

育成すべき資質・能力から見た情報科の存在意義と 望まれる指導内容・方法

松田 稔樹^{1) 2)}

要 旨

新学習指導要領の基本方針は、「育成すべき資質・能力を明確にし、それに応じた教育内容と方法を考える」ことであった。しかし、従来の各教科の内容は削減しない方針を当初から示すなど、矛盾した設計が行われた。情報教育に関しても、既に中学校で必須になっていたプログラミングを小学校や高校でも必須化することが目的とされ、後付けでプログラミング的思考(力)という資質・能力が定義された。本稿では、本来の基本方針に則り、情報教育で育成すべき情報活用能力や生きる力との関係で、情報科の存在意義や望まれる指導内容・方法を考察し、合わせてプログラミングの扱い方も再考する。

キーワード：ICT 問題解決力、資質・能力の3観点、情報の科学的な理解、詳細な vs. 体系的な理解、問題解決の縦糸・横糸モデル、プログラミング的思考、内容 vs. 指導法としてのプログラミング

1. はじめに～資質・能力の3観点の弊害

1.1 インプットとアウトプットの同一視

新高等学校学習指導要領(文部科学省 2018a)が告示され、情報科の目標、内容、指導方法等の指針が示された。今回の学習指導要領改訂は、教育課程編成の原理を、コンテンツ・ベースド(何を教えるべきかの議論を重視する)から、コンピテンシー・ベースド(育成すべき資質・能力の議論を優先する)へと変革した点の特徴だと言う。しかし、議論開始当初から従来の各教科の内容は削減しないという方針を示すなど、矛盾した議論展開がなされた。また、育成すべき資質・能力は、結局のところ学校教育法の学力の3要素の延長上で、知識・技能、思考力・判断力・表現力等、学びに向かう力と人間性等の3本柱だとされた。

そもそも学力の3要素は、(知識と技能を1つにまとめれば)従来から各教科共通で用いられてきた学力の評価観点に、ほぼ対応している。結果的に、学習成果(アウトプット)と、資質・能力を育成する教育課程や学習過程の構成(インプット)とを区別しない点に根本的な問題がある。料理に喩えると、できあがり期待する「味」「食感」「見栄え」をレシピに書くようなものであり、「○○が××になるように調理します」と言ったところで、料理の参考にはならない。レシピとは、材料

と調理法を示すもので、求める味、食感、見栄えはその結果として得られるものだという図式を無視している。実際、インプットとアウトプットとを区別しないことにより、コミュニケーション能力を高めるためにコミュニケーション活動を行う、問題解決力を育成するために問題解決活動を行う、だからアクティブラーニングが必要だという誤った認識が流布している。

1.2 観点の独立性の低さ

新学習指導要領の情報科の目標には、

情報に関する科学的な見方・考え方を働かせ、情報技術を活用して問題の発見・解決を行う学習活動を通して、問題の発見・解決に向けて情報と情報技術を適切かつ効果的に活用し、情報社会に主体的に参画するための資質・能力を次のとおり育成することを目指す。

として、

- (1) 情報と情報技術及びこれらを活用して問題を発見・解決する方法について理解を深め技能を習得するとともに、情報社会と人との関わりについての理解を深めるようにする。
- (2) 様々な事象を情報とその結び付きとして捉え、問題の発見・解決に向けて情報と情報技術を適切かつ効果的に活用する力を養う。
- (3) 情報と情報技術を適切に活用するとともに、情報社会に主体的に参画する態度を養う。

という3本柱が示されている。下線部には繰り返しが多く、「～を通して」(インプット)と「～を育成する」(アウトプット)に同じフレーズが出現する。正に、「～

2019年2月1日受付 2019年2月8日受理

1) 東京工業大学リベラルアーツ研究教育院

2) 江戸川大学情報教育研究所

という活動をすれば、「～が身につく」(やっているうちにできるようになる⇒できないのは学習者が悪い)という学習者に責任を転嫁する構図である。また、3観点に対応して3つの柱に分けているが、それらの独立性が低いから、同じフレーズが複数の柱に繰り返し出てくる結果になっている。結局、3つの柱に分けて示すことに何の意味があるのか分からない。

内容に関しても、知識・技能として

(ア) 情報やメディアの特性を踏まえ、情報と情報技術を活用して問題を発見・解決する方法を身に付けること。

とある一方で、思考力・判断力・表現力等としても

(ア) 目的や状況に応じて、情報と情報技術を適切かつ効果的に活用して問題を発見・解決する方法について考えること。

とあり、類似したフレーズの繰り返しである。両者の違い(下線部以外)に着目すると、「情報やメディアの特性」に着目して活用するのが知識・技能、「目的や状況」に着目して活用するのが思考力・判断力・表現力等ということになる。ただし、後者には「適切かつ効果的」とあるので、「目的や状況」と「特性」とを対応づけて活用を考えるのが、後者かもしれない。しかし、そうだとすると、目的(良さ)や状況(制約)を考えずに特性だけを考慮して活用することも可能だと捉えていることになるが、その解釈は受け入れがたい。

学習指導要領解説(文部科学省 2018b)も、内容アとイの(ア)、アとイの(イ)という括弧がついた内容項目ごとに一括りで解説している。そしてアの「理解させる」に対し、イは「(生徒が)～を考える」から、教師を主語にした「～する力を養う」と言い換えただけである。指導事例も、アとイを区別して指導することが

不可能なためか、全を一括りで示している。

中央教育審議会(2016)答申は、情報活用能力も3観点に基づき見直した(表1)。知識・技能には、情報の科学的な理解と情報社会に参画する態度とが混在し、機器操作の技能も含まれた。思考力等は情報活用の実践力に、学びに向かう力は(より良い)情報社会(の構築)に参画する態度にそれぞれ対応しているように読めるが、後者には影の部分を考えるという側面が欠落している。また、全教科が3要素を指導し、分担や統合は意識されていない。教科目標も情報活用能力も同じ3要素にしたため、教科の指導をしていれば自ずと情報教育になるとの誤解を助長するリスクも高い。

一方、1997年の定義(文部科学省2002)は、情報活用能力を「生きる力」の下に位置づけ、情報活用の実践力、情報の科学的な理解、情報社会に参画する態度に分解した。その上で、各教科・学校段階の役割(分担や統合化)がそれらと対応づけて検討され、技術・家庭科や情報科の意義が明確にされていた(図1)。

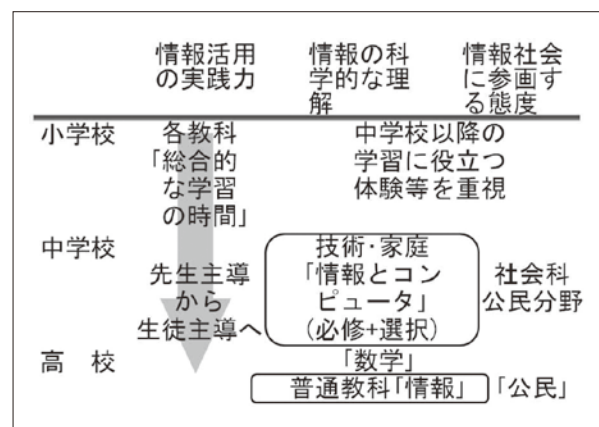


図1. 情報活用能力を系統的に育成する枠組み

表1 3観点で見直した情報活用能力と共通教科「情報」の目標(下線の種類で旧3本柱との関係を示している)

観点	共通教科「情報」の目標	情報活用能力を構成する資質・能力
知識・技能	情報と情報技術及びこれらを活用して問題を発見・解決する方法について理解を深め技能を習得するとともに、 <u>情報社会と人との関わり</u> についての理解を深めるようにする。	情報と情報技術を活用した問題の発見・解決等の方法や、 <u>情報化の進展が社会の中で果たす役割や影響</u> 、 <u>情報に関する法・制度やマナー</u> 、 <u>個人が果たす役割や責任等</u> について、 <u>情報の科学的な理解</u> に裏打ちされた形で理解し、 <u>情報と情報技術を適切に活用するために必要な技能を身に付けていること。</u>
思考力・判断力・表現力等	様々な事象を情報とその結び付きとして捉え、 <u>問題の発見・解決に向けて情報と情報技術を適切かつ効果的に活用する力を養う。</u>	様々な事象を情報とその結び付きの視点から捉え、 <u>複数の情報を結びつけて新たな意味を見出す力</u> や、 <u>問題の発見・解決等に向けて情報技術を適切かつ効果的に活用する力を身に付けていること。</u>
学びに向かう力・人間性等	情報と情報技術を適切に活用するとともに、 <u>情報社会に主体的に参画する態度を養う。</u>	<u>情報や情報技術を適切かつ効果的に活用して情報社会に主体的に参画し、その発展に寄与しようとする態度等を身に付けていること。</u>

結局のところ、3観点で示したことは、無駄な記述を増やし、分かりにくさを増大させ、教師や生徒に負担と責任を押しつけたに過ぎない。全ての教科が同じ3観点で示された結果、育成すべき資質・能力の総体や各教科の関連性も見えなくなり、カリキュラム・マネジメントの負担や困難度を増やしただけである(図2)。原因は、資質・能力を要素に分解して各教科の役割分担を考える(図1)のと、特定の要素に対応づけて後始末を現場に丸投げすること(図2)の違いにある。

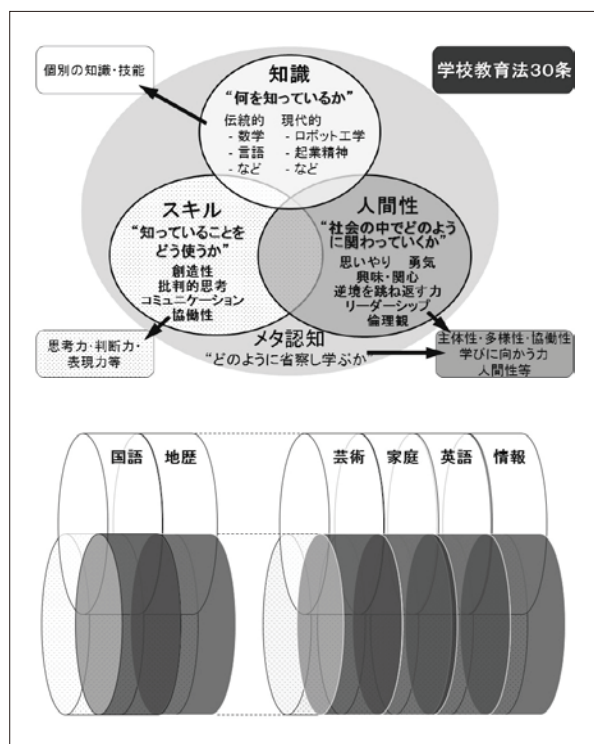


図2. 各教科の3要素は下のように教科独立なのか?

2. 育成すべき資質・能力とプログラミング

中央教育審議会答申にも引用された、小学校段階における論理的思考力や創造性、問題解決能力等の育成とプログラミング教育に関する有識者会議(2016)の報告は、プログラミング教育で育成すべき資質・能力も、以下のように3要素で示した。どの学校段階でも3要素の指導が必要との見解である。

知識・技能:

- (小)身近な生活でコンピュータが活用されていることや、問題の解決には必要な手順があることに気付くこと。
- (中)社会におけるコンピュータの役割や影響を理解するとともに、簡単なプログラムを作成できるようにすること。

(高)コンピュータの働きを科学的に理解するとともに、実際の問題解決にコンピュータを活用できるようにすること。

思考力・判断力・表現力等: 発達の段階に即して、「プログラミング的思考」(自分が意図する一連の活動を実現するために、どのような動きの組合せが必要であり、一つ一つの動きに対応した記号を、どのように組み合わせたらいいのか、記号の組合せをどのように改善していけば、より意図した活動に近づくのか、といったことを論理的に考えていく力)を育成すること。学びに向かう力・人間性等: 発達の段階に即して、コンピュータの働きを、よりよい人生や社会づくりに生かそうとする態度を涵養すること。

一方、筆者は、プログラミングは教育内容としての側面と学習活動(指導法)としての側面があり、学校段階や場面に応じて、以下のように導入目的を分けて考える必要があると考えている(松田・金井 2017)。

- ①将来の職業選択を考える機会を提供⇒小学校
- ②コンピュータが動作する原理や自動化のメリットを理解⇒中学校
- ③プログラミングの知識・技能を習得⇒専門
- ④アルゴリズム的思考を習得⇒数学
- ⑤情報社会に参画する態度に結びつく情報の科学的な理解を養う⇒情報科

端的に言えば、小学校段階のプログラミング指導は、学びに向かう態度(興味・関心を高める以前に、食わず嫌いを作らないこと)に焦点を当てればよい。また、教育内容としてのプログラミング(=プログラミング教育)は、高校の専門教育(か、各学校段階の特別活動や社会教育)で行うべきと考えている。なお、高校の共通科目にある選択科目は専門教育的側面もあるとして、前回改訂で普通教科から共通教科に名称変更された。よって、情報Ⅱで行うプログラミング教育は、③に位置づく。②、④、⑤では、指導法としてプログラミング活動を取り入れる余地があるが、指導法である以上、他の選択肢もあり、必須であるとは考えない。

上記をふまえて、もう一度、有識者会議の報告を読み返すと、全ての学校段階でプログラミングを必須にする理由は、もっぱら、プログラミング的思考を育成するためだと解釈される。小学校と高校の知識・技能や学びに向かう力・人間性等は、プログラミングを必須とする根拠にならないと読めるからである。そうだとすると、プログラミングはこれらの目標を達成するための指導法として位置づけるべきであり、指導法であるなら、それを使うかどうかは教師に選択の余地を残すべきである。

3. 情報科におけるプログラミングの位置づけ

3.1 情報科の存在意義

情報活用能力を3要素で見直した結果、何を情報科が担うのか、情報科の存在意義がどこにあるのかは、かえって見えにくくなった。表1を見れば明らかなように、必然的に、情報科の目標は情報活用能力の一部に限定される。全体として、従来の情報科で重視されてきた情報の科学的な理解をふまえた情報活用の実践力や情報社会に参画する態度の育成という趣旨が見えなくなっている。そもそも、情報社会に参画する態度がほとんど無くなっているし、どこが情報の科学的な理解なのか不明確である。

筆者は、全ての生徒のための情報科の目標は、正に、「情報の科学的な理解をふまえた情報活用の実践力(問題解決力)や情報社会に参画する態度の育成」だと考えている。その際、情報の科学的な理解をふまえることで、問題解決の質(効率や判断の信頼性等)が高まることを認識させ、日常的に情報技術を自己学習できる力を養う必要がある。高校の教育課程において、3つの柱を統合する唯一の場が情報科であり、そこに情報科の存在意義があると考えている。

なお、「情報の科学的な理解」とはそもそも何か、についてもここで問題にすべきである。詳細は、それを考察している松田(2018)に譲るが、コンピュータや情報技術の詳細な動作メカニズムを理解させ、ブラックボックス化の解消を目標にするのは間違いである。そもそも、専門家ですらどこかの段階から先はブラックボックスとして理解しているし、ブラックボックス化は技術を発展させる鍵だからである。例えば、ある関数を使った時にどんな副作用を起こすかをふまえて使わなければいけないとしたら、それは質の悪い(使えない)技術だと言える。実際、〇〇年問題などのトラブルを起こしてきたのはこの種の関数であり、その原因は技術者の手抜きである。これを解決するのに必要なのは技術者倫理教育であって、情報教育ではない。

3.2 プログラムとコンピュータの特性

先の有識者会議の報告は、「プログラムによってコンピュータに意図した処理を行うよう指示できる」と強調している。しかし、プログラムはそれ単体で動作するわけではなく、データを解釈しながら動作する。データが、プログラムを組んだ人の想定範囲内に収まっていれば、意図した通りに動くかもしれないが、そうでない場合には何が起るか分からない。漢字変換ソフトやAIプログラムのように学習機能を持つプログラ

ムは、勝手に学習し、賢くなることを意図しており、「学習してくれること」は想定内でも、「何をアウトプットするか」は想定できない。ユーザが想定外の使い方(例えば、誤った単語の区切り)をすれば、学習データが歪み、ありえない(意図しない、あるいは期待外れの)結果を生むかもしれない。

結局、「プログラムは設計した人の意図通りに動く」と理解させることで、むしろコンピュータの特性を誤解させる恐れがある。プログラムにはバグがつきものだし、悪意のあるデータを食わせてプログラムを誤動作させ、不正侵入するクラッキング手法もある。下手なプログラミングの指導は、筆者の考える目標達成(情報の科学的な理解をふまえた情報社会に参画する態度の育成)の妨げにもなりうる。

そもそも、コンピュータのメモリ上では、データとプログラムの区別は曖昧である。AIプログラムでは、学習結果をデータに反映させる手法もあれば、データ自体をプログラムとして動作させる手法もある。コンピュータの特性は、このような自由度にあり、それが多様な代替案を生むというメリットと、リスクを高めるというデメリットの両面をもたらし。金井・松田(2018)は、以上の立場から、情報システムを社会に導入する際、その社会的リスクや個人単位で行える防衛策を考える力を育成するために、プログラミングではなく、情報システムの設計を仮想的に体験するゲーミング教材を開発し、その効果を検証している。

3.3 どんな特性を何のためにどう理解すべきか

コンピュータに限らず、どんな道具(技術)も、与えられた選択肢を適切な順序で選び、正しく使わなければ目的の成果は得られない。プログラムの特殊性は、それを逐次的に(状況を確認しながら次の指示を)行うのではなく、予め全行程を指示し、自動実行できる点にある。その意味で、プログラムの思考の定義は不正確であり、誤解を与える。仮に、プログラミング教育の汎用性・転移可能性を強調する目的でこの特殊性を省くなら、そのような能力は他の教育でいくらでも育成できる。コンピュータを使うにしても、後述するように、PowerPointのアニメーションを作成するレベルで十分である。

上記の特殊性は、事前にあらゆる事態を想定し、対処法を準備することを求める。これがプログラミングの難易度を上げ、3.2に述べた想定外を生む。つまり、プログラミングの本質を理解するには、短時間で指導できるような簡易な(低次元の)プログラムを作成しても意味がない。あらゆる事態の想定が困難であることをふまえて、フェールセーフの考え方を取り入れ、ど

う信頼性・安全性を担保するか、実現方法も関連づけた理解を要する。

ただし、一般人と専門家では、求める理解の程度や仕方は当然異なる。一般人なら、ホワイトリストとブラックリストの違いが、フェールセーフの考え方や方法の類題になる。つまり、事例や観点を知れば十分であり、自ら実現に責任を負うことはない。時間をかけずに、プログラミングの困難さの本質を理解させるなら、プログラムを組む作業よりも、より上位のシステム設計の体験を仮想的に提供の方が効果的だと考えている。筆者らが開発した教材では、前述の学習機能を持ったプログラミング手法をとれ入れた場合の影響についても考えさせている。

情報科にプログラミング教育を入れるべきだとする人たちは、結局、現状でも消化できていない学習指導要領の内容を削減もせず、新たな内容を追加するという愚行を犯している。ブラックボックス化を許容することを前提として、どこまでを最低限理解することが市民に求められるのか、その境界を明らかにすることこそ、情報科教育学が取り組むべき課題である。その責務を果たさずに、プログラミングを入れることや入試に出題しやすくすることを優先するのは、生徒を置き去りにした議論であり、技術者倫理ならぬ教育者・研究者倫理が問われる問題である。

なお、松田(2017b)では、上の課題に対する答として、問題解決の縦糸・横糸モデルの情報科版を提案し、その中で、覚えるべき内部知識と参照すればいい外部知識とを切り分けている。内部知識としては、例えば、情報技術の理解枠や、問題解決の良さと情報技術の特性との関係(1.2で述べた目的と特性とを関連づけるルールに相当)、情報システムの典型例(トラブル、原因、影響、対策)などを位置づけ、自己学習する力をつけることを目指している。言い替えると、自己学習に必要な知識か否か、が内部知識とする1つの基準となっている。逆向き設計の理論を提唱している Wiggins and McTighe(2012)は、重要な観念を言わば内部知識に位置づけているが、その選択基準は不明確であり、経験則に依存しているように見える。松田の提案は、思考・学習のモデルを明示することで、その基準を明確にしようとするものである。

3.4 算数科や理科におけるプログラミング活動

3.3で、プログラムの思考の育成は、他教科でいくらでもできるし、PowerPointのアニメーションを作成するレベルで十分であると述べた。ここでは、文部科学省(2018c)の手引に示された小学校算数科における指導例を批判的に検討し、新たな提案を行う。また、

最近、理科教育法の授業で学生が作成した高校理科「探求活動」の指導計画に、プログラミング活動を取り入れて改善した案を紹介する。

3.4.1 算数科の事例と新たな提案

手引きに示された例は、「正多角形をかく」場合について考えるというものである。端的に言えば、

- 正三角形をかくという課題に対して、「辺の長さ」と角の大きさが全て等しい」「円に内接する」「中心角の大きさが全て等しい」など、正多角形の意味や性質を使って紙の上で作図する場合に倣って、コンピュータで作図する場合も考える。
- 「長さ100進む」「左に120度曲がる」といったコンピュータが理解できる命令を組み合わせて正三角形をかくことができる。
- 長さを変えて、大きさの異なる正三角形をかく。
- 「長さ100進む」「左に120度曲がる」を3回記述するのではなく、これらを「3回繰り返す」と記述する方法を知る。
- 三角形以外の多角形をかく。

という具合に学習を進めることが想定されている。手引きには、「数学的な見方・考え方」を働かせながら、……意図した一連の活動に対して、……「必要な動きを分けて考える」「動きに対応した命令にする」「それらを組み合わせる」「必要に応じて継続的に改善する」といった試行錯誤を行う中で、プログラミング的思考を働かせている、と述べられている。

しかし、この活動のどこが「数学的な見方・考え方」を働かせた活動なのか、また、それとプログラミング的思考とはどう区別されるのか、が明確に分かる説明は無い。そもそも、紙の上に正多角形をかく活動(例えば、定規で長さ5cmの線分ABをかく。分度器でBから60度の角度の点Cに印をつけ、定規でBC上にBC'=5cmになる線分をかく。定規を使ってB'とAをつなぐ。)では、プログラミング的思考は必要無いのかも明確に書かれていない。さらに、プログラミング言語を知らない児童にとって、「長さ100進む」「左に120度曲がる」「3回繰り返す」といった命令を教えるのか教えないのかも明確に書かれていない。教えるならば、この三角形をかく例を使って教えるのか、それとは異なる例で教えるのかも明確に書かれていない。

本稿で提案するのは、PowerPointを活用した指導例である。上述の指導例は、少なくとも正多角形の意味や性質を学んだ後に行うものであり、それらを「実際に図形をかくという活動に応用できる」ことが学習目標だと解釈される。これは、Bloom et al.(1956)の教育目標の分類学に当てはめれば、下から3段階目の

「応用」レベルの目標である。一方、本稿で提案するのは、より上位の「分析」「総合」段階や、「評価」段階までの達成を見通した指導例である。「評価」段階は、現実的な問題解決で、数学的な知識を活用することの有用性をより強く体験できる課題を提供することで達成できる。また、単に、「繰り返し」を使うといったコーディング・テクニック(特定の命令の活用)の工夫ではなく、より多様な数学的知識の活用方法を考えさせることで「分析」「総合」レベルの目標が達成できる。

まず、WordやPowerPointの作図機能を使うことと、Scratchなどのプログラミング言語を使って作図することと、どちらが現実世界で頻繁にあり、有用性が高いだろうか。後者は、基本的にプログラミングの世界に閉じた課題であり、資料の作成やプレゼンなどの日常的問題解決には役立たない。これが、PowerPointの活用を提案する理由である。プログラミングでしか役立たない数学的な知識なら、そんな知識は一般人には役立たないと評価する方が自然である。しかし、一般人が日常的に行う活動の中で、数学的な知識が必要になるなら、それは学ぶ価値があると自覚できる。これが、評価レベルの目標達成に影響を及ぼす。また、学びに向かう力の育成と密接に関連している。

図3は、実際にはアニメーション機能を使って課題提示する。与えられた1本の線分から、複製機能や書式設定の回転角指定機能を使って、多角形のかき方を説明するアニメーションを作成する(左上)。アニメーションとして実現するから、線分で描画しなければならないが、逆に、アニメーションを実現するには、図形を要素に分解し、各要素に適切な順序で動作を指定すればよいことが理解できる。これが、プログラミング的思考の本質であり、身近な活用例になる。右上は、多角形を途中まで作成し、それを複製して線対称に移動させることで、図形を完成させる例である。また、左下は、正三角形を複製し左右反転させた後、平行移動させることで、正六角形を作っている過程を示している。どれだけ平行移動したらよいかを考え、頂点から底辺に引いた角の2等分線が、もう1つの三角形の頂

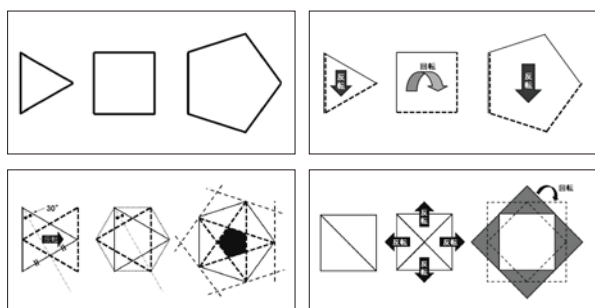


図3. 正多角形の知識を活用したアニメーション作成

点と重なるように移動すればいいことに気づく。

右下の例は、正方形が4つまたは2つの直角二等辺三角形で構成されることや、4つの直角二等辺三角形を外側に反転させることで、面積2倍の正方形になるという性質を説明するアニメーションである。このように、正多角形を線や図形に分解し、さまざまな性質を考え、表現する活動をすることで、数学的知識の体系化(分析)や、総合的な活用を指導できると期待される。

3.4.2 理科「探究活動」の指導案

本学の理科教育法の授業では、毎年、高校理科「〇〇基礎」科目の探究活動の指導案を書かせ、模擬授業をさせている(松田ほか 2018)。2018年度は、物理基礎で学習する「 $x-t$ グラフや $v-t$ グラフから、それがどんな動きを表現しているか、実験して確かめよう」という探求活動例が提案された。

これに対して、筆者が指摘し、助言した内容は以下の通りである。まず、 $x-t$ 、 $v-t$ グラフだけでは、直線上の動きしか表せず、現実実験できる運動としては、2次関数レベルのものに限られるから、探求活動としてはあまり深まらないと予想される。また、生徒役の中には、あえて、 t 方向にグラフの線が行ったり来たりする誤った例を挙げている者もいたが、時間軸が戻らないこと(1つの t に対して複数の x や v が存在することは無いこと)は実験以前に気づかせないといけない。

そこで、この活動をより発展させ、理科で学んだ知識を日常生活に結びつけて活用させる体験をさせるなら、2次元平面や3次元空間での動きをシミュレーションするように、 $x-t$ 、 $y-t$ 、 $z-t$ グラフや、 v_x-t 、 v_y-t 、 v_z-t グラフで動きを表現させるとよい。そして、実際に動作を可視化するには、プログラミングにより、 $x-t$ あるいは v_x-t グラフを画面上またはモデルカーの動きとして実現してみるのがよい。実際、それは自動運転技術やゲームの製作などに利用できるものであり、STEM教育へと発展させられる可能性がある。

4. 技術・家庭科と情報科が担う役割

松田(2017b)の縦糸・横糸モデルとの関係で述べれば、中学校技術・家庭科では、各領域で問題解決の手順を学び、それを縦糸・横糸の手順として理解する。また、コミュニケーションや計測・制御のプログラミング活動を行う中で、より良く問題解決するには、「情報の活用」「多様な良さ」「トレードオフとその解消」「多様な代替案」「良さに応じた選択」「意思決定の権利と責任」「転ばぬ先の杖」「合理的判断の知識」など、情動的な見方・考え方等の活用が必要であることを手

順と関連づけて指導する。見方・考え方のメタな理解を促すために、他の技術領域、他の教科でも見方・考え方を指導することが望ましい(松田 2018)。

ちなみに、プログラミング的思考の「自分が意図する一連の活動_a」を実現するために、どのような動き_bの組合せが必要であり、一つ一つの動きに対応した記号_cを、どのように組み合わせたらいいのか、記号の組合せ_dをどのように改善していけば、より意図した活動_eに近づくのか、といったことを論理的に考えていく力」という文言の下線部を以下のように書き換えてみる。

- a. ものづくり, エネルギー変換, 解の導出/証明
- b. 材料とその加工, 機械・電気部品, 定理・公式
- c. 作業工程, 部品, 変数や値
- d. 作業工程, 動作メカニズム, 式変換の論理
- e. 材料や作業の無駄解消, エネルギーの効率的な活用, 見落としや間違いの無い論理

つまり、このような思考は、あらゆる科学・技術の本質的思考であり、プログラミングを通じてでなければ養えないものではない。しかし、これらの共通性、背景にある汎用的な思考の枠組みをこれまで明示的に指導してきたかについては疑問が残る。転移可能な汎用的思考力・判断力・表現力等を育成するという観点から、新たな教科の指導内容・方法を考えることこそが、これからの教育に本質的に求められている。

一方、縦糸・横糸モデルでは、領域固有知識はフレーム形式で5W1H等のスロットを持つと考えており、プログラム化の目的(Why)として自動化があり、それがコスト削減や可用性を高める等のMeritに寄与する一方で、Demeritとして安全性や透明性を損なう恐れがあることを理解させる。Whenとしては当該知識を代替案発想過程で主に活用すること、Whereとしては計測・制御や対話インタフェース、シミュレーションなどのケースで当該技術がよく活用されることを関連づける。Whatは辞書の定義であり、Whoはプログラムを作る人、使う人が持つ権利と負うべき責任、受ける影響などを関連づける。また、下位知識としては、プログラミングの方法論として、アルゴリズム型、ルールベース型、機械学習型などの分類を覚え、それぞれのメリット/デメリットやトレードオフ関係の理解へとリンクを張るように促す。

知識は単に覚えるレベルから、簡単な演習問題に回答できるレベルまでが、個々のスロットの知識の理解・定着を図る段階である。それらを問題解決の手順や、見方・考え方と関連づけ、活性化し、他の技術等とも関連づけ、社会的影響なども考え、トレードオフ解消策を考えられるようにするのが、高校の情報科の役割であり、分析・総合・評価レベルに該当する。

「情報の科学的な理解をふまえた情報活用の実践力

(問題解決力)や情報社会に参画する態度の育成」を重視する立場から、筆者は、現状の情報科目では、「社会と情報」の方が、「情報の科学」よりも、全ての生徒が学ぶべき情報科の科目として適切だと考えている。もちろん、「社会と情報」の目標を達成していれば、「情報の科学」を選択すればよい。しかし、情報の科学的な理解に裏打ちされた情報社会に参画する態度を育成するという点で、中学校の技術・家庭科だけでは時間的にも不十分であるし、社会科や公民科では、下線部が保障されない。

その意味で、多くの人は「情報I」を「情報の科学」に対応づけて理解しようとするかもしれないが、あえて、「社会と情報」と対応づけて理解することが重要である。実際、学習指導要領解説(文部科学省 2018b)を読むと、情報Iの(1)(2)(4)は、「社会と情報」と重なる部分が多い。

情報I (1) ← 社会と情報(1)(3)
 (2) ← (1)(2)
 (4) ← (2)(3)(4)

また、情報I(3)のプログラミングやモデル化とシミュレーションについては、既に2節で述べてきた通り、⑤の指導における教育方法として位置づけることが適切である。そもそも、プログラミングは魔法の杖では無く、プログラムを組まなくてもモデル化やシミュレーションはできる。モデル化は既存教科で指導すべき範囲であり、プログラミングは、時間的制約で不可能だったシミュレーションを作業の効率化で可能にしているに過ぎない。以下にその例を示す。

筆者は、シミュレーション&ゲーミング学会に所属しており、この分野では、ありとあらゆる分野のモデル化とシミュレーションが研究されている。ただし、多くの研究は、プログラミングを用いず、カードゲームやボードゲーム的な道具が用いられる。ここで紹介するのは、大沼(1997)の「廃棄物処理」ゲームである。このゲームの開発目的は、教育的利用ではなく、社会心理学的立場から環境問題を考えることにある。以下の下線部を除いた部分がオリジナル・ルールである。

- ①：5～6人で1グループ。
- ②：各自4枚の産業廃棄物(カード)を受け取る。
 (廃棄物は、安全：危険=3：1の割合で24枚。)
 ←③-0：全員が1ポイント抛出(-1ポイント)
- ③-1：毎回1枚を処理(原則として裏にして出す)。
- ③-2：危険物を安全に廃棄するには、カードを表にして出し、8ポイントを支払う。
- ④-1：裏にして出されたカードが「疑わしい」と思った人は、2ポイント払えば検査できる。危険物なら廃棄した人は-20ポイント。

←④-2: 誰も検査しなければ、サイコロを振る。和が5か6なら抛出金から2ポイント払って検査し、危険物なら廃棄した人は-20ポイント。

←④-3: 抛出金不足になったら③-0と同様に抛出。

⑤: ③-1に戻り、カードが全員無くなるまで続ける。

⑥: ゲーム終了後、不法処理された危険廃棄物があれば、1枚につき全員が4ポイント減点される。

下線部は、筆者が独自に考えたルールである。各グループには、最初に、下線部のルールを導入するか、オリジナル・ルールのまま行うかを選択させる。その後、プレイを行い、各グループの結果を比較する。筆者がある授業で試行したところ、オリジナル・ルールで行ったグループは、危険廃棄物の不法処理が多数発生し、結果的に全員が12点を追加減点された。一方、下線部のルールを導入したグループは、全ての危険廃棄物が安全に処理された。つまり、グループ全体の減点は、圧倒的に新ルールの方が少なかった。

さて、上述の2グループのみに基づく結果で、ここから何らかの結論を出すことは妥当であろうか。多くの人は、もっとシミュレーションを繰り返すことを要求するだろう。その際、同じグループでの繰り返しや、より多様な人で構成されたグループでの試行を求めよう。ここに、シミュレーションを行う上での「良さ」の検討があり、それを実現する方法の工夫が必要になる。人を集められるか、時間的、空間的制約の克服も考える必要がある。ネットワークを介してゲームを行えるようにするという工夫もあれば、人の特性を何らかの形でモデル化し、プログラム等で処理する工夫もある。ここに、情報技術を用いる意義があるのであって、元のゲームのルールを考えることが情報科の目標ではない。

5. まとめ

本稿では、最新の学習指導要領のベースになっている考え方を批判的に検討し、情報活用能力や共通教科「情報」(情報科)の内容構成に関する疑問点を指摘した。新たな学習指導要領は、学校現場や教員に混乱を招く恐れがあり、それを回避するために、どんな点に注意をして指導すべきか、筆者の考えを明示した。特に、プログラミング的思考という概念の危うさや、プログラミング教育が機器操作教育と同様の過ちを犯さないための取り組みについて、提案を行った。

参考文献

Bloom, B. S., Engelhart, M.D., Furst, E.J., Hill, W.H., and Krathwohl, D.R.(1956) Taxonomy of

educational objectives: The Classification of Educational Goals, Handbook I: Cognitive domain. David Mckay

中央教育審議会(2016) 幼稚園, 小学校, 中学校, 高等学校及び特別支援学校の学習指導要領等の改善及び必要な方策等について(答申), 文部科学省

金井文哉・松田稔樹(2018) 情報システムの設計・特性・社会的影響の相互関係を理解させるゲーミング教材の開発, 信学技法, 117(469), 125-130

松田稔樹(2017a) 機器操作能力から問題解決力へ: 情報教育の課題と展望, Informatio, 14, 3-12

松田稔樹(2017b) 情報科で育成すべき問題解決力と思考・判断・表現方法の指導, Informatio, 14, 43-54

松田稔樹(2018) 「情報の科学的な理解」の本質をふまえた情報科の指導のあり方, Informatio, 15, 3-13

松田稔樹, 金井文哉(2017) 共通教科「情報」でプログラミングを扱う意義とそれに即した教材開発の方向性, 日本情報科教育学会第10回全国大会講演論文集, 109-110

松田稔樹, 鶴田拓真, 水野佑美(2018) 課題学習・探求活動の指導力向上を支援する仮想授業ゲームのための指導案分析, 日本シミュレーション&ゲーミング学会全国大会報告集, 2018年春号, 90-95

文部科学省(2018a) 高等学校学習指導要領. http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/07/11/1384661_6_1_2.pdf

文部科学省(2018b) 高等学校学習指導要領解説・情報編. http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/07/13/1407073_11.pdf

文部科学省(2018c) 小学校プログラミング教育の手引(第二版), http://www.mext.go.jp/component/a_menu/education/micro_detail/_icsFiles/afieldfile/2018/11/06/1403162_02_1.pdf

文部科学省(2002) 情報教育の実践と学校の情報化. http://www.mext.go.jp/a_menu/shotou/zyouhou/020706.htm

大沼進(1997) ゲームコラム9: 廃棄物処理ゲームと二次的ジレンマ. In 広瀬幸雄, シミュレーション世界の社会心理学: ゲームで解く葛藤と共存, ナカニシヤ出版, 149-152

小学校段階における論理的思考力や創造性, 問題解決能力等の育成とプログラミング教育に関する有識者会議(2016) 小学校段階におけるプログラミング教育の在り方について, 文部科学省

Wiggins, G. and McTighe, J.(著) / 西岡加名恵(訳): 理解をもたらすカリキュラム設計. 日本標準(2012)