

本事例の基礎データ

カテゴリ	プログラミングに関する指導方法		
学校種	高等学校	事例提供者	都立町田高等学校
学年	1 学年	教科等	情報科、情報の科学
単元（題材）名	役に立つプログラムを考えよう（アルゴリズムとプログラム）		
主な ICT 機器	・デスクトップ PC（CALL 教室）		
ワンポイント	情報 I を見据え、情報の科学におけるプログラミングの授業を Python 言語を用いて行う。		
「情報活用能力 #東京モデル」の位置付け	プログラミング	STEP 5	<ul style="list-style-type: none"> ・問題解決に向け、計画を最適化し、評価、改善しながら実行できる ・プログラミングに関し、情報の傾向や変化を捉え、効率的な解決策を考察できる ・効率化の観点から、プログラミングによる問題解決を最適化し、モデル化しようとする

本事例における教育の情報化について

【ポイント 1】	<ul style="list-style-type: none"> ● CALL 教室において、一人 1 台のコンピュータを活用する。 CALL 教室での大画面でプログラミングにも最適化された高速で高度な環境を一人 1 台活用することにより、来るべき情報 I の開発環境をイメージできる。
【ポイント 2】	<ul style="list-style-type: none"> ● Python 言語の活用 文部科学省「情報 I 研修資料」においても取り上げられている Python 言語を用い、より高度で効率的な環境を取り入れる。
【ポイント 3】	<ul style="list-style-type: none"> ● テキストエディタとフリーウェア統合環境および CUI の活用 統合的なフリーウェア開発環境「Anaconda」を取り入れるとともに、一般的に良く用いられているテキストエディタ「Terapad」を活用し、あえて CUI（Character-based User Interface）を取り入れることによって、小中学校とは違った本格的なプログラミングを体験する。

本単元（題材）における指導の流れ

時間	●主な学習活動 ・ 児童の活動	○支援・留意点 ☆評価
1 ～ 3	●コンピュータと情報処理 ・ CPU と論理回路 ・ コンピュータにおける四則演算と計算の限界	○ 2進法についての復習を交えながら進めていく。 ☆ and, or, not での演算を理解し、コンピュータでの計算の仕組みを理解する。
4 ～ 5	●基本的なアルゴリズム ・ 順次処理と加法・減法 ・ 繰り返しと乗法 ・ 判断分岐と除法	○ 前回までのコンピュータにおける四則演算を振り返りながら、計算とアルゴリズムについて触れていく。 ☆ 順次・繰り返し・判断分岐の3つのアルゴリズムを用いて考え表現できる。
8 ～ 10	●基本的なプログラム【ポイント1、2】 ・ 統合環境と CUI ・ Python 言語について ・ 変数と演算子 ・ 配列（リスト） ・ 関数	○ 一人1台の環境を用いながら、Python 言語の特徴と文法の基礎について触れていく。 ☆ Python での表現（変数・演算子・配列）を理解し、身近な関数を考え制作することができる。
10 ～ 13	●応用的なプログラム【ポイント2、3】 ・ テキストエディタと文字コード ・ ディレクトリとパス、階層構造 ・ 繰り返し ・ 判断分岐 ・ モジュールと乱数	○ テキストエディタを用いて、文字コードがどのように扱われているのかに触れるとともに、階層構造から情報デザインについても考えさせる。 ☆ 繰り返しや判断分岐を適切に考え表現するとともに、テキストファイルから Python を実行できる。
14 ～ 16	●課題制作【ポイント1、2、3】 ・ 乱数を使った問題解決 課題制作・発表・評価改善	○ 5行程度の実際のサンプルを見せながら、これに準じた内容で構わない旨強調するとともに、まずは完成させることを優先させる。 ☆ 「役に立ちそうな」「目的をもった」プログラムの制作を通して、問題解決と評価改善を行うことができる。

本時の流れ

段階	● 主な学習活動 ・ 児童の活動	○ 支援・留意点 ☆ 評価
導入	<ul style="list-style-type: none"> ● 乱数と、判断分岐または繰り返しを利用したプログラムの作成 ・ 一人1台環境で、個人で行う。 ・ 作業時間は2時間 ・ 意味が分からないものは作成しない。 	<ul style="list-style-type: none"> ○ 教員のサンプルを見せてイメージさせる。 ☆ 条件にあった具体的なプログラムをイメージできる。
<p>「役に立ちそうな」プログラムを考え、自分で作成しよう！</p>		
展開	<ul style="list-style-type: none"> ● ゴールのイメージ ・ 今までのワークシート等を見ながら、「役に立ちそうな」「面白そうな」プログラムの内容を考える。 ● アルゴリズムのイメージ ・ どのようなアルゴリズムが必要になるのかを考える。 ● コーディングとデバッグ作業【ポイント1、2、3】 ・ ゴールやアルゴリズムをもとに、今までの学習内容を参考にして具体的なコーディング・デバッグ作業を行う。 ● デモンストレーション ・ プログラムを実行し、ペアの相手に見学してもらう。 	<ul style="list-style-type: none"> ○ 机間指導をしながら、手が止まっている生徒に声をかけるとともに、苦手な生徒をフォローする。また、デモンストレーション時には、距離をとって後ろから見てもらうように指導する。 ☆ 自ら考えたゴールのイメージにあったプログラムを制作できる。 【思考力・判断力・表現力等】
まとめ	<ul style="list-style-type: none"> ● 振り返りと評価改善 ・ 他の生徒の内容を参考にして、自らのプログラムを振り返り、評価改善を行う。 	<ul style="list-style-type: none"> ○ 完成できなかった生徒は、周りの作品の内容を参考に完成できるように指導する。 ☆ 自らのプログラムを評価し、より良いものに改善しようとする。 【学びに向かう力・人間性等】

授業の実際

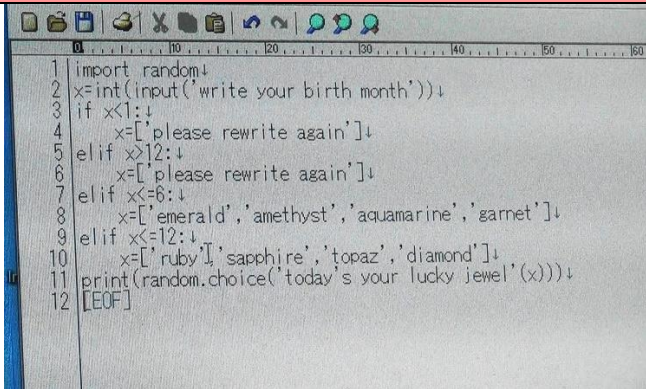
【ポイント1】一人1台のPCの活用



CALL 教室での一人1台環境

CALL 教室において、一人1台の大画面かつ高速実行環境を用いることにより、生産的で効果的な開発が行えるような取り組みを行っている。また、将来的に、Web やサーバからの各種サービスを利用できるような構成としている。

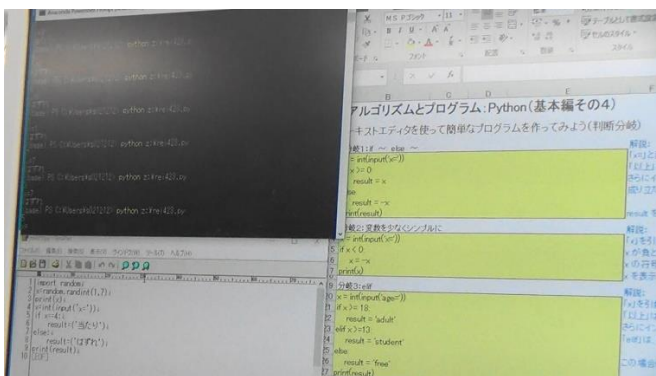
【ポイント2】●Python 言語の活用



Python 言語

「情報 I」を意識し、汎用性が高いテキスト言語 Python を活用するとともに、中学校までのプログラミング学習成果を基に、演算や変数の表記方法、簡単なアルゴリズムがどのように表現されるのかを学習する。

【ポイント3】●テキストエディタとフリーウェア統合環境および CUI の活用



テキストエディタと CUI 開発環境

テキストエディタ TeraPad や CUI 環境を用いることにより、他のプログラミング言語を扱う上でも汎用性が高くなるようにするとともに、フリーウェア統合環境を用いて、自宅等でも自らが主体的・実践的に取り組めるように工夫している。

今後に向けて

●生徒の学習意欲や能力に応じた発展的な内容の検討

現在、生徒の状況から、基礎の習得にどうしても時間がかかってしまうため、本格的な課題制作の時間がなかなか取りにくい状況となっている。より学習意欲が高く、発展的な内容をおこないたい生徒のために、課題制作に多くの時間を割くことができるようなカリキュラム上の工夫を考えていく。

●プログラミングの学習成果を他の分野に活用

プログラミングを身に付けた生徒が、例えばモデル化とシミュレーション等の分野でそれを活用し、効果的なシミュレーションを行えるようにしたり、Web制作を行った生徒が、Web上で Javascript 等のプログラミングを行うことができるようにしたりするなど、カリキュラム上の工夫を考えていく。