|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | **第１章　Pythonを見てみよう**  １　Python とは  ２　Python を使う | | | | | プログラミングとは何か，Pythonとは何かということを理解させる。  　Pythonのインストール手順を理解させる。32ビット版と64ビット版の両方があること，今回は64ビット版を使うことを伝える。  　IDLE，シェルウィンドウ，エディタウィンドウとは何なのかを理解させる。  　実際にシェルウィンドウで「2+3」などのいろいろな数値計算を試させ，機能を確認させる。括弧を複数使用した数式や「\*\*」を使った累乗の計算など。たとえば「2\*\*0.5」が「1.41･･･」となることなどを示す。  　IDLEの利用方法を理解させる。print()を例として，プログラムを作成・保存・実行する手順に慣れさせる。  　print()で表示するときの数値と文字列の違い，そして文字列のときにシングルクォーテーション，ダブルクォーテーションが必要なことを理解させる。  　プログラム中の文の区切りとインデントなど，Pythonでの基本的なプログラムの記述方法を理解させる。  　#（ハッシュ）を使ったコメントの挿入の仕方とコメントの意味，使い方を理解させる。 | | | | | | | 0.5  0.5 | | 他のプログラミング言語と比較したPythonの特徴「コードが読みやすい」「ライブラリが豊富」「動作が簡単に確認できる」「科学技術分野，システム管理，Webサービス，アプリケーション開発など，広く利用されている」などにも触れるとよい。  　教員の画面を生徒に示す場合，初期設定だとコードの文字が小さく，生徒は見づらいだろう。IDLEでの文字サイズ変更の方法を生徒に教えるとともに，教員が提示する文字サイズもあわせて変更しておくとよい。  Pythonのプログラム自体は単純なテキストであり，IDLE以外のエディタでも使えることを理解させたい。できれば，メモ帳などの他のテキストエディタでも作成させ，文字エンコーディングをUTF-8で保存すること，そして拡張子を「.py」とすることも理解させたい。  　コメントで使う#（ハッシュ）はシャープではないことに，生徒は興味を示すだろう。時間があれば，「'''」によるコメントの挿入方法や，実行したくない行をコメントにする方法なども教えたい。  　「a=2」などの変数の使い方，「if a>0:」などのif文などは，今後学んでいくことを伝え，ここではPythonでは「インデントを使って処理のまとまりを示すこと」「インデントが重要な意味を持つこと」を確実に理解させたい。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | **第２章　プログラムをつくってみよう**  １　変数を使ってみよう  ２　入力されたデータを処理してみよう | | | | | 「変数はデータの入れ物」という概念，そしてPythonでの変数名の付け方や変数の使い方を理解させる。  　プログラミングの世界では「=」は代入を意味することを強調する。  　算術演算子を例にして「演算」と「演算子」の意味を理解させる。シェルウィンドウを使ってすべての算術演算子の働きを試させる。  　=が代入演算子であること，そしてさまざまな累算代入演算子の働きを理解させる。  　データ型の意味を理解させる。たとえば数字を使ってstr（文字型）とint（整数型）の違いを説明する。  　リストの意味を理解させ，実際にいくつかの例をprint()で試させる。時間があればタプル型や辞書型があることも教える。  　input()の働きを理解させ，実際にシェルウィンドウで動作を確認させる。  　文字列の結合と繰り返しの方法を理解させる。input()を使って「入力した文字列を結合する」あるいは「入力した文字列を繰り返す」例を教えた後，生徒に文字列を操作するプログラムを自由に作らせ体験させる。  　データ型の変換について理解させる。数値と文字列の違いについては詳しく説明する。  　「年齢当て」プログラムを作成させ，なぜこのプログラムで年齢を計算できるのかを考えさせる。10の位の数字をx， 1の位の数字をyとする。これらの和をa，差をbとすると，「x+y=a」「x-y=b」の連立方程式の解は「x=0.5(a+b)」「y=0.5(a-b)」となる。元の数は10x+y。したがって10x+y=0.5(11\*a+9\*b)となり，｢5.5\*a+4.5\*b」で求めることができる。 | | | | | | | 0.5  0.5 | | 変数の概念は丁寧に説明したい。またPythonでは「大文字と小文字が区別される」ことは強調したい。  　=が代入であることを感覚的に理解しづらいプログラミング初心者は多い。十分に注意して教えていただきたい。  　生徒にはPythonでは変数の宣言をすることなくデータ型が決まることを意識させたい。  　リストのインデックス番号は0からはじまることを強調したい。実際にリストを作成し，print()で出力させて試させると良い。  　時間があれば，辞書型のデータを作成し，print()で出力させ，試させると良い。  　input()で入力した文字列を処理してprint()で表示させる例は，「input()→print()」を何回も繰り返すプログラムを作成するとよいだろう。ＡＩ会話風になり，生徒は興味を示す。生徒が作成したプログラムを相互評価させるとより効果的だ。  　「年齢当て」プログラムでは，なぜ「5.5\*a+4.5\*b」で元の数値（年齢）が表示されるのかをじっくりと考えさせたい。連立方程式を示し，別の表現でもよいことも伝えるとよい。  　また，小数点以下を表示させないためにint(5.5\*a+4.5\*b)となっている。このintの意味を生徒に考えさせるとよい。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | ３　条件によって処理を変えてみよう  ４　処理を繰り返してみよう | | | | | 「if」「if～else」「if～elif～else」の条件分岐の使い方を理解させる。  　andやor，notの論理演算子を使うことで複数の条件が設定できることを理解させる。  　input()とifを使って，対話的に点数によって処理を3つに分けるプログラムを作成させる。質問内容や閾値を変更して，いろいろな応用プログラムを作らせ，生徒の興味を引き出す。  　時間があればandやor，notの論理演算子を使った対話的なプログラムを自由に作らせ，相互評価をさせる。  　リストの各要素に対して繰り返し処理を行う「for～」の使い方を理解させ，実際に試させる。文字列のそれぞれの文字に対して繰り返し処理を行う「for～in～」の使い方を理解させる。  　イテラブルオブジェクトとは何なのかということを理解させる。  　「for～in　range()」の使い方を理解させる。range(start,stop,step)の形式で，print()を使って数値を表示するさまざまな例を体験させる。  　「while ～」の使い方を理解させる。この場合,条件の設定を誤ると無限ループになってしまうことを理解させる。 | | | | | | | 0.5  0.5 | | Pythonではif文などのブロックを，インデントを使って定義することを強調したい。またIDLEを使うと「if」「elif」「else」などのキーワードが赤色など別の色で表示されることを意識させたい。  　生徒は，「if」「elif」「else」の行の最後に「:」を付け忘れることが多い。忘れないように強調しよう。  　「値が等しい」の場合，「=」ではなく，「==」とすることに注意させたい。input()とifを使うと，AIと会話をしているようなプログラムも作成可能だ。答えが限定されるような質問と，その質問に対する興味深い応答を生徒に考えさせるとよいだろう。  　Pythonのイテラブルオブジェクトは生徒にとって理解しづらい概念だろう。生徒は，「for 変数 in～」で使う変数の意味が理解しづらいものである。イテラブルオブジェクトの各要素が次々と変数に代入されるイメージを，具体例を挙げて説明するとよいだろう。  　range(5)とした場合「0～4」の値がカウンタ変数に渡されることを強調したい。「0～5」と勘違いする生徒は多いと思われる。  　whileで繰り返し処理の条件設定は重要だ。生徒には，条件式を慎重に検討させたい。また「より大きい（>）」と「以上（>=）」，そして「より小さい(未満)（<）」と「以下（<=）」の違いを意識させたい。条件の設定によっては無限ループになってしまうことに十分に注意させたい。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | 改訂版 プログラミング入門Python編 | | |
|  | 選択 | 発行所 | | 数研出版 | | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | ５　任意の関数をつくってみよう  ６　標準ライブラリを使ってみよう | | | | | 関数とは何なのか，そして任意の関数を作ることによるメリットを理解させる。print()やint()などすでに学んだ関数を思い出させ，組み込み関数の意味を理解させる。  　任意の関数を作る「def　関数名（引数）:（インデント）処理（インデント）return 戻り値」の書き方と，関数のよびだし方法を理解させる。  　「分」を「秒」に変換する関数を使ったプログラムを作らせ，その使い方を理解させる。いろいろな値を引数に設定して，その働きを実感させる。  　グローバル変数とローカル変数の違い，スコープを理解させ，globalを使ったプログラムを体験させる。  　「標準ライブラリ」「モジュール」のそれぞれの意味を理解させる。  　標準ライブラリにはmath，random，tkinterなどさまざまなものがあり，目的に応じて有効に活用できることを理解させる。  　randomモジュールを例にしてimportの方法を理解させる。実際に乱数を発生するプログラムを作成・実行させる。P17の練習問題など，いろいろな例を体験させる。  　これからtkinterを利用していくこと，tkinterにはさまざまなメリットがあることを理解させる。  　tkinterを使ってウィンドウを表示し，ウィンドウのサイズを指定する方法を理解させる。本書ではtkinterにtkという別名を付けて利用することも意識させる。 | | | | | | | 0.5  0.5 | | 生徒にとっては，「関数」の概念を理解するのは難しいことだ。とりあえず任意の関数の利用方法とメリットを理解させることが重要だろう。その後で，生徒の理解度に応じて，すでに学んだ組み込み関数との関係や，組み込み関数の意味に触れるとよいだろう。  　グローバル変数とローカル変数の扱いは十分に注意して教えたい。特にPythonのグローバル変数の扱いには癖がある。「関数内で，globalで宣言しないで代入するとローカル変数として扱われる」ということは強調したい。  　Pythonではさまざまなライブラリがあり，これらを使うことで多様な機能が実現することを強調したい。  　生徒には本書P32や，Python公式サイトにある「Python標準ライブラリ」を確認させるとよい。たとえばシェルウィンドウで，  「import calendar」  「print(calendar.month(2025,5))」の2行を入力すると，2025年5月のカレンダーがすぐに作れることを体験させるとよい。  　random()は「0以上1未満の小数」，randint(0,10)では「0以上10以下の整数」を出力することに注意させたい。  　tkinterでは実際に生徒にプログラムを作らせ，体験させることが重要である。「root」「mainloop」などの意味は，具体的な操作を通して教えるとよいだろう。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | ７　図形を描いてみよう  ８　ウィジェットを使ってみよう | | | | | tkinterを使って，キャンバスを配置し，図形を描画する方法を理解させる。  　線create\_line()，円create\_oval()，長方形create\_rectangle()を使ったプログラムを作らせ，任意の「お絵描き」を体験させる。できれば相互評価をさせる。  　図形に対する色の設定方法を理解させる。色の設定方法には，カラーネームを使った方法とRGBカラーコードを使った方法があることを理解させる。実際に任意の「お絵描き」で，さまざまな色をつけさせ体感させる。  　tkinterのウィジェットとは何か，どのような種類があるのか，どのようにして利用するのかを理解させる。  　オプションを付けてButtonウィジェットを設定する方法を理解させる。ラベルのテキストを任意の文字列に変更して表示させる。  　オプションを付けてEntryウィジェットを設定する方法を理解させる。時間があれば，tkinterに共通するfontの設定方法を理解させ，ボタンに表示させるテキストのサイズを変更させる。  　Buttonをクリックすると文字が表示される仕組みを理解させ，実際にプログラムを作成・実行させる。  　tkinterでの画像の設定方法を理解させ，Buttonをクリックすると画像が表示されるプログラムを作成・実行させる。 | | | | | | | 0.5  0.5 | | 任意の「お絵描き」プログラミングには生徒は大いに興味を示す。Pythonを学ばせるためのよい題材といえる。時間があれば生徒に任意のイラストを描かせ相互評価をさせると，効果的な授業が展開できる。  　ただし生徒は複雑な「お絵描き」に挑戦し，時間がなくなり中途半端に終わってしまうことも多い。作業時間を決めて，効率よくプログラミングする技術も学ばせたい。  　色の指定方法は，他のさまざまなプログラミング言語にも応用できる。「red」「green」「blue」などのカラーネーム,そして「#ｆｆ0000」などの16進法によるカラーコードの指定方法に慣れさせたい。  　tkinterでのウィジェット設定方法は独特ともいえる。この手順に，十分に慣れさせたい。  　できればtkinterでのfontの設定方法も教えたい。fontオプションにはさまざまなものがあるが，生徒が理解しやすく，またよく使うのはサイズだろう。サイズの変更方法だけは確実にできるようにさせるとよい。  　ここで，tkinterでのPhotoImageを使った画像の設定方法に慣れさせたい。あらかじめさまざまな画像ファイルを用意し，複数のボタンを設定させよう。任意の画像を表示する仕組みを作らせると，生徒の興味を引き出すことができる。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | ９　いろいろなイベントを使ってみよう  １０　図形を動かしてみよう  １１　ファイルを読み込んでみよう | | | | | tkinterで利用できるさまざまなイベントを例として，プログラミング言語におけるイベントの意味と一般的な利用方法を理解させる。  　ウィジェットにマウスイベントを関連づけるプログラムを作成・実行させ，bind()の使い方とその働きを理解させる。  　キーボ-ドイベントを使ったプログラムを作成・実行させ，その意味を理解させる。  　時間があれば，キーボ-ドイベントを使った任意のプログラムを自由に作成させ，相互評価させる。たとえば「"a"のキーをタイプしたら～が実行される」のように，ifと組み合わせた応用的なプログラムを考えさせる。  　after()を使ったタイマー処理の方法を，「図形を動かす」ことを例にして理解させる。  　図形をキャンバスと同じ色で塗ることで消し，新たな座標に図形を描く･･･この繰り返しによって図形を動かす方法を理解させる。  　図形を動かす具体的なプログラムを作成・実行させ，スピードや方向を変更させ，その結果を確認させる。  　Pythonでファイルを読み込んで利用する手順を理解させる。  　辞書型データの構造を理解させ，具体的なデータを作成させる。  　「dictionary[key]=value」が，キー（key）に対するバリュー（value）を設定していることを理解させる。 | | | | | | | 0.5  0.5  0.5 | | イベントの概念は，生徒には理解しづらいものである。クリックやキーボードのタイプに反応するプログラムを，実際の動作を確認させながらじっくりと学ばせたい。  　「Keytype(event)」，  「event.keysym」という書き方は初心者には難しい。実際のプログラムと対比して確実に指導したい。  　キーボードイベントを使ったプログラムはさまざまな機能に応用することができる。生徒は，キーボードイベントとifを組み合わせた任意のプログラムを作り，その機能を体験することに強い興味を示す。  　Pythonで図形を動かす方法はいくつかある。今回はキャンバスと同じ色を塗って消した後，新たな座標に図形を描くことで図形を動かしている。  　生徒は「動き」に強い興味を示すことが予想される。さまざまな要素を変更したプログラムを作成させ，after()の使い方や各変数と座標，スピードの関係を理解させたい。  　ファイルのopen()とclose()，そしてread()という，プログラミングでのファイル操作の基本を教えたい。  　生徒はread()，splitlines()が具体的に何を処理しているのか理解しづらいだろう。具体的な処理結果を示し，丁寧に説明したい。  　辞書型データでは「キー」と「バリュー」の関係を理解するのが難しい。具体例を挙げて丁寧に説明したい。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | | |
| **教務** |  |  | **第３章　ゲームをつくってみよう**  １　タイピングゲームをつくる | | | | | ランダムに落ちてくるアルファベットを，キーボードでタイプして撃ち落とすゲーム「タイピングゲーム」の仕組みを理解させ，作成・実行させる。  　このプログラム全体の流れを理解させる。また，繰り返し処理とイベント処理の役割を理解させる。  　「モジュールのインポート」「変数の宣言」「ゲーム画面の初期設定」の内容を理解させ，入力させる。  　キーボード入力を受け取る関数Keytype(event)，ゲームをコントロールする関数ControlGame()，ターゲット文字をつくる関数CreateTarget()，ターゲット文字を動かす関数MoveTarget()，ゲームオーバーとなってからゲームを再開するまでの処理を行う関数WaitNextGame()の各関数の意味を理解させ，入力させる。  　プログラムの最後に，ゲーム全体の処理に関する３行「ControlGame()」，  「root.bind("<Key>",Keytype)」，  「root.mainloop()」を入力させる。プログラム全体の構造を理解させる。  　プログラムを実行させ，問題が発生したらその原因を考えさせ修正させる。  　リストtargetlistは，画面に表示される文字（ターゲット文字）の情報を保管していること，そして1つのターゲットをtargetで表した場合，target[0]が「表示される文字」，target[1]が「ラベルのオブジェクト」，target[2]が「x座標」，target[3]が「y座標」を表していることを理解させる。  　時間があれば，文字の落下速度や文字の種類を任意のものに変更させるなど，プログラムを自由に改良させる。 | | | | | | | 1.0 | | 生徒には変数counttimeの役割がわかりにくいかもしれない。変数counttimeは，移動処理を実行した回数を示す。このcounttimeが１ずつ増え，「counttime%5==0」で5の倍数になったときにターゲット文字を作成していることを理解させたい。  　target[0]は，  moji[random.randint(0,25)]により，a～zから乱数で選んだ1文字となる。target[1]は，  tk.Label(text=targetmoji, font=("",20),bg="white")で生成されるラベルのオブジェクトである。target[2]は，  x=random.randint(0,470)により生成される。0～470の範囲の整数のx座標から，乱数によってそのうちの１つが選ばれる。target[3]はy=0により，生成時は0となるy座標である。  　入力した文字とターゲットの文字が一致すると，target[1].destroy()により該当のラベルは削除される。  　このプログラムではinputkeyの値は何かのキーが押されたときに変更される。このため，たとえば「a」のキーを一度押せば画面上にあるすべての「a」がヒットするし，そのままキーを押さなければその後出現する「a」がヒットすることになる。これを防ぐには，たとえば関数ControlGame()内で「inputkey=""」を入れるなどの方法がある。 | | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | | |

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **年度** | | | | 学習指導計画 | | | | | | | | | | | |  | | | | |
| **校長** | **教 科**  **科 目** | |  | | 必修 | **単**  **位** |  | | **学**  **年** |  | | **組** |  | 教材 | | | 書 名 | | 改訂版 プログラミング入門Python編 | |
|  | 選択 | 発行所 | | 数研出版 | |
| **担 当**  **教 師** | |  | | |  | | | | |  | | | | | |  | | | |
| **教頭** |  | | |  | | | | |  | | | | | | **記載者** |  | | **印** |
|  | **月** | **週** | **指導項目** | | | | | **指導内容** | | | | | | | **配当**  **時間** | | **留意点** | | | |
| **教務** |  |  | ※｢1｣または｢2｣のいずれかを選択する。  ◆総合演習  １　ピンポンゲームをつくる  ２　魔法陣をつくる | | | | | 左右の矢印キーでラケットを動かして，動くボールを打ち返すゲーム「ピンポンゲーム」の仕組みを理解させ，作成・実行させる。  　ボールの動きを処理する関数MoveBall()とラケットの動きを処理する関数MoveRacket()の各関数の意味を理解させ，入力させる。  　ラケットの位置やスピードなどの設定を自由に変更させ，動作を検証させる。gamespeed（処理の繰り返し間隔時間）とry（上端からラケットまでの距離）の値を小さくすると難易度が上がり，値を大きくすると難易度が下がることを体験させる。上下方向のはね返りの条件が「if (by==ry-ball and ～)」となっているため，ラケットではね返るには「by==ry-ball」になる必要がある。このため，変数の初期値を変更させる場合，とりあえずryおよびbvyは5の倍数にさせる。  　補充問題として時間がある生徒には，ゲームの経過時間を表示する機能を追加させる。  　列，行の数が奇数の魔方陣を作成する仕組みを理解させる。入力した次数が偶数の場合は，奇数を入力するまで繰り返す仕組みを理解させ，プログラムを入力させる。  　一番下の行の中央に「1」を入れ，「2」以降の数は，直前の数を記入したマスの右下のマスに入れる。右下のマスが範囲外に出る場合は，一番上の行や一番左の列に入れる規則を理解させ，プログラムを入力させる。  　補充問題として，時間がある生徒には，作成した魔方陣の「各行の和が一致する」ことを検証する機能を追加させる。さらに興味を持ち，時間がある生徒には，｢各列の和が一致する｣｢1からN\*Nまでのすべての数が重複なく入っている｣ことを検証するプログラムも考えさせる。 | | | | | | | 0.5  )  い  ず  れ  か  を  選  択  す  る  (  0.5 | | 生徒には，ボールが動く仕組みを確実に理解させたい。はじめに（bx，by）の座標にあるボールをキャンバスと同色の灰色で塗って見えなくし，次にボールの座標をx方向およびy方向の速度分bvx，bvyだけ変化させ（「bx+=bvx」，「by+=bvy」），変化した新しい座標に，赤色（red）でボールを描いている。そしてこの処理を行う関数MoveBall()を繰り返している。  　ラケットは，キーが押されたときに実行する関数MoveRacket()をbindさせている。  （root.bind("<Key>",MoveRacket)）生徒にはキーボードイベントの使い方を復習させ，知識の定着を図りたい。  　最初にNが奇数の魔方陣を作成するルールを理解させることが重要だ。実際に紙に書いたり，ワークシートに入力させたりすると理解が深まる。  　直前の数を記入したマスの右下のマスが範囲外に出るときの位置は，なぜ%演算子を使うと計算できるのかをよく理解させたい。  　補充問題では，「最初hantei\_waとbatuを0とし，最初の行の和をhantei\_waに保存後，他の行の和とhantei\_waを比較して一致しない場合はbatuを1にする」という処理がわかりにくいかもしれない。時間をかけて理解させたい。 | | | |
|  |
|
| **科主任** |
|  |
|
|  |
| 記  入  上  の  注  意 | 1. 教科または科目を，同一学年にて複数人で担当する場合，必ず共同で作成し同一計画により実施する． 2. 年間行事予定計画を踏まえて，また学習指導要領に則り，計画を立て実施する． 3. 他教科との連絡や，教科における３ヶ年カリキュラムとの関係を考えて作成する． 4. この用紙は各学年・各科毎に担当教師＋１部を作成し，１部は教務に提出する． | | | | | | | | | | | | | | | | | | |