入っているものもあります。このような場合には、実行ボタンをクリックして実行結果を確認していきましょう。

また、すでに書かれているコードを修正することも可能です。いろいろなコードを試して確認してみましょう。テキストセルも編集が可能です。ダブルクリックをすると編集ができます。ここで入力をすることができます。

さらに、テキストセルも追加できます。テキストセルを有効に使って学習内容のメモをしていきましょう。このような演習問題もありますので、取り組んでみてください。終了する際は、ノートブックを保存し、タブを閉じます。実際に確認してみましょう。

ノートブックの保存は、Ctrl + Sもしくは「ファイル」から「保存」をクリックします。保存が終わると、「すべての変更が保存されました」と表示されます。この表示が出たことを確認してタブを閉じましょう。以上でPythonプログラミング入門コラボ版の利用方法の説明を終わります。

**IP-text02**

このビデオでは、Pythonプログラミング入門コラボ版のテキストについて説明します。Pythonプログラミング入門コラボ版に移動するためには、このリンクをクリックしてください。リンクをクリックしてみます。練習を進める際は、ノートブックをドライブにコピーしてから進めることをおすすめします。

Pythonプログラミング入門の利用方法に関するコンテンツには、コピーの仕方やコンテンツの利用方法が説明されていますので、そちらも確認してください。

**IP-text03**

このビデオでは、Pythonプログラミング入門PDF版のテキストについて説明します。Pythonプログラミング入門のPDF版のテキストになっています。内容についてはコラボ版のものと同一です。

内容の復習や実際にプログラミングをして分からなかったところを調べるときなどに利用すると良いと思います。