

プログラミングを「道具」とした中学校図形領域における課題学習
—小学生用ソフトを活用して—

Problem Learning in Junior High School Graphic Area
with Programming as "Tool":
Utilizing Elementary School Pupil Software

上原 昭三*

Shozo UEHARA

抄 録

小学校用のプログラミングソフトは、通常の言葉を用いて命令（プログラム）ができるようになっており、操作が容易である。また、図形描画のプログラムを作ることは、図形の性質など多くの数学的知識・技能を使った論理的思考を必要とするものであり、算数科だけでなく中学校数学科においても効果的な学習を展開することができると考えられる。

今回筆者が担当する科目「算数Ⅱ」で、小学校用のプログラミングソフトを教材とした授業を行った。その際、中学校の内容も取り入れた課題解決学習を行った。本稿は、その実践報告である。

I. プログラミングを用いた図形指導～中学校数学への転用可能性

1. プログラミングを使った算数の授業～新学習指導要領より

小学校学習指導要領（2017）では「児童がプログラミングを体験しながら、コンピュータに意図した処理を行わせるために必要な論理的思考力を身に付けるための学習活動」が位置付けられ、算数科では、その一つの例として、「正多角形の作図を行う学習に関連して 正確な繰り返し作業を行う必要があり、更に一部を変えることでいろいろな正多角形を同様に考えることができる場面などで取り扱うこと。」と記載されている。これを受けて、「小学校プログラミング教育の手引（第二版）」（2018）では、具体的な指導過程が記されている。図1は手引きの記述に基づき筆者がその授業の流れをまとめたものである。

左の囲みは、児童の活動、吹き出しは予想される児童の反応である。定規と分度器を用いた作図とプログラミングによる作図の双方を行うことによって、コンピュータを使うことのよさを実感させ、またプログラムを作成する中で図形の構成要素に着目した考察や論理的思考を引き出すよう計画されたものである。

* 関西国際大学教育学部 教育総合研究所学内研究員

2. 多角形の指導における算数と中学校数学のつながり

中学校数学科の大きな特徴は、「数学的な推論を図形の性質などの考察で活用することである」(文科省, 2017) と言われている。多角形についても、三角形合同や特殊な三角形, 四角形の性質などについて演繹的に確かめ, 論理的に考察し表現する学習が行われることになっている。つまり, 算数科で直観的または帰納的に確かめられた図形の性質を, 証明することを通してより深めるわけである。したがって, それぞれの図形(三角形, 四角形, 正多角形など)の性質については小学校で学習済みとはいえ中学校数学科の学習対象でもあるといえる。

例えば, 図1の授業においても, その考察や説明の場面で演繹的な方法や数学

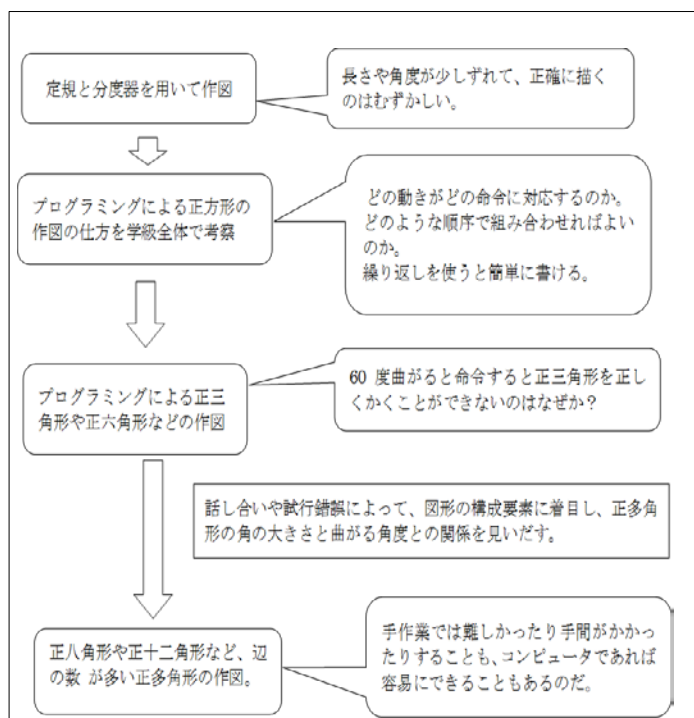


図1

的な記号や表現法を用いた場面を設定することによって非常に質の高い学習活動を展開できると考えられる。具体的には, 「 60° では正三角形が正しく描くことができないのはなぜか?」に対して小学生では試行錯誤の結果「 120° にすればよい」ことに気づき, さらに「なぜ 120° にすればよいのか?」という疑問から辺の延長線と辺によってできた角(外角)が指示する角であることを理解すると予想される。算数科では, その気づきをもとに正六角形などの作図プログラムを作成するのであるが, そこに, 「どうして外角に曲がる命令と同じ長さの線分を引く命令を頂点の数分繰り返すことで正多角形が描けるのか?」という問いを加味すれば, 中学校の知識を活用した演繹的な考察を必要となり, 中学生にとっても思考を深める学習にすることができるのではないだろうか。

このように, 図形領域においては「問い」の質を少し工夫することによって算数科の授業を中学校のものに転用することが可能である。

3. 中学校における課題学習の手法としての小学校プログラミング

小学校では, プログラミング言語を覚えたり, プログラミングの技能を習得したりすることそれ自体をねらいとはしないため, 通常の言葉を用いて命令(プログラム)が書けるソフトが開発されている。図1で紹介した授業もそのようなソフトの使用を前提としている。つまり, 中学生にとってはソフトの操作やプログラム作成そのものについては, 習得や理解が容易であり学習の目的ではなく「道具」としての使用できるわけである。

筆者は、これまで図形領域での課題学習として、折り紙やグラフ電卓、目盛りのついたロープなど一定の制限を持った「道具」で図形を作図し、その手順や根拠を考察・議論する授業を実践してきた。いずれも、活用型授業として効果が確認できたものである。

例えば、「紙を折る」という「道具」で図形を作る場合、「紙を折ってできる折り目を直線とする。」
「折って重なる部分は等しい。」という条件だけをもとに作図することになる。また、「グラフ電卓」の場合は、作りたい図形が座標平面上に描かれたときの各辺の方程式を予想し、その式を電卓に入力することによって作図するわけである。いずれも、定規や物差し、分度器などを使って手で描く場合に比べ多くの制限（長さが測れない。角度が測れない。など）がある。

このような限られた条件下で、図形を作成するには、数学的な考え方を働かせ、多くの数学的知識を呼び起こし、数学的な技能を活用しなければならないわけである。また、「道具」を使った作図は、数式を使って問題を解く通常の学習（算数・数学）と違い子供たちにとって一種の創作活動であるため、図形が完成した時には、達成感が得られると思われる。さらに、一度描いたら消しにくい紙の上に描く作業と違いこれらの「道具」では、何度でも直前の状態に戻せるため、試行錯誤が起こりやすいと考えられるのである。

このような視点で見えていくと、ソフトを操作してプログラムを作成することは、折り紙や電卓、ロープなどと同様に「道具」として使用した場合に学習効果を高める要素を含んでいることが分かる。その上、他の「道具」に比べ、ICT環境が整えば、その授業のために特別に教具を用意する必要がなく事前準備が軽減できるという利点がある。つまり、算数科で実施予定のプログラミングは、中学校での課題学習に非常に有効であるといえる。

II. 実践のねらい

今回の実践は、秋学期科目「算数Ⅱ」（教育学部3年生9名受講）の3コマ分を使い、次の（1）～（3）を狙いとして実施した。

- （1）小学生用のプログラミングソフトを活用した図形領域における課題学習の授業案を作成する。
- （2）大学生を生徒に見立て授業案にもとづいた授業を行い、現場での実施に向けて想定できる効果と問題点や課題を抽出する。
- （3）生徒の立場で、実際にソフトを活用して授業を受けることを通して、小学校教員を目指す学生のプログラミング教育における技能と考え方および意欲を伸長させる。

III. 実践の概要～「ロボットに図形を描かせてみよう」（全3時間）

今回は、Web上に公開されているソフト「プログル」を使用した。このソフトには、図2にある①～⑤の命令が設定されている。この授業は、これらの限られた命令（条件）組み合わせて、課題となる図形を描く一連の命令群（プログラム）を組み立てさせようというものである。

- ① () 前に進みます
 ロボットが前に進むことによって、数値分だけの線分を引くことができる。
- ② () ° 右に向きます
 ロボットが現在の進行方向から数値分だけ右に向きを変えることができる。
- ③ () ° 左に向きます
 ロボットが現在の進行方向から数値分だけ左に向きを変えることができる。
- ④ () 回繰り返すこと
 この命令の下位にある動作を数値分だけ繰り返す。この命令を使うことによって、同じ作業が複数ある場合、プログラムを短く編集することができる。
- ⑤ () 前にジャンプ
 線を引かずに数値分だけ現在向いている方向にロボットを動かすことができる。この命令を使うことで、線分の始点を任意に定めることができる。
- ※ () 内には数値が入る。また、この他、色の設定に関するものが2つあるが、図形の作成に直接かかわらないものであるため、今回は使用しないこととした。

図2

例えば、正方形を描かせる場合は、図3に記したような命令となる。これは、「90° 右に向きを変え、100だけ線分を引く」ことを4回繰り返すという命令である。つまり、最初の向きから右へ直角に向きを変え線分を引き、その線分の終点で右90° 曲がり同じだけ線分を引くという作業を繰り返すことを意味するものである。向きの変更(90°)が4回、線分を引くことが4回繰り返すわけであるから、正方形が描けるというわけである。

4回繰り返すこと
 90° 右に向く
 100前に進む

図3

1. プログラミングソフトを使ってみよう…1時間目前半

基本的な操作について、スクリーンに投影させ実演しながら教師の方で説明する。続いて、正方形を描かせる命令(プログラム)を生徒とやり取りしながら作り、PCで実行させて確認していく。

用意されている命令やその操作の仕方は、中学生にとってはそれほど難解なものではない。そこで、時間短縮も考えて全体で確認するのがよいと思われる。ただ、一方的な実演と説明では、定着が不十分になることも危惧されるわけであるから、じっくり生徒たちとやり取りし、確認しながら進めていく必要がある。

生徒たちが、操作方法(特にプログラムの組み方)を理解したことを確かめるため、簡単な図形の作図プログラムを作らせる。正方形は、「前に同じ長さだけ4

100前に進む
 90° 右に向く
 100前に進む
 90° 右に向く
 100前に進む
 90° 右に向く
 100前に進む

図4

回進む」(同じ長さの線分を引く),「同じ向きに 90° 4回曲がる」(4つの線分のなす角が 90° にする)ことによってつくることができる。このような、正方形の条件とそこから考えられる作図法(ロボットの動き)は生徒にとって思いつきやすいものであり、操作方法の定着を確認する「問い」として有効であろう。

今回の授業においても、早い段階で(ほぼ即答に近い)図4のような解答が返されている。その後、他の学生から繰り返し命令を使うことについて助言があり、図3にある短いプログラムに修正された。このやり取りによって、繰り返し命令によるプログラムの編集ということも含め各命令の意味や使い方が全体化された。

2. 基本的な図形(多角形)を描くプログラムを考えよう…1時間目後半

課題として、基本的な多角形(正三角形, ひし形, ひし形でない平行四辺形, 等脚台形)を描くプログラム作成を提示する。「紙上でプログラム案を作成(個人)→他者と検討(ペアまたは4人程度のグループ)→PCでの実行による検証」という流れで課題解決に取り組ませる。

ここからが、多角形描画のプログラム作成という課題解決である。図形についての既習事項(性質や成り立つ条件)を活用しながら、限られた条件(使える命令の数と内容)のもとで目的の図形を描く手順を作っていくことになる。その際、フリーハンドなどで描くときにはほとんど意識していない一つ一つの操作を分解し、その意味を確認するとともに、その操作をPC上で実現するには、使える命令をどのように編集すればよいかを考えなければならない。よく知っているつもりであっても、少し戸惑う生徒が少なくないと思われる。そこで、この時間の課題としては、一目「簡単にできそう」と思えるようなその性質などをよく理解している図形から入っていくのがよい。今回は、4つの図形を課題として用意した。

(1) 正三角形

事前に、正方形のプログラムを確認しており、それと同じ考え方を使えばよい。生徒の多くも、「4回繰り返し」を「3回繰り返し」に、「 90° 右に向く」を「 60° 右に向く」に変えるだけでよいと判断することが予想される。しかし、その変更でプログラムを実行しても正三角形は描けないのである。「(\quad) $^\circ$ 右を向く」という命令の意味をよく吟味すると、入力する角度は、ロボットが進んできた向きに対して次に曲がる角度すなわち多角形の外角を意味している。そこに気づけば、角度を 120° に変えることで、プログラムを正しく修正できるわけである。「簡単そうに見えて少し戸惑う」最初に行うにはちょうど良い課題であると思われる。

「内角が 60° でも、ロボットは 120° 回転させないといけないのが難しかった。」(学生の感想) 今回の実践でも、曲がる角度を 60° と思い込み、実行した図形が正三角形にならず少し考える学生も見られたが、すぐに曲がる角度が外角であることに気づき 120° と修正していた。文科系の学生とはいえ、正三角形の形状や性質、外角と内角の関係について熟知しているわけであるが、つい思い込みで勘違いしてしまったということであろう。したがって、中学生においても同様の反

応をする生徒が多数出現することが十分予想できるわけである。その時の「あれ?」「どうして?」をとりあげ、少し命令の意味を冷静に考えさせれば、簡単に見える課題であるが生徒の思考を深めさせることができると考えられる。さらに、その「気づき」を説明させるようにすれば、外角の定義についても再度確認させることができ、「外角の和はどのような多角形であっても 360° である」ことなどについても実感を伴った理解につながるのではないだろうか。

(2) 正方形ではないひし形

ひし形の定義(4つの辺が等しい)とひし形が平行四辺形であることによる性質(2組の対角がそれぞれ等しい)を使って命令を組み立てる課題である。フリーハンドなどで「見た目からイメージできる形を描く」との違い、プログラムを作る場合は、図形の要素(辺、角)の性質をよく確認しておかなければならない。つまり、この課題を解決することで、ひし形の定義および四角形の包摂関係(台形、平行四辺形、長方形、ひし形、正方形)を再確認することにつながるわけである。また、正方形や正三角形と違い角度を自分で設定することや繰り返しを行う命令をどのように作るかなどプログラム作成ならではの難易度の変化もある課題である。

図5は、今回の授業で現れた学生とのやり取りである。S1のつぶやきに教師が応じ、全体で「ひし形」の特徴を確認することができた場面である。「ひし形」をイメージしたあと、命令を作ろうとする際、特徴を言葉にすることの必要性を感じたことからS1が発せられている。

S ₁ :ひし形ってどんなんやった?
T ₁ :ひし形には、どのような特徴がありましたか?
S ₂ :対角線が 90° ?
T ₂ :「対角線が垂直に交わる」でしたね。それから
S ₃ :4つの辺が・・・等しい?
T ₃ :「4つの辺が等しい」。他にありましたか?
S ₄ :向いあう角が同じ?
.....

図5

この「問い」が生徒から思わず出てくるのがこの課題の一つの狙いでもある。この問いが生じることでこれまで、受動的に習ってきた(いわゆる「憶えてきた」)図形の定義や性質を、使える知識として能動的に学びなおせる(復習)きっかけが得られるのである。平行四辺形や長方形などにくらべると(小中の授業での)扱っても小さく、身近に多く見かけるにもかかわらずその定義や性質を意識する機会が少ない「ひし形」を課題にすることで、多くの生徒からこの「問い」が生まれやすいと考え、2つ目の課題に据えているわけである。今回は、教師から発問し、図形の形状を注視させながら意見を誘導し確認しているが、生徒が疑問を持っているのであれば、各自に教科書等で確認するよう促すことも考えられる。いずれにしても、(生徒が)図形の定義や性質をしっかりと整理しておくことの必要性を感じ進んでそれ確認することが肝要である。

次は、「確認できたいくつかの性質から命令に使えるものを選択する」、「数値を自分で設定し具体的な命令を作る」という2つの作業である。前者は、正方形や正三角形の経験から、「辺の長さ」と

「角」に関する性質に目が行きやすく判断に迷うことは少ないと思われる。しかし後者については、(筆者の経験では) 以下のように指導者側が考えるほど容易ではない。

例えば、「2組の向かいある角がそれぞれ等しい」という性質から 50° 、 130° といった特定の角度になる内角を決め、具体的なひし形を設計することになる。要するに、一般化された定理を特殊化する作業である。実は、算数や数学(中学校)の学習の多くが、具体的な事例(図形など)から共通する法則を抽出して一般化(記号、式、言葉で表現)する方向になっている。法則に合う数値を適当に選ぶという機会は少ないのである。したがって、数学のあまり得意ではない生徒にとって、「適当に」角を設定することもちょっとしたハードルになり、そこで思考が止まっている場合が少なくない。さらに、一つの角度を決めたとき、残りの角度をどのように決めていくのかということも少し考えが必要である。好き勝手な数値にするわけにはいかない。「向かい合う角が等しい」と「4角形の内角の和は 360° 」を組み合わせて、計算によってもう一方の角を決定できることを見出さなければ見通しが持てないわけである。このように、簡単に見える作業であるが生徒たちにとっては、深い思考を必要とする数学的な考え方(特殊化)を実践する場面でもある。

具体的な数値(曲がる角度と辺の長さ)が決まれば、命令を書いていけばよいのであるが前回までと違い曲がる角度が2種類となるため、繰り返しの回数と範囲を変えないといけない。この作業でも、命令を解釈しロボットの動きを念頭に浮かべながら、作られる図について考察することという、「深い思考」が行われることになる。

(3) ひし形ではない平行四辺形

この課題は、これまでのプログラムの作り方が定着しているかどうかを確認するためのものである。(2)のプログラムの進む(辺の長を描く)命令の一方の数値を変えるだけで良いことに気づけば簡単に作ることができるわけである。今回も、すべての学生が短い時間で完成させている。

(4) 等脚台形

等脚台形は、平行でない1組の向かい合う辺の長さが等しい台形である。手で描く場合は、これまでの四角形とさほど難易度に違いはない。しかし、このプログラミングソフトで描かせようとするとき少し考えることが増える。

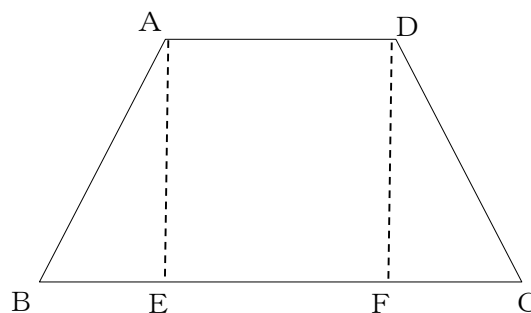


図6

命令をするには、各辺の長さや角度(外角)を設定しなければならないが、これまでの四角形と異なりすべての辺の長さが正確に入力できる数値(自然数)になるものばかりではない。図6の台形ABCDでいえば、ABとADいずれも自然数にできるものが限られている。したがって、課題

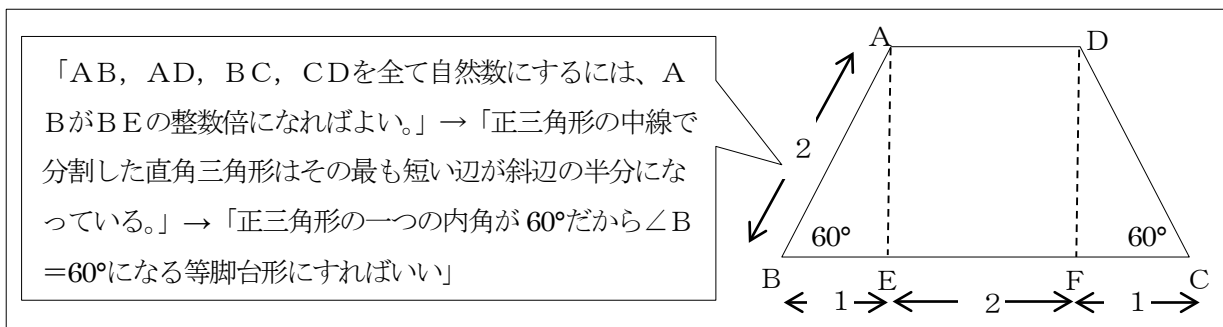


図6

(1) ~ (3) のように辺の長さを任意に定めることができないし、角度についても特別な角でなければならなくなるわけである。つまり、図形の要素を分析して、命令に変換するという作業の他に、すべての辺が自然数にできる等脚台形を求めるという数学的な問題が加わっている。後半の問題を解決するには、直角三角形ABEのABとAEが整数比にできるものを探さなければならない。結論から言えば、 $\angle B = \angle C = 60^\circ$ とすればよいのであるが、そこに行きつくためには、等脚台形の性質以外の既習事項を活用した多少複雑な思考過程を要する(図7)ことになる。

このように、この課題は図形を描くプログラムを作成するために、新たに数学的な課題を見出し、既習事項を活用して解決を図るものになっている。「具体的な(数学以外の)課題解決のために、数学的な問題を設定する」という数学的な活用力を育む上で有効なものになっていると考えている。

3. 描いてみたい図(図形)のプログラムを作成しよう…2時間目

各自ロボットに描かせてみたい図形を構想し、そのプログラムおよび、描画手順の意味とそれが意図する図形になる数学的根拠を示した説明を作成する。

ソフトの使い方に慣れたところで、次は機械が図を描くということの「よさ」と自身の発想を生かして図形を作成しようというわけである。人が描くのと違い、機械は正しく命令しておけば正確に何度でもその動作を行うことができる。それが「よさ」の一つであろう。例えば、正二十角形の作図を普通の人間が行うことは、技術的にも時間的にも大変困難なことである。ところが、機械を使えば、正三角形の作図プログラムと同様に3行の命令で描くことができるのである。

そのような「よさ」を活かしつつ、前時に習得したプログラムの技法を組み合わせで自分なりに「おもしろい」と思える図形を描くプログラムを作成しようというのがこの課題である。この課題では、用意された命令の組み合わせで描くことができるかどうかを検討しながら図形を構想することが必要であり、その分難易度が高い課題解決活動になる。与えられたものではなく、自身がより興味深い図形を、限られた命令の組み合わせで描けるかどうかを試行錯誤的に検討し、数学的に検証しながらそのプログラムを作成することは、「主体的」で「深い」学びにつながる作業になるのではないだろうか。

今回の実践では課題を提示する際「できるだけユニークなものを提案してほしい」「多角形に限らなくてもよい」ということを伝えた。学生たちも「ありきたりでない図」を作成しよう工夫してくれたようである。プログラムを作って実行し、修正を重ねるといった作業を長時間繰り返していた。

予想した通り描画させることができたときは、歓声上がる場面も見られた。図8（吹き出しは筆者）は学生が作成したプログラムである。立方体の見取り図であるが、平面図形の組み合わせ（正方形と平行四辺形）空間図形の見取り図が描かれているという意外性（普段は意識しない）や点線を「ジャンプの繰り返し」で描くなど多くの工夫や発想に富んだものである。この他にも、星形多角形や花びらの形など興味深い作品が作られている。

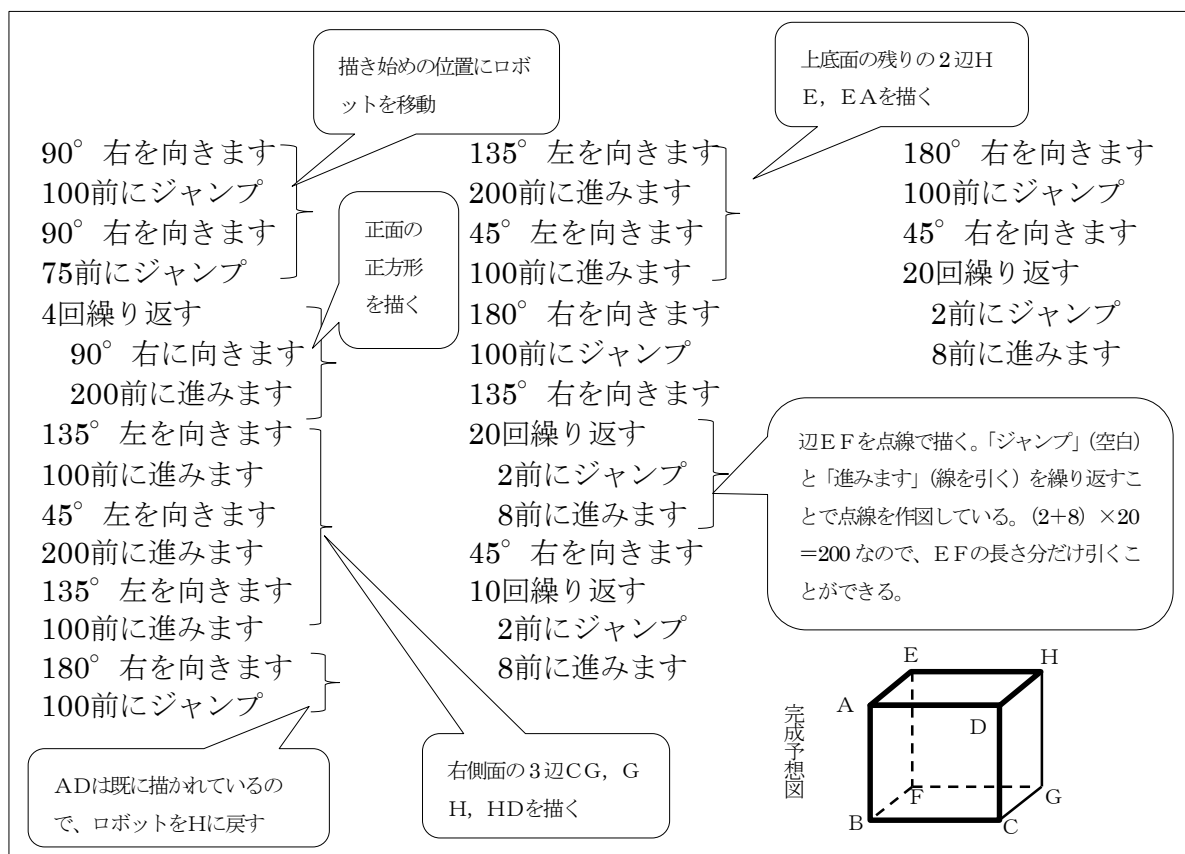


図8

一方、プログラム作成と同時求めたその意味と根拠については、図9のように各命令の意図やそれによって描かれる部分など、手順についての説明がほとんどであり、定義、定理などを根拠としてその図になることを立証するような記述は少なく、数学を活用して論理的に説明するという点に関しては、課題が残る結果となった。

まず正方形を描くので、進んで90° 曲がると同じ長さ4回繰り返します。そして奥行は辺の長さの2分の1で、角度は45° なので、辺の長さの2分の1進む前に135° 左になり、前に進みます。そして、左に45° 曲がりまた辺の長さ進みます。それを繰り返し、奥行きなどを描いていき、実線の部分は完成します。点線の部分を作るには「2前にジャンプ」「8前に進みます」を繰り返し進むべき距離だけ進むようにします。

図9

4. 他者のつくったプログラムを読み解こう…3時間目

グループ（2～4名程度）を作り、作成したプログラム（命令）を発表する。発表者以外は、そのプログラムを読み解き、予想図をワークシートに記入する。それぞれが予想した図についてプログラムと照らし合わせながら、その根拠を説明する。最後に、作成者がプログラムを実行させ、各自の予想を検証する。

この時間は、各自が作成した作品をグループ内で発表しあうことになる。ただ、発表者がプログラムを説明し、それを実行してみせるといふ一方方向ではなく、プログラムを解読させるといふ少しクイズ的な要素を入れてみたい。自分が作ったプログラムは、もともと作りたい図形が見えているのであるからその意味はよく分かるが、他者が作ったものはただ文字で書かれた命令が並んでいるだけであり、その一つ一つを理解しながら頭の中で整理し図形を組み立てなければならない。簡単ではない作業となる。このような念頭操作を行うことは、文字や記号で書かれた条件にもとづいた図形をイメージする力を鍛えることになり、数学的な考え方の伸長につながるものである。

今回の実践では、人数が少ないこともありグループを作らず発表者以外の全員で解読作業を行った。工夫を凝らした長いプログラムが多かったこともあり、「考えるのは楽しかったけど、なかなか想像していた図とは違うものが出てきてびっくりしました。」「純粋に楽しかった。文字だけからどのような図形が出来上がっていくのかを考えるのは思ったより難しかった。」（学生の感想）と、深く考えながらもなかなか正しい図を（紙に）描くところまでは至らなかった者が多かったようである。

一方、「自分が考えられないような作図をしていたり、円に似たような作図を用いるような図があり自分も参考になるような図があった。一つ一つ確認することができたら解くことができた。」（学生の感想）と他者発表から自身の作品の改善点を見出したり、さらなる創造の意欲を得たりする者もいた。

IV. 実践を終えて

今回実践は、次の3つの目的をもって行ったものである。授業を振り返り、実践の成果と課題について、それぞれの項目ごとに検討する。

- (1)小学生用のプログラミングソフトを活用した図形領域における課題学習の授業案を作成する。
- (2)大学生を生徒に見立て授業案にもとづいた授業を行い、現場での実施に向けて想定できる効果と問題点や課題を抽出する。
- (3)生徒の立場で、実際にソフトを活用して授業を受けることを通して、小学校教員を目指す学生のプログラミング教育における技能と考え方および意欲を伸長させる。

(1) について

筆者が中学校現場で行い効果があった実践を参考に、Web上に公開されているソフト「プログ

ル」を教材とする中学校2年生図形領域における課題学習のプラン（3時間設定）を作成することができた。プログラムを作成する過程は、数学的な法則を活用して図形を作成する方法を論理的に導くプロセスを含んでいる。一定の制限を有する手法（紙を折る、関数電卓に式を入力、ロープをつかう、定規とコンパスのみで描く 等）で工夫して図形を作成するという点で、筆者が実践してきた課題学習と共通するものであり、学習効果が期待できる。特に、ICT環境さえ整えば、特別な教具を事前に準備する必要がなく実施しやすいという利点がある。

今回は、図形の論証と三角形・四角形の性質と成立条件に関する学習後の課題学習として3時間の設定の授業案とした。人数も少なく大学生が対象という言うこともあり、PCソフトの操作の習得も速く課題への移行もスムーズであったが、基本的なものとして提示した課題を最後まで到達できないものもいる状況であった。大人数が対象であり、PC操作になれていない生徒が少なくない中学校においては、より前半部に時間を要することが予想されることから、1時間目後半部の課題の質（難易度）と数について再度検討が必要である。

2時間目の内容について、事前に宿題として図形とプログラムの構想を考えさせる（ワークシートの事前配布）ことも有効である。今回も、学生の意欲の高まりとともに工夫のアイデアが続々と浮かんでくるようで、そのために時間内に終わらない者もあった。単元末に位置付ける授業のため、多くの時間を費やしにくい、生徒が主体的に思索、創作するこの時間は確保すべきかと思う。

（2）について

既習事項を活用した課題学習の場合（全く関係ないとは言えないだろうが）上位学年で学ぶ知識はあまり影響されない。むしろ、その問題解決で活用される主要な既習事項について熟知しているかどうかが大切である。その意味で、数学の学習を特に苦手としない生徒を想定した場合、今回提案した授業では、大学生で実施しても反応に大きく違いがないと思われる。学生の取り組み状況から、この授業の効果および課題として以下の①～⑤を確認している。

①外角と内角の間違いは起こりやすい→プログラミンの難しさと興味深さの気づき

ロボットが曲がる角度が図形の外側の角（外角）であることは、言われてみればわかることであるが、通常図形をイメージするときは内角が見えており、外角を意識することが少ない。そのため、正三角形の作図の際曲がる角度を 60° と入力して失敗するケースは多いと思われる。

しかし、そのことが機械に指示することの難しさであり興味深さでもある。今回の実践でも、やはり「できた」つもりで自信満々でプログラムを実行し、「あれ？」→「そうか！」と失敗から気づきを得た学生が数名あった。下記は第1時間目授業後の学生の感想であるが、その多くが、手作業にない機械に命令することの難しさを意識したことを記している。

感想1：自分で動きを想像して、「的確に伝える」と「端的に伝える」のをしないといけないので頭の中で整理しないとできないなと思った。

感想2：角度や長さに気を付けないと自分の考えをロボットに伝えることができない。一度図形を完成すると少し応用するだけで向きなどを変えられる。

感想3：頭の中で分かっているのに、それを「向きも分からない、命令通りにしか動けないロボットに指示するのが難しかった。当たり前なことでも、その理屈を理解しなければいけないことが分かった。

感想4：作図するほうが楽にできるが、どのように進めていくかなど頭で考えながら行うことができるので、とても良い授業だと思う。また角度や性質も改めて復習することができた。

感想5：右に進めという命令を、「まず何度右に向いてからどれくらい進むなどといったように、幾つかに分けて指令をしなければいけない」というまどろっこしさもあるけど、同じ図形を描くのも、指令の簡略化ができる。

感想6：内角が 60° でも、ロボットは 120° 回転させないといけないのが難しかった。ロボットがどう動くのかイメージしながら考えないといけないのも大変だった。

②図形の性質、定義が即座に浮かばないことも多い

図5のやり取りにあるように、それまでの授業で習得した図形の性質であっても、課題解決（プログラム作成）の場面に対峙したとき、即座に必要なものが浮かばず見通しが立たない生徒も少なくない。そのまま放置すれば、困難感を増大させ意欲を減退させてしまう場合が多い。今回の実践では、生徒のつぶやきを拾い全体を図形の性質確認の方向に向けさせて、復習を行うことができた。その結果、全員が一定の見通し（「図形の定義や性質（決定条件）を基に考える」）を持つことができ、課題に取り組み続けることができた。このように、生徒たちが、課題となる図形の定義・性質に目を向け既習事項を確認・復習する場面を作れるような仕掛けを用意しておくことが大切である。

③プログラムの実行が課題解決の検証・評価となるため、生徒の達成感と創作への意欲喚起する

プログラミングを課題とする課題学習のよさの1つは、頻繁に検証作業ができることである。できたプログラムを実行すれば、課題完成に向けてどこまで到達できているのか（または、誤った方向に向かっているのか）が確認できるわけである。今回の授業でも、課題の難易度が上がるにつれて、プログラム作成→実行→修正を繰り返す姿が見られた。特に実行→修正の間で何度も頭をひねる場面もあったが、それだけにソフトが予想通りの図形を描いた時には達成感を得られたように見受けられた。先にも述べたように時間を忘れて創作に取り組むほど集中できたのも、このような試行錯誤の容易さによるところが大きいのではないだろうか。

④プログラム作成に数学的思考を活用しているものの、論証方法を活用した説明に課題

図形描画のプログラム作成では、学生たちが数学的な知識・法則を活用して論理的に筋道立てながら作業を進めてきたことは先に述べたことから明らかである。一方、思考の過程を振り返り、

それを整理して言語化することに関しては、大学生であっても難しいことが分かった。

今回は、作ったプログラムについての説明をワークシートに記述することを求めたが、図9、図10のように「その命令で何が描けたか」といった命令の意味についての説明にとどまっている。しかし、彼らが一つ一つの命令に至るまでには、その背景にその根拠となる数学的な定理・法則が存在しており、当然命令を作る過程でそれに基づいた数学的な推論が行われたはずである。

すべての辺の長さが等しいので前に進む。距離は全て同じにする。星のとがっている部分は 60° 、内に折れている部分は 120° なので、星のとがっている部分を作るときは「 120° 向く」にし、内に折れている部分は 120° 向かせた方と逆の方向に 60° 向かせなければいけないので、「 60° 左を向く」にする。曲がって前に進む作業を6回繰り返すと「六芒星」になる

図10

にもかかわらず、その点については説明の必要性に気付いていないかのようである。

この件については、数学的な論証の効果や必要性、活かし方（方法と数学的根拠を同時に説明する場合など）の指導の在り方を含め、授業プランの再検討が必要である。

⑤数学的思考の面白さや既習事項を深く理解しなおす必要性を実感する機会となっている。

下記は、一連の授業の最後に書かれた感想の1部である。1時間目の感想で既にプログラムを作ることを通じて、既習事項のより深い理解の必要性を多くの学生が実感していたことは読み取れていたが、授業の進行と共によりその意識は深まったようである。また、進んで考えることの楽しさについての記述も多く、「主体的な学び」への効果が期待できる。

感想7：繰り返しが多く、円を描こうとするととても時間がかかることが分かった。一つ一つ計算して考えておかなければ、正確な図形を描くことができないことが分かった。

感想8：プログラミングをするには、角度や長さの計算を緻密にしなければいけないと知りました。ただ勘で図を作成するだけではダメで想像していたより難しかったです。

感想9：考えるのは楽しかったけど、なかなか想像していた図とは違うものが出てきてびっくりしました。

感想10：純粋に楽しかった。文字だけからどのような図形が出来上がっていくのかを考えるのは思ったより難しかった。

感想11：一つ一つ考えなければ、上手に図形を描くことができないことが分かりました。人では正確に描けない図形でもコンピュータなら正確に描くことができるととても面白かったです。

(3) について

学生たちには、当初から小学校で授業することを意識しながら課題に取り組むよう指示しており、ソフトの使い方の習得はもとより、児童に説明するポイントなどを考えながら取り組んでくれたようである。また、ここで課題として提示した図形は、すべて小学校でも扱うものである。その論証は行わないもののプログラムの根拠となる図形の性質は、彼らが指導するものである。つまり、今回の課題学習は、少し工夫することによって小学校（5，6年）の授業に転用できるものである。

図11は、科目「算数Ⅱ」の課題として作成させた小学校で実施した場合の授業案に関する発表スライドから引用したものである。生徒の側から取り組んだ際に気づいた教師側の配慮を織り込み指導上の留意点を加えたものになっている。まだまだ小学校現場に持ち込んで行うには、検討改善が必要であるが、彼らにとって新学習指導要領に示された算数科におけるプログラミング教育を実施する上での最低限の材料を得たのではないかと考えている。

<p style="text-align: center;">ねらい</p> <ul style="list-style-type: none">・プログラムを使って正多角形をかく方法を考えることができる。・外角と内角が違う図形を作図する際、外角を命令で使うことが理解できる。・プログラムで正多角形をかくことによって、正多角形の性質を再確認する。	<p style="text-align: center;">導入</p> <ul style="list-style-type: none">・既習事項である正方形や正三角形、正多角形の性質を子どもの発言によって再確認する。 <p>性質 →正三角形は角度がすべて同じ・辺の長さがすべて同じ</p> <ul style="list-style-type: none">・正三角形、正四角形、正六角形を前に写しだし、内角の大きさ、辺の長さが同じことを視覚的に確認する。
<p style="text-align: center;">展開</p> <ul style="list-style-type: none">・プログラムの基本操作を知り、正方形を作図する・正三角形を誰が1番早くかけるか競争し、終わった子は黒板に名前マグネットを貼りに来させる（内角で命令したら正三角形にならないというまづきを与える。）・早くできた子が自分の命令を発表する・失敗例を先生がプロジェクターに移し、どうすればうまくいくかを子どもに発言させる。・正三角形と正四角形の書き方を振り返った後、正六角形をかくヒントを与えて各自で考える	<p style="text-align: center;">まとめ</p> <ul style="list-style-type: none">・各図形の書き方を振り返る（正三角形の例：長さや角度は同じ、角度は外側の角度を命令に使う…など）・本時で学んだことを生かしてどんな図形が作れそうか、どんな図形が作りたいかを話し合う。（例：6以上の正多角形など）・パソコンがあれば家でも出来るということを伝える。

図11 学生の作った小学校での授業展開例「正多角形と円」

以上のように、小学校用プログラミングソフトの中学校図形領域への活用は、修正すべき課題があるものの、その学習効果を期待できるものであるといえる。今回の実践で見えてきた課題を分析し、授業案を改善し中学校現場での本格的な実践につなげたいと考えている。

参考・引用文献

特定非営利活動法人みんなのコード (2017) 「授業で使えるプログラミング教材『プログル』」

<https://proguru.jp/>

文部科学省 (2017), 「小学校学習指導要領」

文部科学省 (2017), 「中学校校学習指導要領解説数学編」

文部科学省 (2018), 「小学校プログラミング教育の手引 (第二版)」

Abstract

The programming software for elementary school has instructions (programs) to use ordinary words, and it is easy to operate. In addition, making graphics drawing programs requires logical thinking using many mathematical knowledge and skills such as the nature of figures, and it is effective not only in mathematics department but also in junior high school mathematics department. It is thought that it is possible to develop.

In the subject "Mathematics II" that I am in charge of this time, I conducted classes using elementary school programming software as a teaching material. At that time, we conducted problem solving learning incorporating the contents of junior high school. This paper is a practical report.