

考えてみよう

# ネットワークを利用した双方向性のあるコンテンツのプログラミング



Ando Akinobu  
**安藤 明伸**  
宮城教育大学教授

## 「ネットワークを利用した双方向性のあるコンテンツのプログラミング」の捉え方

新学習指導要領で、新たに追加された「生活や社会における問題を、ネットワークを利用した双方向性のあるコンテンツのプログラミングによって解決する活動」について、学習指導要領解説を元にその捉え方について説明します。

(2) 生活や社会における問題を、ネットワークを利用した双方向性のあるコンテンツのプログラミングによって解決する活動を通して、次の事項を身に付けることができるよう指導する。

ア 情報通信ネットワークの構成と、情報を利用するための基本的な仕組みを理解し、安全・適切なプログラムの制作、動作の確認及びデバッグ等ができること。

イ 問題を見だして課題を設定し、使用するメディアを複合する方法とその効果的な利用方法等を構想して情報処理の手順を具体化するとともに、制作の過程や結果の評価、改善及び修正について考えること。

ここでいうコンテンツとは、デジタル化された文字、音声、静止画、動画などを、人間にとって意味のある情報として表現した内容を意味しています。プログラムの中で、単にメディアを用いれば良いのではなく、情報の技術の見方・考え方(社会からの要求、使用時の安全性、システム、経済性、情報の倫理やセキュリティ等)に着目して、最適なものとなるよう用心することが大切です。必要以上にデザインに凝ることが無いように配慮しつつ、見やすさ、わかりやすさを考慮して、大きさや色、配置、音色等について考えさせて下さい。その際は、手書きでも構わないので、画面がどのように遷移するか、何をどの辺に表示するかなどワークシート等に設計させると良いでしょう。

また、ネットワークを利用した双方向性とは、使用者の働きかけ(入力)によって、応答(出力)する機能であり、その一部の処理の過程にコンピュータ間の情報通信が含まれることを意味しています。利用するネットワークは、インターネットに限らず、例えば、校内 LAN、あるいは特定の場所だけで通信できるネットワーク環境も考えられます。学習指導要領としては、この記述に留まっており、何か特定の通信技術や通信形態、プロトコル等

は規定されていません。技術の原理や法則、基礎的な技術の仕組みとして指導した内容と、実際に生徒が問題解決の取組みの中で行う活動とで扱う情報技術について、相違点などを理解させた上で、生徒の実態やコンピュータ室の環境を考慮して、生徒達にとって学びやすいものを取り上げると良いでしょう。

同様に、使用するプログラミング言語は、小学校での関連する学習経験などの生徒の実態を踏まえるとともに、課題の解決に必要な機能、プログラムの制作やデバッグのしやすさ、(3)で使用する言語との関連などに配慮して選択することが書かれています。テキスト型言語の方がより高度な処理を書くことができますが、「書く、つくる」ことが目的とならないよう注意が必要です。学習指導要領解説には、この学習では、プログラムの命令の意味を覚えさせるよりも、課題の解決のために処理の手順(アルゴリズム)を考えさせることに重点を置くなど、情報の技術によって課題を解決する力の育成を意識した実習となるよう配慮するよう言及されています。

双方向性のあるプログラムは、情報を入力し送信する処理が必須となります。そのためには通信上の処理の取り決めが欠かせません。こうした発想から、通信プロトコルの必要性へと関連させて指導することが可能です。

局所的な処理に集中しすぎると、全体的な処理の流れが捉えられなくなる可能性があります。そこで、課題の解決策を構想する際には、自分の考えを整理し、よりよい発想を生み出せるよう、アクティビティ図のような統一モデリング言語等を適切に用いることについて指導する必要があります。アクティビティ図も、必要以上に細かな書き方指導ではなく、思考を整理する手段として活用し、生徒達の考え方を評価してあげるようにして欲しいと思います。アクティビティ図としては、まずは大まかにどのような機能・処理が行われるかを書かせ、それを見ながら、実際のプログラミング言語でどのように記述するか分解して思考させると良いでしょう。

# 移行期間における「ネットワークを利用した 双方向性のあるコンテンツ」での試行

仙台市立広瀬中学校 教諭 鈴木 康洋

新学習指導要領で新たに指導することとなるネットワークを利用した双方向性のあるコンテンツのプログラミングによる問題の解決(以下「双方向のプログラミング」)は、従来のデジタル作品の設計・制作よりも多くの指導事項が含まれるため、全面実施に向けて移行期間でどのような準備を進めておくかが鍵となります。それと同時に、授業時数が変わらない中で指導事項が増えていることから、題材及びその指導計画の大幅な見直しが欠かせません。そして、入学してくる生徒は小学校でどのようなプログラミング言語でどのような体験をして、どこまでの知識があるのかという小学校との接続、双方向のプログラミングと計測・制御のプログラミングとの関連、技術による問題の解決の前過程として位置づけられた生活や社会を支える技術の指導、アクティビティ図等の活用など、移行期間で準備・検討することはたくさんあります。そこで今回は、学習指導要領解説にも例示され、かつ新学習指導要領で目指す学習活動がイメージしやすい入門的な題材として、ブロックプログラミングによる簡易的なチャットを制作することで問題解決していくプログラミングを第1学年で試行してみました。本稿では、それを踏まえて移行期間の取組みを提案したいと思います。

## 題材の設定

まず、今の移行期間の段階で第1学年を対象にしたのは、今年の第1学年の生徒たちは小学校でプログラミングは殆ど未経験であることから、ほぼプログラミングをゼロから指導するには、どのような工夫が必要になるのか検討したいと思ったからです。そして、使用するプログラミング開発環境は、アーテックロボ 2.0 を用いて Studuino:bit を使用しました。この意図は、①Scratch3.0 ベースでプログラミングできることで、中学校の他の教科でのプログラミングの実施にも展開しやすいのではないかとということ、②今後、学区内の小学校でScratch を扱うことが見込まれることから、少しでも多くの児童が操作方法などを既に習得して入学してくるのではないかとということ、③計測・制御のプログラミングにおいても、同じ開発環境で指導できるため、プログラミング言語の言語仕様の教え直しが不要になることを期待しています。表1は今回提案する指導計画の一部です。もちろんこの表に示した指導計画を実施する前に、情報の技

術における(1)の項目として、原理・法則及び基礎的な技術の仕組み等について指導し、情報の技術の見方・考え方に気付かせることができるよう指導しています。また、最後の「提言をまとめる」活動は、本来はDの(4)の内容ですが、今回は1年生で学習させていることから、題材のまとめとして位置づけております。

表1 題材指導計画(一部)

↓ 題材指導の流れ	・抽象化された手順をアクティビティ図で示し、手順を分解しその組み合わせ方を理解する。
	・「アーテックロボ 2.0」で信号機のプログラムを制作し、順次・反復・条件分岐を理解する。
	・「アーテックロボ2.0」のWi-Fi機能を利用したプログラムを制作し、情報を送受信する時のルール・取り決めの必要性について理解する。
	・前時で制作したメッセージ送受信プログラムの問題を保護者の立場として見いだす。既存のメッセージ共有アプリケーションを分析し、Scratchで実現できる課題を設定する。
	・設定した課題を解決するために、アクティビティ図で思考を整理しながら、制作とデバッグする。
	・自分たちの制作したプログラムについて、目的や設定した条件等を確認し、よりよいプログラムを制作するためにどうしたらよいか、ペアでの話し合い活動をする。 ・制作したプログラムを基に情報技術と社会とのかかわりについて考え、ワークシートに提言をまとめる。

## プログラムの作成

チャットの場合は、端末間でどのように情報が送受信されるのか、それがどのような通信技術なのか、通信する規則・ルールをどう決めるかが大切なので、それらのことをアクティビティ図の書き方も指導しながら確認するための時間を取ります。プログラミング的思考も意識させるため、意図する一連の活動は何なのか、それを実現するためにどのような処理が必要か大まかに捉えて一度アクティビティ図を書かせます(図1)。

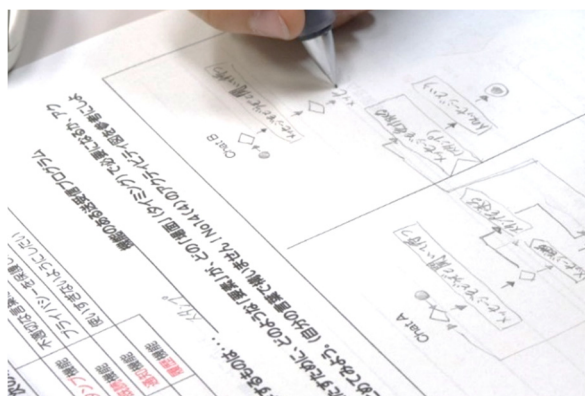


図1 ワークシートでアクティビティ図を書いてプログラムを設計する

次の時間は、アクティビティ図からプログラミング言語の処理に落とし込む考え方を覚えるために、信号機を題材にして、画面内で画像をスプライト(キャラクター)としてどう扱うか、音をどのように扱えるかというメディアの利用方法に触れつつ、今後の計測・制御の原体験としてアーテックロボ 2.0 のLED を光らせるといった発展的内容や、(1)の原理・法則的な知識との関連で RGB の値を変更して LED の色を変えたりすることも含めるようにします(ここで注意することは、あくまでも双方向のプログラミングとして指導するという点です)。これを1単位時間で扱うには指導内容が多く、実質的には操作方法を教師が指導するという形になってしまうため、可能であれば2単位時間くらい取れると生徒自身の考えを反映させるようなプログラミングもできると思います。この段階では通信機能を有しないプログラムの作成にしておきます。そして、その次の時間に、信号機の LED の切り替えのタイミングとパソコンでの画像や音の変化を、通信機能を使ってプログラミングします。通信機能を単に用いるのではなく、単体でできないことが複数台をネットワークでつなげることで可能になり、情報システムが構築され、互いが連携して処理をすることの意義やインパクトについて触れるようにします。

こうした基礎的なプログラミングの知識及び技能を体験的に習得させた上で、チャットをモデルとしたプログラムによる問題解決に取り組んでいきます(図2)。問題解決の取り組みは、何か立派なものを「つくる」ことを目的とするのではなく、やはり生徒自身が気づいた問題に対して、どのように課題を設定できるのか、その課題に対してどのようなアルゴリズムを考えられるかが大切だと捉えています。

そのため生徒自身に思考させるためには、プログラミングや言語仕様に関する知識も必要となりますが、今回対象とした中1の段階ではこれらのことを十分理解するよ



図2 送信側と受信側の処理の流れを比較して説明している様子

うな経験を有していませんでした。そこで今回の授業では、3段階のヒントカードを用意しました。実際の授業では、ある程度の量を印刷し教室に置いておき、必要なときに取りに来させました。表2は、その時の指導過程です。最初のヒントは、Scratch3.0 および Studuino:bit 自体の言語仕様に関するヒントです。これは、生徒が考えると予想される解決策で用いられるアルゴリズムごとに用意しました。例えば、発言履歴を見られるようにしたい、不適切な発言をフィルタリングする、スタンプ機能、着信を知らせる画面変化と着信音などから、保護者の立場として、使用時間を制限できる機能の必要性など、様々です。実際に生徒が取り組む中で、大いに悩んでほしいと思っています。しかし、その悩みがアルゴリズム的な思考であれば良いのですが、これはどのように設定するのか、配列(リスト)はどのように使うのかといったことは、プログラミングを行ううえでの基礎的な知識・技能であるため、ここは悩ませるのではなく、ヒントから気づきを得られるように考えて用意しました。第2ヒントは、アクティビティ図で処理の基本的な流れだけを示しました。本来であれば、このあたりからはプログラミングの醍醐味であるので、こうしたヒントは出来るだけ避けるべきかもしれません。しかし、今回の生徒を考えた場合、プログラミング経験が少ないことで、課題を解決する手順が出てこず制作に取り組みなくなるのではないかと懸念したことと、この第2ヒントで大まかにでも必要となる処理がイメージできれば、具体的な命令レベルに分解することを楽しめるといったからです。そして最終ヒントは、サンプルプログラムの提示です。ただし、あくまでも最小限の機能を実現するためのプログラムに留め、生徒たちにはプログラムを読解する力を要求するようなヒントにしています。実際のプログラミングにおいても、インターネット上からサンプルコードを見つけ、それを自分の目的に合うように改良することは少なくありません。他者が書いたプログラムを読み取り、どこをどのように変えることで自分のプログラムに適用できるのか考えることは、決して答えを見るような行為ではないことも伝える方が良いでしょう。今回の試行での反省点としては、生徒たちにヒントを見ることは自分で考えることを諦めることではないということを十分に伝えられず、ヒントを見たら負けなのだというような意識が残ってしまい、考えさせすぎてしまったことです。このことは喜ばしいことである反面、それだけを熟考できるほどの時間的余裕もないのが大変悩ましいところでした。最終的には、多くの生徒たちがヒントを参考にしながらも、そ

から着想を得て自分たちのプログラムを制作し、アイデアを実現していくことの面白さ楽しさに気づいているようでした。

### 今後の展望

今回は、1年生での実践をもとにして提案いたしましたのが、ほぼ初めてプログラミングを行う生徒たちを対象にしても、この流れで色々な課題が設定され、制作したアルゴリズムも様々でした。これが、2年生や3年生であればもっと複雑な課題を設定できることでしょうか。実際にScratch で作られたチャットが社会で活用されているわけではありませんが、色々な通信方法や送る・受け取るというデータのやり取りを決める重要性、設定した課題を生徒なりのアイデアで多様に解決していこうとする姿をみて、新学習指導要領で求められている資質・能力の重要性を認識することもできました。新学習指導要領にお

ける双方向のプログラミングの指導は、確かにこれまでの指導とは異なる難しさを感じます。より指導しやすい教材も今後出てくるでしょう。しかし、あまりにも目的に特化しすぎた教材や特定の目的のためのプログラミング言語で行う問題解決では、一般化したり、他の目的の解決に対して流用したりしにくいのではないかと思います。中学校においても、技術分野だけでなく他の教科でもプログラミングを推進して行くことや、技術分野内でも計測・制御のプログラミングによる問題の解決や他の内容においてもプログラミングを活用していけるよう、汎用性の高いプログラミング言語を扱えると良いと感じます。そのためにも教師自身がプログラミング能力を高めることと、これまで指導することがあまりなかった、配列やイベントなどの教え方など、教師自身が移行期間中に勉強しておくことが望ましいといえます。

表2 指導過程の一例

段階	主な学習活動	予想される生徒の反応	指導上の留意点
導入 10分	1 前時の学習内容の確認と、本時の学習課題を把握する。	・前時の学習内容を振り返り、本時の学習課題を確認する。	
	学習課題：送受信するプログラムの問題を解決する方法を考えよう。		
展開 30分	2 アーテックロボを接続し、メッセージ送受信プログラムを開く。	・前時で作成したメッセージ送受信プログラムを開き、ペアを組み、ネットワークグループを構築する。	・Stduino:bit の「ネットワークグループの指定」で、1～17 のグループを作らせる。
	手立てⅠ：アクティビティ図による思考の整理		
	3 ペアで課題を共有し、課題解決に向けて、アクティビティ図を書く。	・課題を確認し、課題解決への手順を整理する。	・アクティビティ図の書き方の指導ではなく、あくまで処理手順の整理をさせることに留意する。
	<div style="border: 1px solid black; padding: 5px; margin: 5px;"> <p>〈予想されるメッセージ送受信プログラムの課題〉</p> <ul style="list-style-type: none"> <li>・既読機能（相手がメッセージを見たことを確認する）</li> <li>・スタンプ機能（感情を相手に伝える）</li> <li>・パスワード機能（不正ログインを防ぐ）</li> <li>・着信機能（メッセージが来たことに気が付く）</li> <li>・タイマー機能（使用の制限をする）</li> <li>・NGワード機能（不適切発言を防ぐ）</li> </ul> </div>		
	4 処理の概要を示したアクティビティ図から、プログラムの具体的な命令の組み合わせを考える。	・論理的推論による試行錯誤をしながら、プログラミングを行い、結果を確認しながらデバッグする。	
手立てⅡ：レベルに応じたヒントカードの活用			
	5 デバッグの中で、ヒントカードを活用する。	・ペアのレベルに応じて、ヒントカードを活用する。	・ヒントカードを有効に活用するよう机間指導をする。
まとめ 10分	6 設定した課題と、課題解決のため制作したプログラムを発表する。	・解決することができた機能の処理の手順を説明し、うまくいかなかった処理があれば、困っている点についても説明する。	・プログラムとして効果的・効率的に変数やリスト（配列）、メッセージが使用されているプログラムについて補足する。