

踊るバズワード ～Behind the Buzzword(15)STEM教育(3)：

STEMを取り入れた「夏休みの自由研究」型パッケージ教育のすすめ

<https://eetimes.itmedia.co.jp/ee/articles/2109/28/news087.html>

今回は、STEMを取り入れた新しい教育を提案します。併せて、プログラミング教育×STEM教育の方程式から導き出せる、「理系日本人補完計画」という壮大な妄想(?)を語ってみます。

2021年09月29日 16時30分 更新

[江端智一, EE Times Japan]



「業界のトレンド」といわれる技術の名称は、“バズワード”になることが少なくありません。“M2M”“ユビキタス”“Web2.0”、そして“AI”。理解不能な技術が登場すると、それに“もっともらしい名前”を付けて分かったフリをするのです。このように作られた名前に世界は踊り、私たち技術者を翻弄した揚げ句、最後は無責任に捨て去りました——ひと言の謝罪もなく。今ここに、かつて“AI”という技術は存在しない」と2年間叫び続けた著者が再び立ち上がります。あなたの「分かったフリ」を冷酷に問い詰め、糾弾するためです。[⇒連載バックナンバー](#)

「夏休みの自由研究」の恐怖

子どもの頃の私は、「夏休みの自由研究」というものに困ったことはありませんでした。好きでした。ワクワクしていました。数に制限がなければ、多分、ひと夏で3本くらいはいけたんじゃないだろうか、と思います。

ええ、分かっています。この冒頭のフレーズだけで、このコラムを読んで頂いている多くの保護者の方と、そして子どもたちを一気に敵に回したことは、十分に認識しています。

「夏休みの自由研究」——この課題が、どれだけの子どもの夏休みの日々に暗い影を落しているのかを私は、よく知っています。最初に知ったのは、数年前、お隣の国韓国で、保護者が、大学生アルバイトを使って「夏休みの自由研究」の外注を行っていることが社会問題になっている、というニュースを見たときです。

その時は、『ふーん』くらいの感想しか持ち合わせていませんでしたが、その後、調べてみたら、我が国にも、この外注サービスを利用する保護者が(少なくとも3桁以上)いることが分かっており、そして、8月下旬になると、必ずこの話題のニュースが取り扱われていることに気が付きました。

どうやら我が国では、この「夏休みの自由研究」において、保護者は『共同研究者』である、とされているフシがあるようなのです。

[こちらの論文「小学生の保護者が子供の夏休みの自由研究に抱える不安」](#)によれば、保護者の9割が夏休みの自由研究に「不安あり」と答えているようです。その中でも、「親がどういったサポートをしたらよいか」(22.8%)、「学年にふさわしい出来栄かどうか」(12.9%)と考えているからです。そして、子どもの方も、保護者のサポートは当然である、と考えている様に見られます。

私を除く江端家におけるインタビュー結果は、おおむね、この論文の内容と同じでした。私を除く全員が「夏休みの自由研究は、宿題だから仕方なくやった」といい、嫁さんに至っては「親のサポートは折り込み済み」と、言い切りました。

ただ、江端家の場合は、ちょっとケースが特殊でして、安易に父親(私)に相談すると、『父親が勝手に暴走しかねない』——自分の娘の年齢や実力と似(そぐ)わない、研究レポートを勝手に完成させてかねない——ので、娘の方で、父親の「利用方法」を考えていたようでした。

この「暴走」の例としては、例えば——10年前に、娘(中学1年生)の文化祭で展示されていた自由研究の中に『声紋分析に関する個人差の調査』なる、そのレポートの中には、特殊な専用装置で測定されたと思えない声紋のハードコ

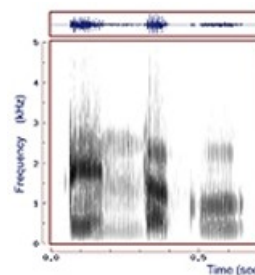
ピーが張ってありました。

中学1年生の研究が「声紋分析」とは…… — 『うん、この子の親、「親バカ」というより「バカ親」だ』と思ったことを、今でも覚えています。

それはさておき。

「夏休みの自由研究に、保護者のサポートは当然である」が当然であるかどうかはさておき、自由研究をサポートする側の保護者には、子供の要求に答えられるだけの理系の知識やスキルを持ち合わせていることが前提となります。

また、仮にそのような知識がなくとも、日本であれば、このような子どもの自由研究をサポートする書籍はあるんじゃないかな、と、図書館やAmazonで調べてみたら — 出るわ出るわ、宝の山状態です。これだけの資料があれば、「外注」なんぞ必要ないように思われます。



しかし、保護者は、子供の研究のサポートができるという自信は備わっていないようです。仮に研究ネタがあったとしても、「実際に実験をする」ことは、さらに何段もフェーズが高くなりますし、これを、研究報告とするなら、それなりの纏め方(背景と動機→課題→課題を解決するアプローチ→実験→実験データの集計→考察)を知っている必要もあります — うん、確かに面倒くさい。私でも「外注」に丸投げしたい、と思うかもしれません。

加えて、夏休みの自由研究をサポートする保護者は、主に母親であり、その母親は文系出身者が多いということも問題であるようです*)。

*) ちなみに、嫁さんに「なぜ"母親"はそんなに子どもの宿題を気にするのか?」と尋ねてみたら、『うーん、コミュニケーション時間とか、距離感が作りだす、"共感度"が、父親より大きいのかもじゃないなあ』と言っていました。

女性の理系比率が低いことはご存じの通りですが、そもそも、我が国では、科学の基礎的概念の理解度が欧米諸国と比較して低いレベルにあることが、データで客観的に示されています*)。

*) [「一般般市民の科学的リテラシーに関する分析と考察」](#)

いずれにしても、「夏休みの自由研究」が、これからも子どもと保護者の両方を、虐待し続けることは、確定的なようです。

□

こんにちは。江端智一です。

今回は、「踊るバズワード ～Behind the Buzzword」のシリーズ「STEM教育」の3回目です。

[前回](#)に引き続き、「プログラミング教育」と「STEM教育」についてお話していきたいと思います。今回はこのバラバラの2つの教育方針として見るのではなく、これを一つの連携システムとして考えてみたいと思っています。

そして、実は文部科学省の官僚の皆さんが、あの経済産業省の「太平洋ベルト地帯構想」や、旧通産省（現経済産業省）の「国産コンピュータ保護政策」にも匹敵する、すごいこと——『理系日本人補完計画』を考えているという、私の壮大な妄想を書きつづけてみたいと思います。

ただ、この妄想、私の中では、それなりに根拠はあります。今回は、この話をします。

度肝を抜く試験問題

まず、「プログラミング教育」について、前半を始めたいと思います。

第1回、第2回と連続して、我が国のプログラミング教育についての、解説と感想（というか、はっきり言って批判）をしてきました。例えば、こんな感じのことです。

既出:「プログラミング教育の義務化」について

江端仮説:「プログラミング教育は、さらに、『プログラミング離れ』を加速させるだけではないか？」

<https://www.kobore.net/diary/?date=20170120>

- (1) 多くの子どもが『プログラミングなんて、一生やるものか』という決意をする、なんてことにはならないだろうか——と。
- (2) プログラムというのは、どの部分をとっても、曖昧(あいまい)さの存在を絶対に許しません。
- (3) 数学と違って、本当の意味で、0点と100点しかない、冷酷にして残酷な世界です。
- (4) プログラミングとは、「論理的であることが望ましい」などという臍抜けた態度では成立しません。
- (5) (A)完全に論理的で、(B)病的なほどの注意と、(C)到底ありえないほどのケースまで想定した膨大な数のテストをくりかえすことが必要になるからです。

前回のコラムの脱稿後も、私は、論文やら資料やらを探しては、手当たり次第に読んでいたのですが、どうも、「プログラミング的思考」というものが分からずに、ずっと迷走していました。

私（江端）は、仕事やコラム執筆で、これまで、数行のスクリプトから、数万ステップのパッケージソフトまで、膨大な数のプログラムを書き続けてきたので、文部科学省のいう「プログラミング的思考」は、ボンヤリとは理解できます。

しかし、これは、無数のプログラミングという作業の経験から得られる思想（イデオロギー）であって、他人に伝達可能な「技術」であるとは思えないのです。何というか、言語で説明できないのです。これを説明することは、『スキーマの教本を読んだだけで、分かったような気になって、そのままスキー板を抱えてゲレンデデビューするような無謀さ』という感じがします。

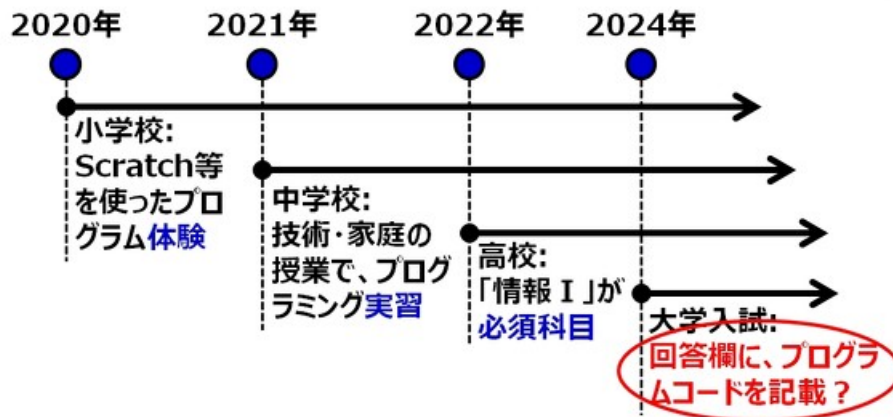
というわけで、今回のコラムでは、アプローチを変えることにしました。

あまり好ましい言い方ではありませんが、教育機関が、子どもに何を与えているのかを、もっとも露骨に示すイベントがあります——それは、定期テストであり、入学試験です。

教育機関は、子どもたちに、テストで高得点を得られるように訓練し……もとい、教育し、入学試験に合格させて、その子どもを別の施設に移転させる……もとい、入学させることで、その任を終えます——ですね？

さて、文部科学省の、今後の「プログラミング教育」のマイルストーンは以下のように予定されています。

「プログラミング教育」の実施スケジュール



どんな問題が出題されるのだろう？

私は、新教科「情報」において、どんな試験問題が出題されるのかを考えてみました。

まず、プログラミングに関する出題と言えば、私もかなり使った「情報処理技術者試験」だろうと考えました。

私は、「(今では廃止されている)情報処理技術者第1種」と、「ネットワークスペシャリスト」に合格しています。ーが、まあ、会社ではあまり評価対象となっていないようです(著者のブログ)し、私もそんな資格でエンジニアの「能力」は評価しません。でも、社会では「受験を試みたという『事実』」は評価されているようです。

情報処理技術者試験のプログラミングに関する出題パターンは、20年前からあまり変わっていないようです。

令和3年春期 応用情報技術者試験 午後

プログラミングに関する問題 (抜粋)

問3 クラスタ分析に用いる k-means 法に関する次の記述を読んで、設問 1～3 に答えよ。

k-means 法によるクラスタ分析は、異なる性質のものが混ざり合った母集団から互いに似た性質をもつものを集め、クラスタと呼ばれる互いに素な部分集合に分類する手法である。新聞記事のビッグデータ検索、店舗の品ぞろえの分類、教師なし機械学

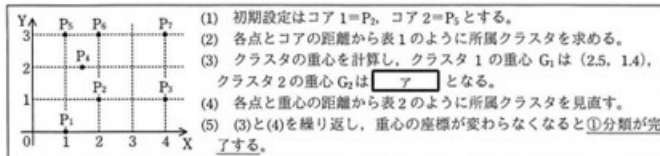


図 1 7 個の点の分類

```
function clustering (N, K) //N はデータ数, K はクラス数
    MaxCount ← 100 //無限ループを防ぐため最大見直し回数を設定
    initial(K) //初期設定
    for( s を 1 から N まで 1 ずつ増やす ) //クラスを決定 (1 回目)
        for( t を 1 から K まで 1 ずつ増やす )
            data_length[t] ← data_dist(s, core[t])
        endfor
        cluster[s] ← min_index(data_length)
    endfor
    for( p を 1 から MaxCount まで 1 ずつ増やす ) //クラスを見直す
        if( p が 1 と等しい )
            for(  ) //1 回目の重心の計算
                coordinate_x[t] ← gravity_x(t)
                coordinate_y[t] ← gravity_y(t)
            endfor
        else
            
            for(  ) //2 回目以降の重心の計算
                if( gravity_x(t) と coordinate_x[t] が異なる //重心を修正する
                    又は gravity_y(t) と coordinate_y[t] が異なる )
                    coordinate_x[t] ← gravity_x(t)
                    coordinate_y[t] ← gravity_y(t)
                flag ← 1
            endfor
        endfor
    endfor
```

設問 2 図 2 中の イ ～ オ に入れる適切な字句を答えよ。

プログラムのコードを入力する、というタイプ

ここがマークシートではなく、実際のコードを、手書きで記入します(ちなみに、回答は、イ:tを1からKまで1ずつ増やす、ウ:flag←0、エ:flagが0と等しい、オ:cluster[s]←min_index(grav_length)、になります)。

まあ、これもプログラミングには違いがないのですが、プログラムの一部を補うだけに過ぎません。文部科学省のいうところの「プログラミング的思考」とは違うかなあ、という感じがしました。

まだ大学入試で、「情報」の試験はまだ始まっていないのですから、当然過去問はありませんが、サンプル問題くらいはどこかにあるんじゃないかと思って、探しまわっていたら、[こちら](#)で見つけました。そのターゲットの問題は、[こちら](#)です。

ダウンロードして、印刷して ―― そして、可能なら、自力で問題を問いてみてください。

―― びっくりしますから。

第2問 表1の文章を読み、後の問い(問1～3)に答えよ。

Mさんは、18歳になって選挙権が得られたのを機に、比例代表選挙の当選者を決定する仕組みに興味を持った。そこで各政党に配分する議席数(当選者数)を決める方法を、友人のKさんとプログラムを用いて検討することにした。

問1 次の文章の空欄ア～ウに入れる最も適当なものを、後の解答群のうちから一つずつ選べ。同じものを繰り返し選んでもよい。

Mさん：表1に、最近行われた選挙結果のうち、ある地域のブロックについて、各政党の得票数を書いてみたよ。

表1 各政党の得票数

	A党	B党	C党	D党
得票数	1200	660	1440	180

Kさん：今回の議席数は6人だったね。得票の総数を議席数で割ると580人なので、これを基準得票数と呼ぶのがいいかな。平均して1議席が何票分の重みがあるかを表す数ということで、そうすると、各政党の得票数が何議席分に相当するかは、各政党の得票数をこの基準得票数で割れば求められるね。

Mさん：その考え方に沿って政党ごとの当選者数を計算するプログラムを書いてみよう。まず、プログラムの中で扱うデータを図1と図2にまとめてみたよ。配列Tomeiには各政党の党名を、配列Tokuhyoには各政党の得票数を格納することにしよう。政党の数は4つなので、各配列の宣言は0から3だね。

i	0	1	2	3
Tomei	A党	B党	C党	D党

図1 各政党名が格納されている配列

i	0	1	2	3
Tokuhyo	1200	660	1440	180

図2 得票数が格納されている配列

Mさん：では、これらのデータを使って、各政党の当選者数を求める図3のプログラムを書いてみよう。実行したら図4の結果が表示されたよ。

```

(01) Tomei = {"A党", "B党", "C党", "D党"}
(02) Tokuhyo = {1200, 660, 1440, 180}
(03) sousuu = 0
(04) giseki = 0
(05) mを0からアまで1ずつ増やしながら繰り返し;
(06)   sousuu = sousuu + Tokuhyo[m]
(07) kizyunsuu = sousuu / giseki
(08) 表示する("基準得票数:" + kizyunsuu)
(09) 表示する("比例配分")
(10) mを0からアまで1ずつ増やしながら繰り返し;
(11)   表示する(Tomei[m], ":", kizyunsuu / ウ)
  
```

図3 得票に比例した各政党の当選者数を求めるプログラム

Kさん：得票数に比例して配分すると小数点のある人数になってしまうね。小数点以下の数はどう考えようか。例えば、A党は2.068966だから2人が当選するのかな。

Mさん：なるほど。切り捨てで計算すると、A党は2人、B党は1人、C党は2人、D党は0人になるね。あれ？ 当選者数の合計は5人で、6人に足りないよ。

Kさん：切り捨ての代わりに四捨五入したらどうだろう。

Mさん：そうだね。ただ、この場合はどの政党も小数点以下が0.5未満だから、切り捨てた場合と変わらないね。だからといって小数点以下を切り上げると、当選者数が合計で9人になるから3人も多くなってしまう。

Kさん：このままでは上手くいかないなあ。先生に聞いてみよう。

基準得票数：580
比例配分
A党：2.068966
B党：1.137931
C党：2.482759
D党：0.310345

図4 各政党の当選者数の表示

ア～ウの解答群
① 0 ② 1 ③ 2 ④ 3 ⑤ 4 ⑥ 5 ⑦ 6 ⑧ Tomei[m]
⑨ Tokuhyo[m] ⑩ sousuu ⑪ giseki ⑫ kizyunsuu

最初、私は、『こちとら、ITでメシを食っているプロのエンジニアだ。高校生の「情報」の問題なんぞ、サクッと問いてやる』——と思っていたんですよ、本当に。

設問	内容
第1問	問1 2011年の東日本大震災で活用されたインターネットのメリットとデータ転送方式の特徴
	問2 用途に応じた図解思考ツールの選択の知識
	問3 コンピュータによる画像処理の基礎技術の知識
	問4 IPアドレスのネットワークアドレスとホストアドレスの設定方法

第1問の問1あたりでは『ほう、過去の災害を実例にしたインターネットの役割に関する問題か。なかなか着眼点がいいな』と言いながらサクッと解き、問2も、『この程度の資料が分からなければ、顧客プレゼンも学会発表もできないしな』と言いつつ、各問2秒で解きました。

問3は、『これって、実際に、画像処理のコーディングをやったことがないと、解けないかもしれないが……、でもまだ暗記の知識で解ける問題だな』と、これも秒単位で解きました。

変だな、と思い始めたのは、問4あたりからでした。

「IPネットワークのサブネットの設定？ これ実際に、ネットワークの設定やってみないと分からないんじゃないかな？ そもそも、高校生が、サブネットのビット長とか設定するかな？というか、デフォルトで192.168.0.0/24をクラスCで使うか、DHCPに設定するだけだろう？」と、ちょっと不穏な気持ちになってきました*）。

*）ちなみに、我が家の娘たちは、DHCPはもちろん、IPアドレスも理解していません。Wifiは、WPA2のパスコードで動き出すと思っています。

衝撃を受けたのは、第2問の「比例代表制度選挙の選出アルゴリズム」の問題でした。

設問		内容
第2問	問1	比例代表選挙の当選者を決定する、 プログラムコードの選択
	問2	上記のプログラムを実際に頭の中で動かしてみたときの、 変数の値の変化の追跡
	問3	上記プログラムの 完成形 と、若干のコード変更による 改良形 のコードの選択

問1は、情報処理技術者試験と、ほぼ同じ趣旨の「コードの穴埋め」問題でしたが ―― これ、前述のK-mean法の問題ほど単純ではなく、問題を正確に読み込まないと正解を導けない内容でした。

実際、私、この問題の内容が読みとれなくて、実際に自分でC言語でプログラミングをして、その結果を問題文と比較した程です。

問2は、自分の頭の中に仮想コンピュータを作って、その仮想コンピュータを実際に動かして、変数が変わっていく様子を追跡するという、デバッグ&トレースを模した問題でした。

そして、問3は、プログラミングの真髄である、『ほんの数行の変更で、プログラミングの内容を劇的に変化させる』コード改修の実例でした ―― しかも、その改修の出題が『うまい』。

仮に、上記、問1、問2の答が分からなくても、ここに入るべきコードは、一意に決まるようにできています。

で ―― 正直に言いますが、私、問題の内容が理解できなくて、プログラムをデバッグ繰返して、問題に記載されている答に近づくように、自分のプログラム(C言語)のデバッグを続けて、正解に至りました。

同業者の方は、すぐに気がつかれたと思いますが、

―― 私の頭の仮想コンピュータの能力は、『受験生以下』

ということです。

私は、コンピュータに対して、ブラックボックスアプローチ(正解に至るまで、何度もやり直す)という手法で、その場をしのいでいる、かなり低レベルのプログラマーである、ということです。

ぼんやりと見えてきた「プログラミング的思考」の正体

そして衝撃を受けたのが、第3問です。まさか、ここまで生々しいリアルなデータ解析を試されるとは思いませんでした ―― 驚いたのは、「難易度」でなく、「想定場面」です。

```

1 #include <stdio.h>
2
3 int main()
4 {
5     int Tokuhyo[] = {1200, 600, 1400, 180};
6     int Tosen[] = {0, 0, 0, 0};
7     int Tosenkel = 0;
8     int giseki = 0;
9     int Hikaku[] = {0, 0, 0, 0};
10    int maxi;
11    int Koho[] = {5, 4, 2, 1};
12
13    for(int m = 0; m < 4; m++){
14        Hikaku[m] = Tokuhyo[m];
15    }
16
17    while(Tosenkel < giseki){
18        int max = 0;
19
20        for(int i = 0; i < 4; i++){
21            if ((max < Hikaku[i]) && (Koho[i] > Tosen[i])){
22                max = Hikaku[i];
23                maxi = i;
24            }
25        }
26
27        Tosen[maxi] = Tosen[maxi] + 1;
28        Tosenkel = Tosenkel + 1;
29        Hikaku[maxi] = Tokuhyo[maxi]/(Tosen[maxi]+1);
30
31        printf("Tokuhyo: %d %d %d %d", Hikaku[0], Hikaku[1], Hikaku[2], Hikaku[3]);
32        printf("Tosen: %d %d %d %d", Tosen[0], Tosen[1], Tosen[2], Tosen[3]);
33    }
34
35    for(int k = 0; k < 4; k++){
36        printf("%d\n", Tosen[k]);
37    }
38
39 }

```

設問		内容
第3問	問1	サッカーチームのマネージャーが纏めた、各種データから見た、「強いチーム」と「弱いチーム」の解析
	問2	「得点」「ショートパス数」「ロングパス数」「反則回数」の相関係数による分析
	問3	「ショートパス本数」に着目した回帰直線による分析
	問4	合計、最小/最大値、平均、分散、標準偏差等の統計値を使った、全体傾向分析

まず設題の設定がなんとも『うまい』。

『サッカーチームのマネージャーが纏めた、各種データから見た、「強いチーム」と「弱いチーム」の解析』、など、高校生の日常に出てくるような設定が分かりやすく、同時に、『根性論や精神論だけでチームを強くしようとする、前世代的な脳筋コーチの批判』に見える点も、私には好印象でした。

そして、誰がどうみたって、"Microsoft Excel"を使っているという前提もリアルですし、相関係数、重回帰、そして、分散や偏差に対する「本質的な理解(×単なる方程式への代入)」を、現実にあるようなデータで試している点、そして、何より、データ解析の当初の目標『強いチームになること』で、設問が一貫されていることに感動しました。

「世界のどこかにいるデータアナリスト」のことではなく、今、ここで(受験会場)で、試験を受けている「お前」が、このデータから、自分のチームを勝利に導く戦略を立ててみる、という挑発的なアプローチが"熱い"です。



これらをまとめると、こんな感じでしょうか。

令和7年情報Ⅰ 大学入学共通テスト

サンプル問題の構成

設問	内容
第1問	問1 2011年の東日本大震災で活用されたインターネットの 問2 インターネット、ツール、 問3 画像処理の基礎知識 問4
第2問	問1 そのまま「選挙」で使える、 問2 実用的なプログラム 問3
第3問	問1 この結果を練習に反映す 問2 れば、インターハイ出場が 問3 可能になるかもしれない 問4 データ分析

文部科学省の“本気”を見た

—— なるほど、「プログラミング的思考」とはこういうことか

と、私にもボンヤリと分かってきました。

特に、このサンプル問題の問2、問3からは、

- (1) なんだか良く分からない数式の計算やら証明する(数学的アプローチ)だけではなく、
- (2) 振り子やバネを振り回して、着地時間を算出する(科学的アプローチ)だけではなく、
- (3) ウイルス感染を確認するためのPCR検査手法を説明する(技術的アプローチ)だけではなく、
- (4) リトマス試験紙で水溶液の酸性やらアルカリ性を調べる(工学的アプローチ)だけではなく、

目の前にある喫緊(きっじん)の問題を —— 完全じゃなくてもいいから —— 今、この時点で、手に入っているモノ(データとコンピュータ(パソコン))だけで、可能な限りアプローチしてみろ、と、煽り立てている様子が見て取れます。

そして、文部科学省の掲げる「プログラミング的思考」とは —— これは、完全な私の解釈ですが、

- そもそも完全な情報なんて、100年後だってそろいやしない
- 不完全な情報から、今できる場当たりの解決方法を考えて、実行と修正を繰り返して続ける
- とりあえず、“データ”さえあれば、“プログラミング”で事態を動かすことができるはずだ

という世界観(または価値観)のことではないか、と思うのです。

これは、政府資料を読んでも、研究者が書いた論文を読んでも、さっぱり分からなかった私が、逆方向（入試問題）から、私なりに解釈してみたものです。

ただ、これは、30年間以上、思い付いたらその場でプログラムを書き、10万行から最大1億行レベルのデータと格闘してきた、「現役ITエンジニア」の解釈です。

私から見える「プログラミング教育」の風景が、文部科学省の担当者の方、教育研究者、現場の教師、特に、現役の受験生たちに同じように見えているか、と問われると、甚だしく疑わしい、とも思います。

いずれにしても、この私の解釈は、100も200もあるかもしれない解釈の、ほんの1つです。思いつき外している可能性は高いですが ―― とりあえず、江端智一人の見解として、今回公開させて頂きました。

蛇足ですが、このサンプル問題「スラスラ解けるなら、大学なんぞ行かずにウチの会社においで」、と言いたいくらいには、難しい問題だと思います。

「日本人の多くは理系科目が嫌いである」

では、ここから後半「STEM教育」に入ります。

私は前回、STEM教育について、以下の仮説を上げていましたが、どうやら、「理系研究員としてのおごった見解」というわけではないようです。

既出:(4) STEM教育の意義(その2)

江端仮説:「(A)放射能と放射性物質/(B)感染と発病/(C)マグニチュードと震度/の違いが理解できる国民を作る」

<https://www.kobore.net/diary/?date=20150604>

(1)『子どもへの放射能被害が不安です』という保護者の皆さんは、今からでもいいから、原子力エネルギーや放射能について、死ぬほど勉強しなければならないと思うのです。

(2)「文系出身者7割」の日本において、多くの人が、10の12乗を越える数字を取り扱う経験がないことも、よく知っています。

(3)しかし、私達は、訳も分からないまま、制御システムに『自分の子どもが殺されるかもしれない』と怯えるだけの日々からは、決別しなければならない ―― と思うのです。

[著者のブログ](#)

「夏休みの自由研究」という小さな話題から展開した論ではありましたが、ここで一つ、はっきりさせたいと思います。

―― 私たち日本人の多くは、STEM教育についてうんぬんを語る以前に、そもそも、理系科目が『嫌い』である

ということです。

「苦手」「不得意」「欠点」「短所」「泣き所」「ウイークポイント」「弱点」「鬼門」というレベルではなく ―― 紙の色が変わるだけの理科の実験、ただ上下振動するだけのバネ、100万年前の大陸の場所、どっちでも構わない核分裂と核融合 ―― こんなことを繰り返されて、理系科目に対する、「嫌悪」とか「憎悪」とか「反感」に近い感情が、子どものころから育てられ、そして、それが、ほぼ大人になって完成している ―― ように思えます。

もちろん、全ての子どもが、そのように思っている訳ではないとは思いますが、我が国の文系:理系の大学出身者数比率 7:3は、この傍証になっていると思います。

ただ、この「理系嫌い」の理由については、

- (1)『夏休みの自由研究』のように無責任に放置された課題によって、子ども(と保護者)が虐待されている状況
- (2)「理系に愛されない子どもは何をしても愛されない」という信念
- (3)理系的知見のない保護者は、同じく理系的知見のない子どもを育てるという誤解
- (4)そもそも、理系の勉強は難しいし、面倒くさい、という事実

など、いろいろなことが言えると思うのですが ―― いくら仮説を出したところで、結局のところ「理系嫌い」が改善される訳ではありませんし、夏休みの自由研究が、これからも、子どもと保護者を悩ませ続ける状況は改善されないでしょう。

STEM教育は、始まる前から破綻している？

前回のコラムで、私は、STEM教育を、「科学・技術・工学・数学」をバラバラに単独として教えるだけではなく、それらを「組み合わせて使う方法」と、その方法を、座学(講義)だけでなく、「現実世界のケースに適用して学ばせる」実践的な教育」と定義しました。

江端定義の「STEM教育」

項目	STEM教育
内容	科学(S)・技術(T)・工学(E)・数学(M)の教育分野を纏めた(単なる)名称
実施事項(超ザックリ)	S,T,E,Mをバラバラ、単独の教科として取り扱うだけでなく、“S” x “T” x “E” x “M”という『かけ算式』に、目の前の問題を放りこんで、「解いてみる」
目的	“STEM”という『かけ算』で、日本や世界が直面している絶望的に難しい社会問題が解ける(かもしれない)
課題	我が国の各分野の業績は立派だが、「組合わせ」で「拡張」して「応用」できる人材がいない

我が国の希望としての、 プログラミング教育 & STEM教育

面倒なので、短縮形として「科学・技術・工学・数学」という必要な食材を、大鍋に入れて、ごった煮として取り扱う方法(メソッド)を学ぶ教育 ―― さらに縮めて

「ごった煮型理系教育」

と称呼しても差し支えない、と考えています*)。

*) 英訳すると、“Hodgepodge (ホッジポッジ)-style science education”となるようです。

しかし、その「理系ごった煮教育(STEM教育)」は、その前提として、理系科目が『嫌い』ではないことが必須です。しかるに、ここまで述べてきたように、我が国の現状は、STEM教育云々(うんぬん)を語る以前の状況にあるのです。



出典: [下野新聞社](#)

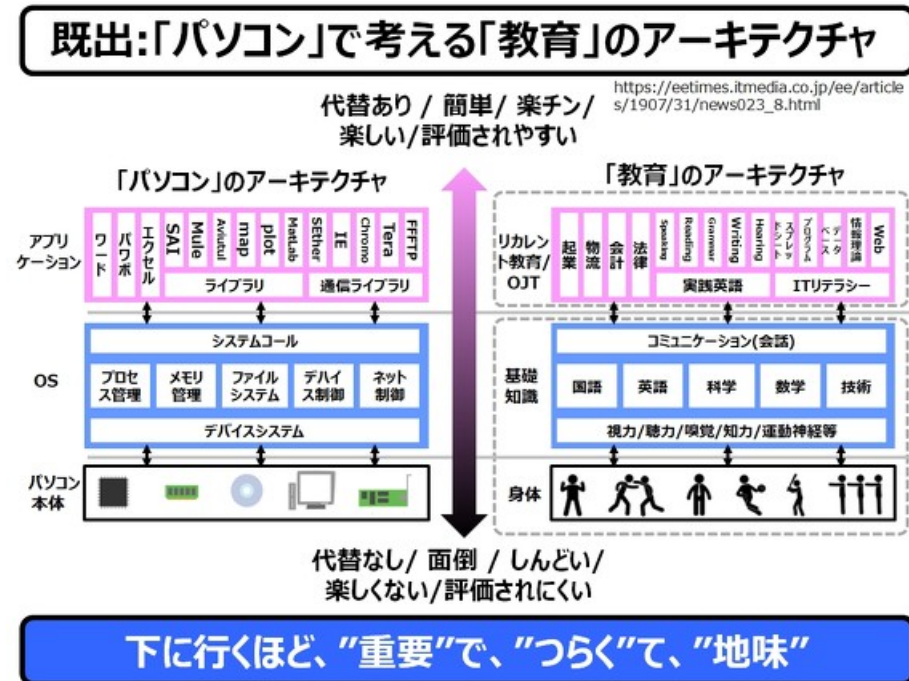
とすれば、そもそも、STEM教育は、それを開始する前から既に破綻している、というべきでしょう。そして、不思議なことに、この我が国の「我が国のマジョリティーの理系嫌悪」が、ちまたのSTEM教育論では、見事なほど完璧に無視されているのです ―― 「見ないフリ」をしているのかもしれませんが。

ただ、この結論のままですと、私の連載が終了してしまっ、それはそれで、私が大変困ることになります。ですから、私は、大きな伏線を張っておいたのです ―― それが、前半の「プログラミング的思考、大絶賛」という壮大な前フリです。

では、ここからは、「STEM教育は、それを開始する前から既に破綻している」をひっくり返す、理論展開を試みたいと思います。

□

以下の図は、「リカレント教育」のコラムの「[なぜ、初等教育の段階から「収入に直結する実学」を教えないのか](https://eetimes.itmedia.co.jp/ee/article/s/1907/31/news023_8.html)」で使ったものです。



学校教育の目的は

- (1)『「パソコン本体」に対応する、『心身共に健康で頑強な肉体』』であり
- (2)『「超高速大量の入出力信号をさばくOS」に対応する、『様々な知識や技術から作られた知能基盤』』であり
- (3)『「インストールしてすぐに使うことのできるアプリケーション」に対応する、『収入に直結する実学』』

となります。

この3つの層で完成している「パソコン」も「教育」も、低コストかつ短期間でさまざまなアプリ(教科)やらサービス(業務)やらを、利用可能とできる点において、優れていると言えます。

ただ、一つ問題があります。最初からこのアーキテクチャの「上位層」の部分だけを作ることはできない、ということです。「下位層」から作っていかねば「上位層」を作れないのです。それは、OSのないパソコンにアプリをインストールすることができないことと同じことです。

そして、「パソコン」も「教育」も、この「下位層」を作る作業が、実に面倒くさいのです。特に人間の場合は、ハードウェア(身体)とOS(基礎学力)を自力で作らなければなりません。学校教育のコンテンツは、パソコンショップや通販で買ってくることはできないのです。

「教育」の「下位層」を作ることは、しんどいし、楽しくないし、そして、大して評価もされません(あの、計算ドリルとか漢字の書き取りを思い出してください)。そして、その最もつらく、楽しくもない作業を、人生で最も忍耐力のない子どもに強いらなければならない点に、この教育アーキテクチャのジレンマがあります。

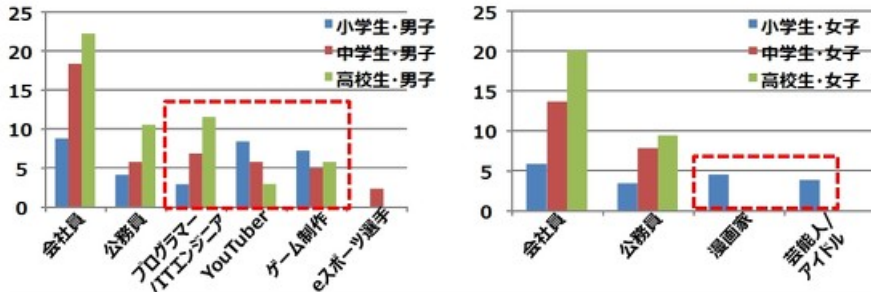
では、この、最もつらい「下位層」を作る作業が、STEM教育(またはプログラミング教育)でラクチンになる、としたらどうでしょうか？これは十分に検討してみる価値があるのではないのでしょうか。

IT系が食い込んできた？「大人になったらなりたいもの」

さて、ここで一つ、子どもたちが「大人になったらなりたいもの」[調査結果](#)をご覧くださいと思います。この情報を、私の視点でグラフでまとめてみました。

「大人になったらなりたいもの」調査

第一生命株式会社殿のデータを使わせて頂きました



女子に、“ITのかけら”もない

驚いたことに、近年、男子においては「IT系」の人気が高くなっています。“YouTuber”は、ITへの深い知識なしには成立しない職業なので、IT系でくっついています（もっとも小学生がそこまで考えているかは不明ですが）。

さらに驚いたことには、女子においては「IT系」について影も形も見えない、ということです。「漫画家」と「アイドル」は、“IT”でくくれるかと、入れてみました^{*)}が、正直無理があると思います。

*) これからの漫画家は「ペンタブ」を普通に使うだろうし、いまどきのアイドルも、生き残るために、自力でYouTubeのコンテンツ編集や情報発信をするだろう、と考えました。

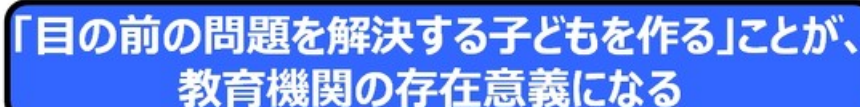
もっとも、これからの時代、スマホやPC抜きの仕事は考えられないので、全ての子どもたちに、ITに関する知識と運用が必要となることは間違いないのですが、『ITを作る側で食っていく』というスタンスは、（現時点では）女子には見られません。

そもそも「プログラミング」というものに人生で一度も関わったことのない子どもに、「プログラミングに興味を持て」というのは無茶というものです（男子は、何も考えていないのか、深く考えていないだけかもしれません）。

「課題解決型の教育方式」への転換

まあ、課題山積ではあるのですが、私の提案するアプローチは、「課題解決型の教育方式」への転換です（——と、自分で言っておいて何ですが、これ、そんなに簡単じゃないだろうことは、私もよく分かっています）。

「目の前のトラブルをサクッと片づけられる人間」の製造



そして、子どもたちの頭の中に"STEMプラットフォーム"を作って、社会に送り出せば良いのです。大丈夫、社会課題は、

上図に示したように、もう「うんざりする程あります」から、ちゃんとしたプラットフォームが作れば、将来も喰いっぱぐれることはありません。

しかし、『そもそも、社会課題って、"STEM"で解決できるものなの?』という疑問が生じるかもしれません。なにしろ、今回一貫して述べてきたように、我が国の国民の多くは「理系嫌い」です。

ですが、現在私たちが直面している社会課題は、ほぼ全てがSTEMの対象になっています。例えば、私の連載の内容を調べてみれば、このことは明らかです。

「STEM」で問題解決できるのか？

江端の「数字で・・・」の連載で取り扱ってきたテーマ

テーマ	Science 科学	Tech. 技術	Eng. 工学	Math 数学
(1)電力不足	エネルギー	発電	配電	統計
(2)環境問題	大気/化学	省エネ	センシング	質量計算
(3)ダイエット	エネルギー保存法則	あやしげな技術、山ほど	あやしげな機械、山ほど	微分方程式
(4)飛び込み自殺	力学、弾性力、重力	(飛び込みの)タイミング制御	運行監理システム	運動方程式
(5)生産性向上	行動経済学	見える化によるボトルネック検知	サプライチェーン運用システム	微分・積分・偏差
(6)介護	エビデンスベース介護	PDCAアプローチ	介護ロボット、支援	社会保障費予測
(7)コロナ	ウイルス学	mRNAワクチン	感染防止グッズ、装置	感染予測

「問題を解決している」とは言えないが、
「問題を“見える”レベル」にしている、と思う

私は、国会中継での代表質問を見ていると、『なぜ、もっと数字やデータを使って、ロジカルに質問や応答ができないんだろう?』とイライラしてきます。そもそも、"STEMプラットフォーム"を使わないで、社会問題を把握しようとする事態が、稚拙で非効率だと思っています。

加えて、私の連載コラムで取り扱ってきた社会課題（「[世界を「数字」で回してみよう](#)」）に限っていえば、STEMの大部分は、「プログラミング」を使って把握することができるのです。—— なぜなら、実際に私がプログラムで計算しまくってきたからです。

「デジタルツイン」

このプログラムの内容は、「シミュレーション」といっても良いものなのですが、今回は「デジタルツイン」という言葉を使って説明したいと思います。

デジタルツインとは、現実の世界から収集したさまざまなデータを、「双子」のように、コンピュータ上で再現することを言います。単なる数値データによるシミュレーションとは違って、コンピュータの仮想空間上でのオブジェクトを、現実世界にあるモノに連動させる、あるいは、意図的に連携させる点が特徴です。

「STEM」と「コンピュータ」は相性が良い

「デジタルツイン(双子)」という考え方

現実世界での現象



仮想世界(コンピュータ)の中に作った現象



しかし、コンピュータの中に、双子(の片方)を作るのは、これで結構大変だったりする

しかし、「デジタルツイン」は、それほど簡単なものではありません。

現実世界は、物理法則が自動的に適用されていますので、対象を観測し続ければ足够了(それも結構、面倒ですが)。

一方、仮想世界は、観測対象の構成要素や周辺環境を、下手すれば分子や原子の微細なレベルまで、コンピュータの中で計算し続けなければならないからです。

例えば、コロナ禍になって、すっかり有名になった「飛沫の3Dシミュレーション」があります。

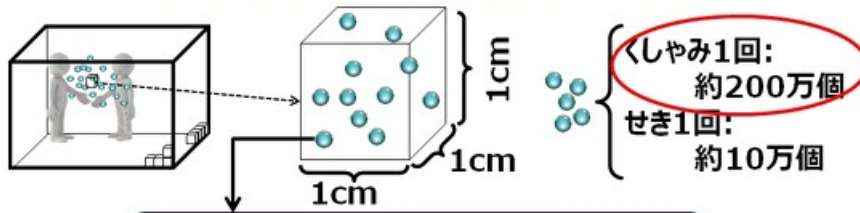
「飛沫の3Dシミュレーション」は、スーパーコンピュータの「富嶽」による成果として公表されることが多いですが――『私のパソコンでもできるんじゃないの』とひそかに思っていました(つまり、スパコンの宣伝戦略の一つではないかと、邪推していたということです)。

私のパソコンでも、合衆国大統領選挙の1万回シミュレーションを14時間で導く程度のスペックはあるからです(関連記事:「[沈黙する人工知能 ～なぜAIは米大統領選の予測に使われなかったのか](#)」)。

それで、今回、とても簡単な想定化での、「飛沫の3Dシミュレーション」の計算量を調べてみたのですが、愕然としました。

どんなに大変かというと・・・

もっとも単純化しても、ざっとこんな感じ



$$\nabla \cdot \mathbf{V} = 0 \quad (1) \rightarrow \text{連続の式}$$

$$\frac{\partial \mathbf{V}}{\partial t} + (\mathbf{V} \cdot \nabla) \mathbf{V} = -\nabla p + \frac{1}{Re} \Delta \mathbf{V} + \mathbf{F} \quad (2)$$

→非圧縮性 Navier-Stokes 方程式

$$\frac{\partial T}{\partial t} + (\mathbf{V} \cdot \nabla) T = \frac{1}{RePr} \Delta T \quad (3)$$

→熱伝導の方程式

$\mathbf{V}$: 速度ベクトル T : 温度 t : 時刻 Re : レイノルズ数
 p : 圧力 $\mathbf{F}$: 浮力(Z方向のみ) Pr : プラントル数

同じ1cm³の中では同じ条件であると仮定して、1/1000秒ごとに、200万個の飛沫粒子の動きを並列計算することになる

「くしゃみ一つ」で、最低でも3つの差分方程式を、20億回/秒という凄い計算量が必要

加えて、空調による空気の流れ、重力、水蒸気圧、マスクにぶつかった/透過した飛沫粒子の軌道修正計算が入ってきますし、ここに、飛沫防止のパーティションが立っていたとしたら、そのための運動方程式を加える必要があります。さらに、人間の体温による粒子速度や空気圧力の変化も必要でしょう。

そもそも、ここでは3つの基本式しか使っていないですが、他の文献では20以上の式を使っているモデルもありました。

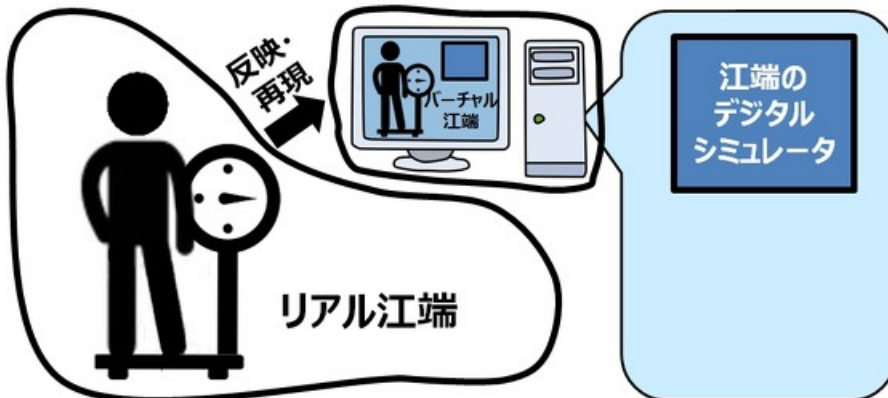
ごめんなさい。私が思い上がっていました。「飛沫の3Dシミュレーション」は、我が家のパソコンごときで太刀打ちできるような計算ではありませんでした。

学校教育で教えるデジタルツインは「ダイエット」だっていい

しかし、学校教育におけるSTEMで使うデジタルツインは、もっと単純な題材で良いのです。私がお勧めするパッケージの一例は、「ダイエット・デジタルツイン」です(関連記事:「[万年ダイエットにささげる、“停滞期の正体”](#)」)。

江端専用 ダイエット・デジタルツイン

“リアル”と“デジタル”の双子(ツインズ)

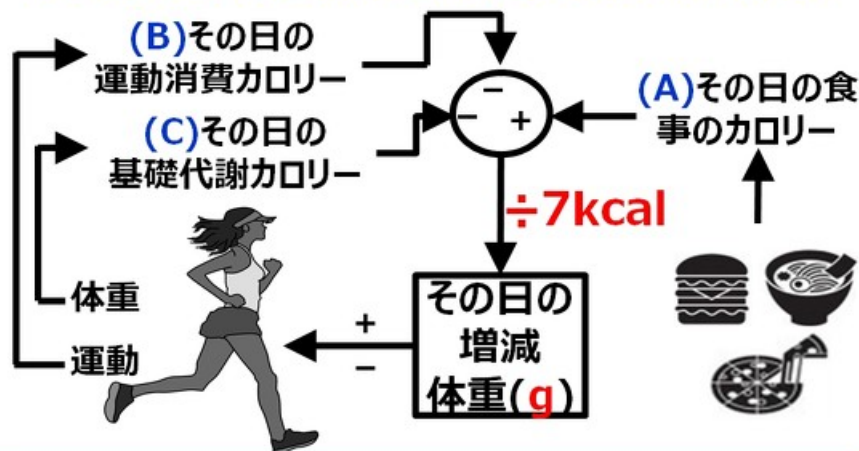


シミュレータの試作とパラメータの設定

で、このデジタルツインの概要はこんな感じです。

「超シンプル体重シミュレータ」

「足して」「引いて」「7で割る」だけの単純な「双子」



(A)=(B)+(C)の時に、体重の変化は止まる

これだけのもので、プログラムのステップ数は、20行もありません。非常に簡単です。[こちら](#)に、「【付録】超シンプル体重変動シミュレータ(C言語版)」がありますので、ご参照ください。

実測値からパラメータを抽出

この「双子」が、なかなか侮れない



江端は、毎日、牛丼大盛り分のカロリー
(780kcal)を削っていた・・・らしい

なぜ、これが「デジタルツイン」であるかという、私は、現実の自分の体(の体重)と、コンピュータの中の仮想空間に作った自分(の体重)を比較しているからです。

結果として、私のダイエット仮説「痩せたければ、食べない」を検証できましたし、「きちんと制御さえすれば、自分の体重を狙いの数値にドンピシャに持っていける」という確証を得ることもできました。

つまり、一切の特殊な装置や手続もなく、コンピュータ(パソコン)とエネルギー保存法則をベースとしたプログラミングだけで、STEM型教育を行い得るという、という一つの証明になっていると思います。

テーマ	Science 科学	Tech. 技術	Eng. 工学	Math 数学
(3)ダイエット	エネルギー保存法則	あやしげな技術・山ほど	あやしげな機械・山ほど	微分方程式

個人的所感としては、「ティーンエイジャの女子であれば、真剣に、『我が事』として、微分方程式のプログラミングと、エネルギー保存法則の勉強にとりかかりそう」な気がしています。

つまり「理系科目が嫌い」であっても、それを別の問題(課題)にすりかえて、そのアプローチとして、特別な高価な装置を使わずに、仮説を次々と試すことができるプログラミングは、その教育にかかるコスト(時間、人材、金)が、飛び抜けて安いのです。

文科省の“壮大な計画”(?)

このように、「STEM教育」×「プログラミング教育」のアプローチ、そして、「将来の社会問題解決への人材育成」という、文部科学省「理系日本人補完計画」の端緒が垣間見えるのです。

—— おそろべし、文部科学省。このような壮大な計画(2つの教育メソッドを組み合わせる)を、国民に隠蔽しながら進めていたとは……。

『あの分かりにくい、政府資料の数々は、このためだったのか』と、ようやく私は腹に落ちました。

そして、STEM教育とか、プログラミング教育に関わっている関係者の何人かも、この実体を知りつつ、“STEM”だの“STEAM”だのという、どーでもいい「言葉遊び」の論文を量産していた理由も、分かってきました。

うん、分かるぞ ―― 分かる。

このような、「理系日本人補完計画」が露呈すれば、間違いなく、"USB"がなんのことも知らないような議員を大臣に推す（[著者のブログ](#)）ような、絶望的なITリテラシー集団である我が国の与党が、文部科学省のこの計画をつぶしにかかるのは目に見えています。

急速にすすむIT化を、なんとか妨害し続けて、町内会の顔で居続けようとするジジイやババアたちの、チンケな自尊心^{*)}も、このような、「問題解決型人材の育成」の阻止に向くでしょう。人口の3割を占める65歳以上の人々も、そちら側につく可能性もあります。

*) 関連記事: [「デジタル時代の敬老精神 ～シニア活用の心構えとは」](#)

なぜ、私はそう思うのか ―― 『10年後の私なら「そう思う」と確信できるから』です。

社会の問題解決の速度に追いつくことなく、年下の若いやつらの言葉に従わされるだけの日々の情けなさは、想像するだけでも恐ろしいです。それは、重度の認知症になってしまって、全ての世界から隔離されるような、恐怖だと思います^{*)}。

*) 関連記事: [「見張れ! ラズパイ ～実家の親を熱中症から救え」](#)

ですが、そのような状況を看過しておいた結果が、『日本のデジタル競争力27位^{*)}』という体たらくです ―― 「技術大国日本? 何、それ、おいしいの?」という感じです。

*) 参考: [IMD WORLD COMPETITIVENESS ONLINE](#)

PCが使えない政治家、遠隔授業のできない教師、リモート会議システム設定すらできない会社の社長、そして、メールを使えない町内会会長 ―― もう、あなたたちに「優しい顔」をする時期は、遠(とう)に過ぎました。

これからは、『ティーンエイジャーにすがりついてでも、死にもの狂いで、世界に追いついてこい』 ―― そして、『何人(なんびと)も、文部科学省の「STEM教育」×「プログラミング教育」の計画を邪魔をするんじゃない』と申し上げておきます。

もっとも、それは、「今回の、私(江端)の壮大な妄想、『理系日本人補完計画』が正解だったら」という条件付きにはなりますが。



□

では、今回の内容をまとめます。

【1】冒頭にて、「夏休みの自由研究」に対する、子どもと保護者の意識から、日本人の根深い「理系嫌い」についての所感を語ってみました。

【2】文部科学省の「プログラミング教育」における「プログラミング的思考」について、(私の頭で)理解可能な資料を見つけられませんでした。そこで、令和7年度(4年後)大学入学共通テストからの出題教科・科目「情報」のサンプル問題を、実際に問いてみて、その意義を推測してみました。

【3】その結果、「目の前にある喫緊(きっきん)の問題を ―― 完全じゃなくてもいいから ―― 今、この時点で、手に入っているモノ(データとコンピュータ(パソコン))だけで、可能な限りアプローチしてみろ」という世界観が見えてきました。

【4】私(江端)は、STEM教育を科学・技術・工学・数学」という必要な食材を、大鍋に入れて、ごった煮として取り扱う方法(メソッド)を学ぶ教育 ―― さらに縮めて「ごった煮型理系教育」と定義しました。その上で、「日本人の理系嫌い」のために、「STEM教育は、それを開始する前から既に頓挫している」のではないかと考えました。

【5】この考え方をひっくり返すために、STEMを使った「課題解決型の教育方式」への転換を提案しました。

しかし、そもそも、社会課題って、そもそも"STEM"で解決できるものなの？ という疑問に対しては、私が執筆してきたコラムの内容分析から、その可能性があることを示しました。

【6】 今回の「プログラミング教育」と「STEM教育」の相乗効果を得る方法として、シミュレーション(デジタルツイン)によるアプローチを提言しました。その実例として、「ダイエット・デジタルツイン」の具体例を示し、恐らく、「ティーンの女子にウケに違いない」という江端の私感を交えて、論を展開しました。

以上です。

料理だって「STEM」である

あらためて考えてみますと、STEMは、別段「何か特別なこと」ではありません ―― というか、物事は大抵のことがSTEMで成立しています。

例えば、次女は「菓子作り」を趣味としているのですが、ことしのバレンタインデーでは、『マカロン』を巡って、大層お怒りのようでした。(正直、私も良く知らないのですが)、マカロンというお菓子は、その難しさにおいて、他の菓子とは比較にならないものだそうです。



マカロンを最高の状態で焼き上げるためには、材料の量、混合比、燃焼温度、放熱時間、外気温や湿度、その他もろもろの要因が完成を左右し、さらに、それらの要因が、独立変数でない ―― 変数どうしが影響し合うということで、つまるところ、最適解が「線形問題」でとて解くことができない、つまり、

―― 手作りマカロンは、数学で言うところの「NP困難問題」である

ということです。

しかるに、『バレンタインに「手作りマカロン」を平気で要求してくる、あの男は馬鹿か?』と、娘は、自分の彼氏を罵倒しまくっていました。

このように、「手作りマカロン」は、高度な数学(Math)であり、忠実なレシピを必要とする技術(Technology)であり、均一の同じ味のマカロンを作るという意味では製造工学(Engineering)で、物質に水溶物を加えて粘性物に変化させて、火力等で燃焼するプロセスは、立派な科学(Science)です ―― つまり、「マカロンはSTEM」なのです。

そもそも、料理のレシピというものは、「プログラム」です。そのプログラム通りに作れば、基本的に、いつでも、どこでも、誰でも、おおむね同じ料理を製造できるというものです ―― プログラムが、どのスマホでも、どのPCでも同じように動くように。

私は、結婚前の花婿修行として、市民会館で開催されていた『男の料理教室』なるものに通っていました。そこで提供されているレシピの一つが、これでした(20年以上も前のフォルダファイルから取り出してきました)。

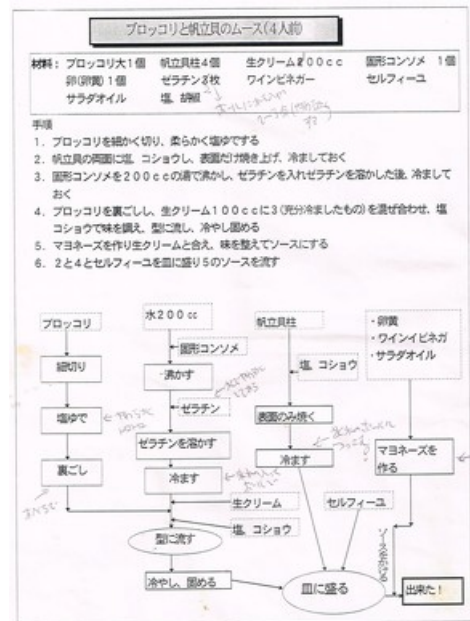
今となっては、あまり珍しくないフローチャートで記載された料理のレシピですが、当時、その「分かりやすさ」に感動したのを覚えています。

料理とは、複数の処理を同時並行に行う工学的処理です。なのに、今なお料理のレシピが、平文だけで記載されているという、その野蛮さに、私は腹を立てています。

しかも、料理というのは、時系列的に化学的変化を続けていきますし、また、調味料は厳密に計測しないと、料理の味を劇的に変えてしまうことがあります(私は、「塩」でよく失敗しました)。

しかも、料理のレシピでは、初出の用語を未定義で平気で使います。

「適当な大きさ」、「塩、少々」、「ひとくち大」、「さっと湯通し」、「ひとつまみ」、「斜め切り」、「味を整える」、―― こういう表現の多くが、多くの人間(理系出身者に多いと思う)から、料理を作るという気力を失わせているように思われます。これは、我が国の「理系嫌い」を生み出すプロセスとも、酷似していると思います。



□

あと蛇足ですが、「調理」というのは、誰が何と言おうとも「工学(エンジニアリング)」であり、完成した料理は「完成したシステム」です。——が、どんなに言葉と時間を尽くして説明しても、嫁さんと娘は、これを認めません。——『飯がまずくなるから』だそうです。

本質を正しく理解し、それを表現する日本人を量産するためにも、「プログラミング教育」と「STEM教育」は必要だと思います。

これは「テキシコー」です

ここまでを読み終えた後輩がコメントを突っ込んできたので、その話も追記しておきます。

後輩:「江端さんは、相変わらず『プログラミング的思考』が分からない」と繰り返していますが、これを見れば、一目瞭然です。「テキシコー」です」

テキシコーとは、「コンピュータを使わずにプログラミング的思考(てきしこう→テキシコー)を育む」という[NHKの教育番組](#)です。

「とりあえず、ガタガタ言わずに、これを全部見た後で、語ってください」と、後輩に言われたので、一応全部視聴しました。——なるほど、「プログラミングを行わないプログラム」というものが理解できました。皆さんにも、第一回目だけでも良いので、視聴してみてください。

ただなあ……この番組で『プログラミング的思考が理解できる』としても、『プログラミング的思考が使えるようになる』とは、別の問題だよな、とも思いました。もっとも、それをNHKの教育番組に期待するのは無茶であることは、分かっています。

私としては、この10回の番組を、100回見ても、1000回見ても、『テキシコー使いになれる』とは思えない。——多分、私なら「楽しいな」と続けて見続けて終わるだけだろう、と思います*)。

*) ちょっと悪意を込めていうと、『プログラミングを自由自在に操れる人が、プログラミング抜きでプログラムを語れ、と命じられたら、多分、こういう番組を作るだろう』とも思いました。

多分、小学校、中学校、高校の中で、子どもたちが、この『テキシコー』を発動する場面は、

(1) 文化祭や運動会を3日前に控えて、いろいろなところから問題が上がりてきて、スケジュールが絶望的な状態になっている(そして、顧問の教諭がクソの役にも立たない)生徒会や実行委員会の役員たち

(2) 修学旅行の班決めを行うに際して、どの班にも受け入れてもらえないために、リア充たちへの懐柔や根回しをしなければならなかった(私のような)陰キャラ学生
(3) 既に、翌朝の試験開始まで10時間という時間を切り、3教科の勉強をしなければならない、一夜漬けを覚悟した(「赤点回避」のみにロックオンした)成績不良生徒

これらは、全て『かつての私』のことですが ―― 追い詰められた最後の段階で発動するものであり、大抵の場合は、発動に失敗します。そもそも、「テキシコー」が発動する場面なんて、子どもの日常にあるかな? と思うんです。

だから、私は、子どもたちが、そのような「テキシコー」が発動できる場面は、せいぜい、定期テストか、入学試験の問題の中の、仮想の世界にしか作れない、と思うのです ―― もっとも、大人になれば、できるだけ手を抜いて、多くの仕事を短時間でさばく必要がありますので、「テキシコー」は重要になってくると思います。

そのように考えればNHKの教育番組で、「テキシコー」を見ておくだけでも十分かもしれませんが ―― 私は、プログラミング的思考を理解するには、プログラムやツールを直接使った方が、てっとり早いように思えるのです。

「[陰湿な人工知能 ～「ハズレ」の中から「マシな奴」を選ぶ](#)」の記事にも書いた、

――「100冊の人工知能の本を読んでも時間の無駄です。最もてっとり早いのは、100行のプログラムを自分で書くことです」

は、プログラミング的思考についても同じである、と、私は思っています。

「科学」とは教科ではなく、「手段」である

後輩:「『夏休みの自由研究に苦しんでいる→日本人の多くが理系嫌い』のロジックは、乱暴に過ぎませんか?」

江端:「いや、子どもと保護者が、夏休みの自由研究に苦しんでいるのは、各種の文献やニュースからも、客観的な事実だ」

後輩:「そうじゃなくてですね、江端さんの現状把握は、そもそも前提が間違っているんですよ」

江端:「?」

後輩:「夏休みの自由研究の内容は、別に理科の実験でなくてもいいですよ。例えば、『豊臣秀吉と徳川家康を、豊洲エリアで合戦させてみたら、どちらが勝利するかをシミュレーションする』という研究だっていいわけです」

江端:「なんだ、その、魅力的なシミュレーション。私は、絶対にその研究発表を見にいくぞ」

後輩:「あるいは、『東京の司法、行政、立法機関を、自分の住んでいる県庁所在地(例えば、秋田県)に移転することが突然決まったとした時の、それらの運用方法についての検討』とか」

江端:「すごいな。聞いただけでワクワクする研究だな」

後輩:「なんで、こういう、歴史上の戦力分析シミュレーションとか、社会科学に関する研究とかが、子どもか出てこないと思いますか?」

江端:「そりゃ、研究内容が難しいからだろう?」

後輩:「違いますよ、江端さん。『理科を題材にした自由研究の方が圧倒的にラクだから』です」

江端:「……はい?」

後輩:「理科……という科学というのは、「仮説→観察→評価→検証」をループを続ける進歩のプロセスです。そももって、何がラクかという「観察」ができる点にあります。極端なことを言えば、自然現象を見て、それを記録さえすれば足りるのです。しかも、科学における観察結果は、誰が何度やっても、同じ結果になり(再現性)があり、大抵の場合、常識的な結果に納まります ―― 後は、適当に感想を書けば終わりです。こんなラクな宿題はありません」

江端:「だから、子どもも保護者も、夏休みの自由研究に、『理科の実験』を選ぶ――と?」

後輩:「理科に関する自由研究のサンプルは、江端さんも書いていましたが、日本中に腐るほどあります。それをパクってくれば足る話です。それすらできないのは、科学の「仮説→観察→評価→検証」という考え方を、子ども、保護者、下手すると教師ですら知らないからです」

江端:「なるほどなあ」

後輩:「自由研究の結果は、どんな手法を用いて、どんな結果になっても、『全てが正解』なんです — 多分、日本人の多くは「夏休みの自由研究の正解」があると思っています。漢字の書き取りや、計算ドリルみたいに、正確で唯一の答えがあると信じているのです。そして、親子して「夏休みの自由研究」の発表(文化祭等での公開を含む)で、恥をかきたくなくて、苦しんでいます」

江端:「ふむ……。定期テストであれば、0~100点の間のどこかの数値で落ちつくけど、『嘲笑』とか『苦笑』という、下限のないマイナス査定を喰らうことは、相当な恐怖かもしれない — 特に「恥の文化」を重視するわれわれ日本人にとっては」

後輩:「中学生で『蝉の抜け殻の収集』を展示したら恥ずかしいですし、高校生で『徳川家康の生涯年表』なんぞを提出したら、黒歴史確定ですよ。自由研究のサンプルをパクるにしても、その年齢に応じた内容を提出しなければなりません」

江端:「それはさておき、つまるところ — 『日本人は理系が嫌い』ではなく、『日本人は、科学というもの(仮説→観察→評価→検証のプロセス)を知らない、または、慣れていない』ということかな?」

後輩:「実際、私たちは正しい意味で"科学"を教えてもらっていませんからね。科学とは、教科ではなくて、手段です。STEM教育とは、"政治"や"経済"や"文学"や、"歴史"や"古典"にすら利用できる「『科学』という便利な道具の使い方」を学ぶことだと思います」

江端:「できるかな。今の子どもたちの勉強の内容は、そこそこ難しい上に、教科も多い。ここに「科学という道具概念」を投入するスキマが残っているかな?」

後輩:「江端さんが今回提案している、『ダイエット・デジタルツイン』みたいな、課題解決型パッケージ教育は、アリだと思いますよ。もっとも、これがうまくいくかどうかは微妙だとは思いますがね — これ、暗記だけさせる理系教育に比べると、比較にならないくらい格段に内容が難しいです」

江端:「そうだよな。子どもたちには、酷だよな」

後輩:「いや、酷なのは、むしろ教師です — このパッケージ教育を強行したら、教職員が退職して日本の教師が半分くらいになるんじゃないかと、心配です」



Profile

江端智一(えばた ともいち)

日本の大手総合電機メーカーの主任研究員。1991年に入社。「サンマとサバ」を2種類のセンサーだけで判別するという電子レンジの食品自動判別アルゴリズムの発明を皮切りに、エンジン制御からネットワーク監視、無線ネットワーク、屋内GPS、鉄道システムまで幅広い分野の研究開発に携わる。

意外な視点から繰り出される特許発明には定評が高く、特許権に関して強いこだわりを持つ。特に熾烈(しれつ)を極めた海外特許庁との戦いにおいて、審査官を交代させるまで戦い抜いて特許査定を奪取した話は、今なお伝説として「本人」が語り継いでいる。共同研究のために赴任した米

国での2年間の生活では、会話の1割の単語だけを拾って残りの9割を推測し、相手の言っている内容を理解しないで会話を強行するという希少な能力を獲得し、凱旋帰国。

私生活においては、辛辣(しんらつ)な切り口で語られるエッセイをWebサイト「[こぼれネット](#)」で発表し続け、カルト的なファンから圧倒的な支持を得ている。また週末には、LANを敷設するために自宅の庭に穴を掘り、侵入検知センサーを設置し、24時間体制のホームセキュリティシステムを構築することを趣味としている。このシステムは現在も拡張を続けており、その完成形態は「本人」も知らない。

本連載の内容は、個人の意見および見解であり、所属する組織を代表したものではありません。

Copyright © ITmedia, Inc. All Rights Reserved.

