

男川っ子の論理的思考力の向上を目指して

～レッツ・プログラミング！ 全教科領域におけるプログラミング学習の展開を手立てに～

◇ 本年度の主題について

第5期科学技術基本計画（H28～H32）において、我が国が目指すべき未来社会の姿としてサイバー空間（仮想空間）とフィジカル空間（現実空間）を高度に融合させたシステムにより、経済発展と社会的課題の解決を両立する人間中心の社会 Society 5.0 が初めて提唱された。Society 5.0 で実現する社会は、IoT（Internet of Things）で全ての人とモノがつながり、様々な知識や情報が共有され、今までにない新たな価値を生み出すことで、世代を超えて互いに尊重し合あえる社会、一人一人が快適で活躍できる社会の実現を目指している。未来を生きる子供たちにとって、社会に溢れる様々な情報を主体的に捉えながら、自分にとって何が重要なのかを判断するとともに、見出した情報を活用しながら他者と協働し、新たな価値の創造に挑んでいけるような、情報活用能力を身に付けることは極めて重要である。2020年度から小学校で完全実施される新学習指導要領では、「主体的・対話的で深い学び」の実現による真に自ら学ぶ力の育成が求められている。その基盤となるのが情報活用能力や言語能力、問題発見・解決能力である。全教科領域におけるICT活用は特別なものではなく、子供たちが自らの学びを深める手立てとしてごく普通に使いこなすスキルや世界に通用する「論理的思考力」の育成が最重要課題となっている。それに対応し、本校は、昨年度より「論理的思考力を育む授業づくり」に取り組み、思考力、判断力、表現力を関連付けて総合的に活用し、子供たちが自ら未来を切り拓いていく力の育成に努めてきた。新しい課題に対応するための「はじめの一步」として、教員研修や子供たちのプログラミング体験、教科学習におけるプログラミングの試行など、小学校におけるプログラミング教育の導入モデルを構築し、育てたい論理的思考力を明確にした。本年度は、全学年でプログラミングを取り入れて授業実践を重ね、小学校での「プログラミング学習年間計画」を構築する。「レッツ・プログラミング」をキーワードにして、全学年の日常的な教科学習において多様なプログラミング学習の展開を実現したい。さらに、アウトプットの段階におけるプログラミング学習用ロボット教材の有効性についても検証していきたい。

1 プログラミング教育のねらいは？

プログラミング教育のねらい（プログラミング教育の手引き 第1版 2018年3月 文部科学省）

- ① 「プログラミング的思考」を育むこと
- ② プログラムの働きやよさ、情報社会がコンピュータ等の情報技術によって支えられていることなどに気付くことができるようにするとともに、コンピュータ等を上手に活用して身近な問題を解決したり、よりよい社会を築いたりしようとする態度を育むこと
- ③ 各教科等での学びをより確実なものにすること

◇教科のねらいを達成するためのプログラミング体験である⇒「**プログラミングで学ぶ**」

2 育てたい論理的思考力とは？

そもそも論理的思考力とは？

論理に基づいて思考する能力（の高さ）という意味で用いられる表現。 **道理や筋道に則って思考を巡らせて結論を導いたり、あるいは、複雑な事柄を分かりやすく説明したりできる能力**として主に捉えられる。英語の logical thinking（ロジカルシンキング）の訳語としても用いられる。

【日本語表現辞典 Weblio 辞書より】

まず、あちらこちらでいわれる「論理的思考」について、大まかに共通理解したい。特に「プログラミング的思考」と混在して使われているので、気を付けたい。もともと、指導要領で取り上げられる以前から、一般社会において「論理的思考の必要性」は必須とされていた。(上掲資料参照)

その上で、今回の指導要領で求められる「論理的思考力」を本校向けにまとめたのが右の資料の①から⑥である。物事を捉え、理解し、見通しをもって考え、問題解決を図る、そして、分かりやすくアウトプットするという表現力まで含めている。子供たちに身に付けさせたい「**論理的思考力**

の**正体**を、新指導要領ですべての学習の基盤として求められている「言語能力」「情報活用能力」「問題解決能力」まで包括するような力としてイメージしてはどうかと考えている。いっそう全教科領域での「表現まで含めた考える力」の育成が大切になる。

一般的に言われるプログラミング的思考を右に示してあるが、コンピュータに指示することが意識された内容になっていると考えられる。

<論理的思考力のとらえ>

- ①まとめて考える力
- ②物事を分けて考える力
- ③見通しをもって順番に考える力
- ④試行錯誤して改善する力
- ⑤他にあてはめて活用する力
- ⑥分かりやすく説明したり、伝えたりする力

<プログラミング的思考といわれる考え方の分類>

- ① 計算や作業を手順ごとに順序だてる(シーケンス) 考え方
- ② 手順のまとまりを繰り返して実行する(ループ) 考え方
- ③ 条件によって手立てや手順を切り替える(分岐) 考え方
- ④ ものごとを Yes/No の組み合わせで考える(真偽値) 考え方
- ⑤ ものごとの性質や手順のまとまりごとに名前を付ける(抽象化) 考え方
- ⑥ 試行錯誤して改善しながら作り上げる(デバッグ) 考え方

<プログラミング教育の手引き 第1版より >

A 学習指導要領に例示されている単元等で実施するもの

B 学習指導要領に例示されていないが、学習指導要領に示される各教科等の内容を指導する中で実施するもの

C 各学校の裁量により実施するもの(A、B及びD以外で、教育課程内で実施するもの)

D クラブ活動など、特定の児童を対象として、教育課程内で実施するもの

3 プログラミング学習の進め方

文部科学省が手引きで示した「プログラミング教育」の分類A~Dが小学校で取り組むべき内容になっている。論理的思考力を育む学習をどのように進めるか考えた。プログラミング学習というと、コンピュータを利用したプログラム作成に意識が行ってしまうが、それだけではない。コンピュータを使わない学習もある。それをアンプラグドプログラミングという。この学習方法は従来の学習の延長線上にあり取り組みやすいものである。

レッツ！プログラミング step 1 (次ページ図参照) で、全学年でアンプラグド型プログラミング(後述)を意識した学習を行い、その上で コンピュータを利用した学習を行っていくことがより高い効果が望める。

レッツ！プログラミング step 2 では アンプラグド型プログラミングに加え、各学年の発達段階を考慮した教科領域でのビジュアルプログラミングを使った学習を実施する。実際には、Viscuit (<https://www.viscuit.com/>)



やScratchなどのビジュアルプログラミング(ブロック型プログラム)により、平易で楽しいプログラミング体験をさせる。自分で作ったプログラムでキャラクターを動かすなどのプログラミングを楽しませることで、自己充実感を味わいたい。

そして、知らず知らずのうちにプログラムの概念を芽生えさせたいと考える。その上で、教科領域のねらいに合わせて、積極的にスクラッチなどを活用した授業を行いたい。(算数や図工、音楽、国語、理科などでスクラッチモデルプログラムあり・男川小学校での実践一覧参照)

スクラッチ以外にも、授業で使える、ビジュアルプログラミング(ブロック型プログラム)はいろいろ紹介されている。5・6年生では、プログル(<https://proguru.jp/>)実践⑥を用いて、ビジュアルプログラミングになれるとともに算数の「公倍数」「多角形」「平均値」の三つの単元で活用することで算数の学習内容の理解を深めることもできる。現場の教員と企業が協力して作りあげて公開された「プログラミングゼミ」というものもある。

(<https://programmingzemi.com/index.html>)スクラッチと同じような使い方ができる。これについては、平成30年度に試してみたい。ただし、本校の子供たちはスクラッチをすでに使いつつあるので、そちらで進めていけばよいと思う。低学年の使い勝手がどうかという点で、使ってみる価値はありそうだと考えている。iPadの導入に伴い、低学年ではスクラッチJrの利用も可能になる。ピョウンキーも利用でき、PC室パソコンと連動して教科学習に生かす使い方も模索できるだろう。

レッツ!プログラミング step 3では、アウトプットを画面上ではなく、実際にロボットを動かすようなフィジカルプログラミングを取り入れたい。ただし、中学校でのネットワークを使った双方向での複雑な制御や高校での専門教科にあたるコーディングをするのではない。あくまで、教科や総合的な学習の時間と連動して進める。例として、29年度に6年生が公開した「電気のはららき」での、「歩行者押しボタン信号」の制御や「モーターカー」(アーテック教材)がある。平成30年度は、これに加えて、iPadでのプログラミングも視野に入れて「Edison」や「Sphero」「micro:bit」など、青山学院の阿部先生や本校助言者の平井先生から紹介していただいた教具も使っていきたい。

以上のようにアンプラグド型プログラミング学習をベースにして、教科領域での様々なプログラミン学習を試行錯誤しながら、男川小学校としての「年間プログラミング学習一覧」が本年度中に全学年で形になっていくとよいと考えている。

ただ、実践を行うにあたって忘れていけないのは、大切なのはプログラムが上手に組めることではなく、すべての基盤として社会全体から求められている「論理的思考力」を育むことである。忘れてはいけないのは……。

大切なことは、各教科等での学びをより確実なものにすることであり、教科のねらいを達成するためのプログラミング体験であり、「プログラミングで学ぶ」ことである。

プログラミングを体験しながら論理的思考力を身に付けるための学習活動とは、子供たちが将来どのような職業に就くとしても、時代を超えて普遍的に求められる力としての「プログラミング的思考」の育成を目指すものであり、プログラミングのための言語を用いて記述する方法(コーディング)を覚え習得することが目的ではない。「**プログラミング的思考**とは、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み合わせが必要か、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うこと

【小学習指導要領解説 総合的な学習の時間編】