

いよいよ次期学習指導要領の具体的内容が公表されました。毎度のことながら、学習指導要領が改訂されると、新しい概念、用語、キャッチフレーズ等が出てきますから、教育現場の先生方にとっては準備や対応方法に神経を費やすことも多くなるかと思われます。

各教科に共通する用語として個人的に気になるものの一つに「プログラミング的思考の育成」があります。解説資料などを読むと、「決してコーディング指導そのものではない。」とか「いろいろな教科で取り組みが可能である。」という主旨のことが書かれています。文言の意味が曖昧だったり、矛盾含みの解説もあつたりしますが、簡単に言えば算数・数学科のみならず各教科で現実の活動場面をプログラミング的に考える事が望まれるということのようです。しかし、算数・数学の教師が効果的に対応するのは少々難しい話です。例えば、われわれの日常生活場面で、

(ア) 30個のリンゴの中から最も大きいものを見つける。

(イ) 30個のリンゴを大きい順に並べる。

(ウ) 30個のリンゴの重さの合計を求める。

といった仕事を比較するなら、おそらく(ア)が最も容易で、(イ)や(ウ)はけっこう手間がかかるでしょう。これに対して、パソコンのプログラミングで似たような仕事を考えてみると、

(ア) 30個のデータの中で最大値を求める。

(イ) 30個のデータを大きい順に並べる。

(ウ) 30個のデータの合計を求める。

という感じになるかと思います。これらの仕事をおこなうプログラムを書こうとすると、プログラム行数が最も少なく済むのは(ウ)です。むしろ(ア)や(イ)の方がはるかに多くの行数やテクニックを必要とします。

なぜでしょうか？実はプログラミング（コーディング）を少しでも経験した人ならそんなことは体で覚えています。時代とともにプログラミング言語はどんどん変化してきましたが、上記の仕事の範囲内では、代入文、繰り返し表現、条件分岐などの基本的コンセプトはほとんど変わっていないと言えます。それらを組み合わせてプログラミングをするので、日常生活の活動とのずれが生じるのです。別の例を言えば、「複数の写真から同じ人の顔を選び出す仕事は人間には至極簡単な仕事であるが、今日のようにスマホの機能として実装されるまでにどれほど多くの年月と技術を要したのだろうか」と思えばわかりやすいでしょう。プログラミング的思考の真のご利益を得るためには、やはり(人間の活動とは必ずしも同じではない)基本的なプログラミングの知識を教師が理解した上で、算数・数学でのいろいろなアルゴリズムの表現に活用しようという姿勢で取り組む必要があるでしょう。言葉の言い換えだけにならないようにしっかり準備や工夫をした上で新しい時代へ船出することにしましょう。