『Pythonで学ぶプログラミング入門』

# 単元の目標

|  |  |
| --- | --- |
| 知識・技能 | アルゴリズムを表現する手段、プログラミングによってコンピュータや情報通信ネットワークを活用する方法について理解し技能を身に付けているようにする。   * プログラム言語の順次・分岐の制御構造と、変数を用いた簡単なプログラムを作成できるようにする。 * リストと繰り返しを用いた簡単なプログラムを作成できるようにする。 * 関数を用いた簡単なプログラムを作成できるようにする。 |
| 思考・判断・表現 | 目的に応じたアルゴリズムを考え適切な方法で表現し、プログラムによりコンピュータや情報通信ネットワークを活用するとともに、その過程を評価し改善することができるようにする。   * 順次、分岐・選択と繰り返しを組み合わせたアルゴリズムを、プログラムとして表現することができるようにする。 * 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができるようにする。 |
| 主体的に学習に取り組む態度 | 問題解決にコンピュータを積極的に活用し、自ら結果を振り返って改善しようとしている。   * （※他単元を含め）本単元は、知識・技能、および思考・判断・表現の習得に集中しているため、態度の評価は他単元の様子と合わせて記録・評価するとよい。 |

# 単元の評価基準

|  |  |
| --- | --- |
| 知識・技能 | 1. プログラム言語の順次・分岐の制御構造と、変数を用いた簡単なプログラムを作成できる。 2. リストと繰り返しを用いた簡単なプログラムを作成できる。 3. 関数を用いた簡単なプログラムを作成できる。 |
| 思考・判断・表現 | 1. 順次、分岐・選択と繰り返しを組み合わせたアルゴリズムを、プログラムとして表現することができる。 2. 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができる。 |
| 態度 | 1. （※他単元を含め）アルゴリズムを考え、問題を解決しようとしている。 |

# 指導と評価の計画

## 小単元等

|  |  |
| --- | --- |
| 小単元等 | 授業時間数 |
| プログラミング入門 | 6時間 |
| プログラミング応用（アルゴリズム） | 2時間 |

## 時間ごとのねらい・学習活動、重点、記録

本単元ではプログラミングを学ぶ。評価のための記録は、プログラムファイルを主とする。テキストの章ごとに、章末に問題集があるので、これを課題とする。

プログラムファイルの記録・保存にあたっては、各校の事情、利用しているサービスを考慮して、ファイルの提出や、オンラインノート/LMSなどを利用するものとする。

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 時間 | ねらい・学習活動 | 重点 | 記録 | 備考 |
| 1 | ○順次実行と変数   * プログラム言語の順次の制御構造と、変数を用いた簡単なプログラムを作成できるようにする。 | 知 |  |  |
| 2 | ○条件分岐による選択（分岐）   * プログラム言語の条件分岐の制御構造を用いた簡単なプログラムを作成できるようにする。 | 知 | ○ | 知①:プログラムファイル |
| 3 | ○リスト   * リストを用いた簡単なプログラムを作成できるようにする。 | 知 |  |  |
| 4 | ○繰り返し（反復）   * 繰り返しを用いた簡単なプログラムを作成できるようにする。 | 知 | ○ | 知②:プログラムファイル |
| 5 | ○関数の定義と利用   * 関数を用いた簡単なプログラムを作成できるようにする。 | 知 | ○ | 知③:プログラムファイル |
| 6 | ○繰り返し（反復）と選択（分岐）の組み合わせ   * 順次、分岐・選択と繰り返しを組み合わせたアルゴリズムを、プログラムとして表現することができるようにする。 | 思 | ○ | 思①:プログラムファイル |
| 7,8 | ○アルゴリズムの実装   * 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができるようにする。 | 思 | ○ | 思②:プログラムファイル |

# 観点別学習状況の指導と評価

## 各時の指導と評価

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 1 | 目標:   * プログラム言語の順次の制御構造と、変数を用いた簡単なプログラムを作成できるようにする。   評価基準:   * プログラム言語の順次の制御構造と、変数を用いた簡単なプログラムを作成できる。   展開:  本時はプログラムを作成・実行するためのツールを初めて使う機会である。ツールやアカウントを準備し、ログインやサンプルプログラムの取り込みなどの準備を行う。  学習事項としては、制御構造としての順次実行、および変数である。サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見る。  簡単なプログラムの作成・実行を通じて、ツールに慣れるとともに、プログラミングの基本的な注意点（大文字・小文字の区別や、文字列の扱い方・数値の扱い方など）に触れることとする。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. プログラミングを学ぶことを伝える。使用するツールについて概要を紹介する（起動方法、ログイン方法、サンプルプログラムの読み込み方法など）。 2. ツールの利用を開始する。起動し、ログインする。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。  * print() * 順次実行 * 文字列の値 * 変数 * 数値 * 算術演算 * 入力の受付 input() * 値の型の変換（文字列から数値） int()  1. まとめ: 本時の内容を振り返る。  * ツールの機能や利用方法（起動、ログイン） * プログラミング   + 順次実行   + 変数   + 値（文字列型の値、数値型の値、int()）   + 入力（input()）、出力（print()） | ツールの機能や使用方法を網羅的に説明すると、分量が多く混乱のもとになる。すぐに利用する機能を少しずつ補足的に説明するとよい。  ※Monaca Educationを利用する場合、ブロック崩しプロジェクトが利用できる。  概念説明に偏らないようにする。サンプルプログラムをインポートして、試行・変更・試行を繰り返しながら説明する。  ※サンプルプログラムを全て手入力すると、不慣れな生徒がいる場合にタイピングで時間がかかってしまう。タイピングの練習と、プログラミングの概念の習得が混在すると理解が困難になるため、キーボード入力を減らす工夫をすること。 |   指導と評価の方法:  本時は、プログラミングやツールに初めて触れる時間である。単元を通じた評価のための、補足的な情報を収集するものとする。  特に、思考・判断・表現、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * ツールを起動し、ログインできる。 * サンプルプログラムを読み込むことができる。 * プログラムを変更・保存し、動作を確認できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 他の制御構造（条件分岐、繰り返し）や、文字列型・整数型以外の型について可能性を考えることができる。 | | 態度 | 「おおむね満足できる」状況（B） | * 細かな入力ミス（スペルミス、二重引用符の一方が欠ける、カッコが閉じていない、など）で諦めることなく、粘り強くプログラミングをしている。 | | 「十分満足できる」状況（A） | * 上記のようなミスをした後、自分の経験を記録・理解して、次のプログラミングで活かしている。 | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 2 | 目標:   * プログラム言語の条件分岐の制御構造を用いた簡単なプログラムを作成できるようにする。   評価基準:   * プログラム言語の条件分岐の制御構造を用いた簡単なプログラムを作成できる。   展開:  本時の授業では、サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見ながら、条件分岐による選択の制御構造を学ぶ。  プログラミングを行う時間としては2時間目であることを鑑み、ツールの使用法や、プログラミングの際の細かな留意点について、折に触れて振り返るとよい。また、テキストでは条件分岐の３つのパターンを扱っている。フローチャートで制御の流れを追いながら授業を進める。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. 引き続き、プログラミングを学ぶことを伝える。毎回同じ処理を同じ順番で実行させるだけでなく、条件に応じて、違う処理をさせられることを伝える。身近なアプリ、ゲームの例を出すとよい（※実行する時間によって、挨拶のメッセージが違う例など）。 2. ツールの利用を開始する。起動し、ログインする。簡単に前時の振り返りを行う。順次実行、変数、入力と出力など。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。 4. 問題集に取り組む。  * ここまでの学習内容の確認のために、第1章・第2章の問題集に取り組む。  1. まとめ: 本時の内容を振り返る。  * 制御構造: 条件分岐   + if文   + ブロック、字下げ（インデント）   + if - else   + if - elif - else * random.randint() | 概念説明に偏らないようにする。サンプルプログラムをインポートして、試行・変更・試行…を繰り返しながら説明する。   * 知①:プログラムファイル |   指導と評価の方法:  思考・判断・表現、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * 制御構造if, if-elseを使ったプログラムを作成できる。 | | 「十分満足できる」状況（A） | * 制御構造if-else if-elseを使ったプログラムを作成できる。 * 乱数を利用するプログラムを作成できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 制御構造の入れ子（例: if文の中にif文）や、複雑な条件（例: 0以上2未満など、論理演算が必要なもの）について考えることができる。 | | 態度 | 「十分満足できる」状況（A） | * 身近なプログラムの中に、条件分岐の動作を見出すことができる。 | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 3 | 目標:   * リストを用いた簡単なプログラムを作成できるようにする。   評価基準:   * リストを用いた簡単なプログラムを作成できる。   展開:  本時の授業では、リストについて学ぶ。サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見ながら、リストの宣言、操作を習得する。  次の時間の「繰り返し」と関係の深い授業となる。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. 引き続き、プログラミングを学ぶことを伝える。一つのリスト型の変数に、複数の値を格納して利用することができることを伝える。学校と教室、アパートやマンションの部屋など、全体に名前があり、部分を指す番号が付いている構造を例示するとよい。 2. ツールの利用を開始する。起動し、ログインする。簡単に前時の振り返り（if文による条件分岐）を行う。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。 4. まとめ: 本時の内容を振り返る。  * リスト   + リスト要素への添え字を使ってアクセスする   + 要素の置き換え   + 空のリスト   + リストへの要素の追加（末尾への追加、途中への追加）   + リストの要素の削除 | 概念説明に偏らないようにする。サンプルプログラムをインポートして、試行・変更・試行…を繰り返しながら説明する。 |   指導と評価の方法:  思考・判断・表現、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * リストを宣言して、添え字を用いて値を利用・変更できる。 | | 十分満足できる」状況（A） | * 要素の追加や削除など、自由に操作できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * リストのリスト（二次元リスト、三次元リスト）などを考えることができる。 | | 態度 | 「十分満足できる」状況（A） | * 身近なデータの中に、リストのような構造を見出すことができる。 | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 4 | 目標:   * 繰り返しを用いた簡単なプログラムを作成できるようにする。   評価基準:   * 繰り返しを用いた簡単なプログラムを作成できる。   展開:  本時の授業では、引き続きプログラミングを学ぶ。サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見ながら、for文とwhile文のそれぞれについて、基本的な記述方法を習得する。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. 同じ処理を繰り返し実行したい場合の制御を学ぶことを伝える。前時のリストと組み合わせる場合が多いことを示唆するとよい。 2. ツールの利用を開始する。起動し、ログインする。簡単に前時の振り返り（リストの操作）を行う。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。 4. 問題集に取り組む（第3章、第4章問題集の各問題） 5. まとめ: 本時の内容を振り返る。  * for in 文 * range() * while文   + 条件式 | 概念説明に偏らないようにする。サンプルプログラムをインポートして、試行・変更・試行…を繰り返しながら説明する。   * 知②:プログラムファイル |   指導と評価の方法:  思考・判断・表現、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * for文を用いて、指定した回数繰り返す処理を行うプログラムを作成できる。 | | 「十分満足できる」状況（A） | * while文を用いて、指定した条件を満たす間繰り返す処理を行うプログラムを作成できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 順次・繰り返し・条件分岐の制御構造を組み合わせた処理を考えることができる。 | | 態度 | 「十分満足できる」状況（A） | * 身近なアプリやサービスの中に、繰り返しの制御を見出そうとしている。 | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 5 | 目標:   * 関数を用いた簡単なプログラムを作成できるようにする。   評価基準:   * 関数を用いた簡単なプログラムを作成できる。   展開:  本時の授業では、引き続きプログラミングを学ぶ。関数を定義し、利用する。引数を渡す、戻り値を受け取る関数を作り、利用する。サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見ながら、関数・引数・戻り値を使うことで得られるメリットを理解する。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. 「関数」の概念を学ぶことを伝える。プログラムのさまざまな場所で、同じ処理をさせたい場合に、それぞれの場所に同じ処理を書くのではなく、１箇所に書いた関数を呼び出すようにする。テキストでは、関数を自作する前に、プログラム言語に組み込まれている「組み込み関数」を説明している。概念的な説明を飛ばし、「画面に表示する」「ユーザーからの入力を受け付ける」などこれまでの授業で利用してきた組み込み関数の例を確認し、「皆が使う機能は、一つ書いておき、皆で使う」という考え方を確認するとよい。 2. ツールの利用を開始する。起動し、ログインする。簡単に前時の振り返り（繰り返しの制御）を行う。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。 4. 問題集に取り組む（第5章問題集の各問題） 5. まとめ: 本時の内容を振り返る。  * 組み込み関数   + input()   + print()   + random.randint()   + range()   + dateの各機能 * 関数の定義   + defキーワード   + 関数名   + 引数   + 戻り値     - return * 関数の利用   + 引数を渡す   + 戻り値を利用する | 概念説明に偏らないようにする。サンプルプログラムをインポートして、試行・変更・試行…を繰り返しながら説明する。   * 知③:プログラムファイル |   指導と評価の方法:  思考・判断・表現、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * 組み込み関数を利用するプログラムを書ける。 * 関数を定義して利用するプログラムを書ける。 | | 「十分満足できる」状況（A） | * 引数のある関数、戻り値のある関数のプログラムを書き、適切に利用できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 引数のある関数、戻り値のある関数それぞれについて、どんな場合に適切か考えて説明できる。 | | 態度 | 「十分満足できる」状況（A） | ※特に無し | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 6 | 目標:   * 順次、分岐・選択と繰り返しを組み合わせたアルゴリズムを、プログラムとして表現することができるようにする。   評価基準:   * 順次、分岐・選択と繰り返しを組み合わせたアルゴリズムを、プログラムとして表現することができる。   展開:  本時の授業では、引き続きプログラミングを学ぶ。これまでに学んだ条件分岐と繰り返しを組み合わせるプログラム、繰り返しを二重にするプログラムを見る。サンプルプログラムを動作させ、一部修正を加えて動作が変わる様子を見ながら、これまでの学習を振り返り、学習した事項を組み合わせたプログラムを作成する。作成したプログラムを動作させ、想定通りの結果が得られることを確認する。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. やや複雑な問題を、これまでに学んだプログラミングの技法を組み合わせて解くことが本時の課題であることを伝える。 2. ツールの利用を開始する。起動し、ログインする。簡単に前時の振り返り（関数）を行う。 3. テキストの記載に従う。説明・サンプルプログラムの実行・サンプルプログラムの内容確認・例題を組み合わせて進める。    * 繰り返しと条件分岐の繰り返し（１から１０まで、「偶数」・「奇数」を表示する）    * 九九の表示    * 繰り返しの中断（break） 4. 問題集に取り組む（第6章問題集の各問題） 5. まとめ: 本時の内容を振り返る。 | 条件分岐、繰り返しなどの書き方（文法事項）は、テキストの前のページを見てよいことを伝える。プログラミングは暗記で解くものではないことを伝える   * 思②:プログラムファイル |   指導と評価の方法:  知識・技能、態度については、追加的な評価の情報とする。   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「十分満足できる」状況（A） | * 繰り返しと条件分岐を組み合わせたプログラムを作成できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 九九を後ろから表示するプログラムなど、やや複雑な制御を考えて実現できる。 | | 態度 | 「おおむね満足できる」状況（B） | ※特に無し | |

|  |  |
| --- | --- |
| 時間 | 指導と評価 |
| 7,  8 | 目標:   * 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができるようにする。   評価基準:   * 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができる。   展開:  引き続きプログラミングを学ぶ。これまでの時間で学んだ内容を用いて、整列（ソート）または探索（サーチ）アルゴリズムをプログラムとして表現する、実習の時間とする。作成したプログラムを動作させ、想定通りの結果が得られることを確認する。  アルゴリズムの概念的な理解のために、（プログラミング、ツールを離れて）探索アルゴリズム、整列アルゴリズムを学ぶ時間を挿入することも考えられるが、プログラミングの習熟が進んでいれば、概念的な説明よりも、実習による学習が有効である可能性は高い。   |  |  | | --- | --- | | 学習活動 | 評価と配慮事項 | | 1. 整列（ソート）または探索（サーチ）の問題を解くアルゴリズムを簡単に確認したあと、開発するアルゴリズムを決める。 2. プログラムを開発する実習時間とする。 | 思②: プログラムファイル |   指導と評価の方法:   |  |  |  | | --- | --- | --- | |  | 評価 | 評価の視点 | | 知識・技能 | 「おおむね満足できる」状況（B） | * 条件分岐、繰り返しの制御構造を持つプログラムを作成できる。 | | 思考・判断・表現 | 「十分満足できる」状況（A） | * 整列（ソート）または探索（サーチ）について、アルゴリズムをプログラムとして表現することができる。 | | 態度 | 「おおむね満足できる」状況（B） | * 日常的な問題について関心を持って考え、アルゴリズムを検討することができる。 * さまざまなデータを入力して、アルゴリズムが機能することを確認している。 | |