

pirika.comの研究記録

Pirika News



溝口の文教堂に読み終わった本を入れるボックスがある。ここに入れた本は文教堂が売ってその売上を寄付する。読み終わった本で社会に貢献！

すごくいい企画だと思う。もっと広がれば良いのに。

ブログ

2023年のブログはものすごく減った。ブームが去ったというか。元々、SNSなどに合わせた文章の書き方の勉強ぐらいの意味しかなかった。刺激的なタイトルの付け方。Twitterに合わせ、最初の三行ぐらいがポイント。大勢が読んでくれないと、なんとなく不安でどんどん時間を使ってしまう。

読んでくれる人は少なくとも、じわじわと検索のトップに這い上がる。そんな自分らしいページの作成に回帰している。

HSPiP新バージョン

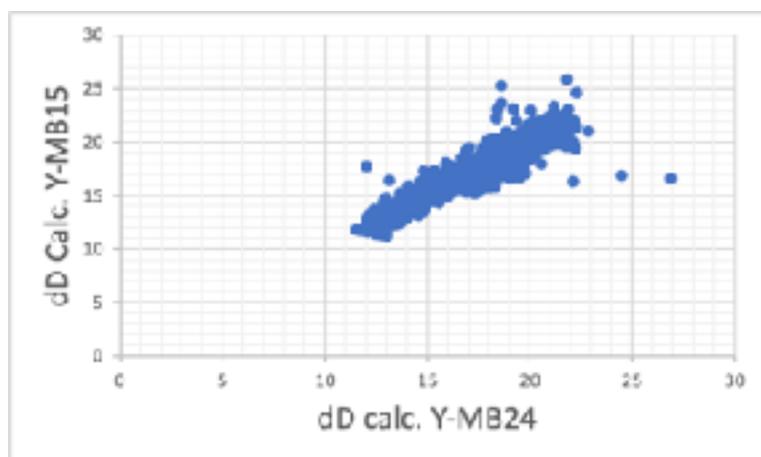
Ver.6ではY-MBが大きく変わるため、計算結果が変わってしまうことがある。

古いバージョンは残しておくことをお勧めする。

~~eY:Program Files¥Hansen-Solubility¥HSPiPのHSPiPをHSPiPv5.4とかに変更してからインストールすれば古いバージョンを残せる。ついでにドキュメントのフォルダーにある、HSPiP-Dataフォルダーも別名にしておけば、クリーン・インストールできる。~~

ver.6ではHSPiP-6にインストールされるので名前を変える必要はない。

Y-MBが新しくなった。HSPiPのは一体いつからバージョンアップしていないのだろうか？も



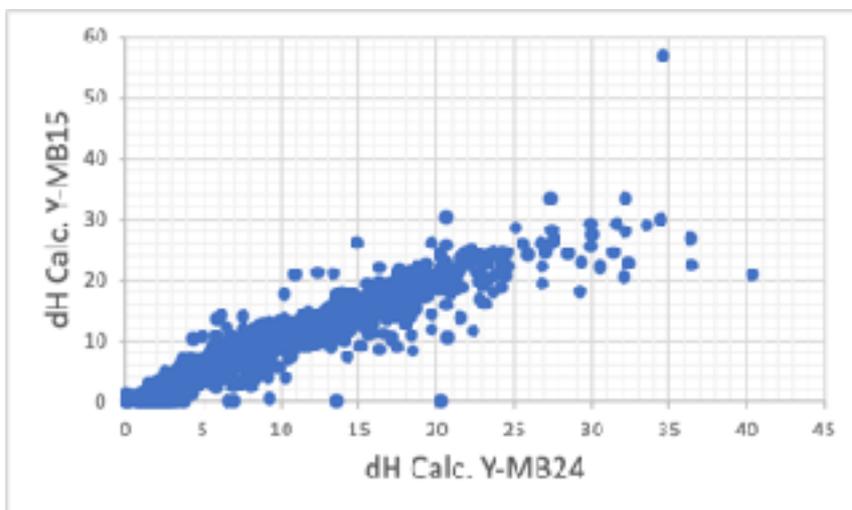
しかしたら10年？ 実際には、ほぼ2年にいっぺんは新しいバージョンのY-MBを作成している。HSPiPに搭載されていないだけだ。Y-MBの計算値を変更すると、HSPiPの色々なところに影響を与えてしまう。一番大きな影響は、オフィシャル値の無い化合物だ。推算値が変わってしまう。

HSPのオフィシャルな値を持つ化合物は1300化合物弱だ。それに3000化合物の準オフィシャルなDBを作っている。屈折率、誘電率、ダイポールモーメント、蒸発潜熱、密度などHSPを決定するのに必要な物性値のあるもの。それから決定されたHSPとY-MBなどの推算値を比較して矛盾の無いものを集積してきた。もう1つ大事なものは推算値がオーバーシュートしないように、限界値を規定するアンカーという化合物を定義している。

そのデータを元にY-MBを再構築した。結果は結構変わっている。

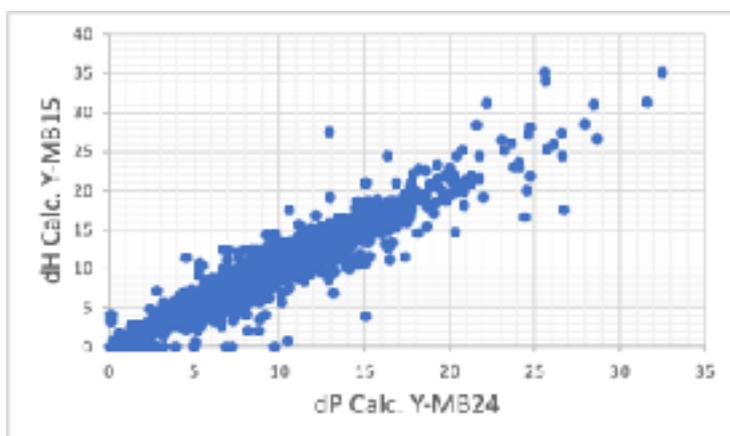
リリース・ノートにも記載したが、論文や特許に使ったバージョンは残しておくことをお勧めする。先生のところにも古いものは無い。論文や特許には使ったバージョンが当然記載されているだろうから、古いバージョンが残っていれば、特許で問題になった時にもすぐに再

計算できる。(こうなってくると、MacユーザーでParallelsを使ってWindows 7まで戻れる私などは、有利だな。いまだにssdフォーマットのものまで読める。)



HSP以外の物性値の推算値も変更になる。統計的にはこの新しいY-MBの方が優れている。しかしユーザーの関心ある化合物の推算値が常に良くなるかという、必ずしもそうとも言えない。その時には前のバージョンを使い続けた方が良い。所詮、Y-MBは推算式だ。バージョンごとに常に変化している。

新しいものと現行のものではHSPの推算値もこのくらい変わっている。



こうした新しいHSPの値や他の物性値を使って物性推算を行なっているものがある。そうした2次利用の物性値はこれから順次改定されていく。

ver.6.1に向けての取り組みだ。

Y-MBをクラッキングして大学独自のハンセンパラメータと自称している、関西の大学はどうするのか見ものだ。またY-MB24をクラッキングするのだろうか？

ver.6に搭載されなかったElectron Donor/Acceptorや数多くのMI用識別子がある。これらは、P-MB (Pirika Molecular Break)として独自に開発・販売を進めていく。

試しにC#版とOfficeScript版を作ってみた。自分で言うのもなんだが、ものすごく良い。

裸の王様と新入社員一括採用

ゆとり教祖様を「裸の王様」と崇めたてまつる社会

新入社員の通年採用はなかなか進まない。そして飽きもせず「ガクチカ」とか意味のないくだらない質問をして、学生を困らせている。「答えの内容ではなく、答え方を見ているんだ」とか言うけど、最初のエントリーシートで篩わけに使っているのだから「答え方なんて見ていない」のは明らかだ。「学歴フィルターで落とすのではない」という言い訳に使っているだけだ。

ダイハツに内定もらっている学生がどうなるのか、心配だよ。

三菱自動車のリコール隠し、SUBARUやスズキの燃費不正などが色々表に出て、こうした不正を行うと会社がどうなるか、ダイハツの不正を行っていた従業員が知らないはずがない。ましてや、トヨタから天下りした社長が知らないという言い訳はできない。

三菱自動車は天下の三菱グループから追い出されて、日産-ルノーに拾われた。どこでも可哀想なのは普通に頑張ってきた社員だ。

電通の過労死の時にも出たが、「社会人になって初めて働き始めた会社」でみんなが普通にブラックに働いていると、それが普通だと思ってしまう。鳥類が卵から生まれて初めて見る動くものを親と勘違いするのと同じだ。いいか悪いかを自分で判断しないで、裸の王様に従う。

その状態にするには、新入社員一括採用は良いやり方だ。他所の会社を経験して優秀な奴が途中で入ってきたら、「あいつ裸でやんの」とか言いかねない。

主張の強い、外国籍の人材は取らない。

昔はもっと違った意味合いがあったのだと思う。東大生だろうが、日大生だろうが、新入社員のスタートラインは同じ。給料も同じ。結果的に東大生の方が早く出世する可能性も高いが、僕みたいに部下なしの主任ぐらいで終わる場合もある。多様性を残しつつ、競争原理を導入するにも新入社員一括採用は優れている。

働き方のブラック度に関しても、今の方が良いとは僕には言えない。

大学の時から週2回ぐらいは徹夜実験をしていた自分は、同じような実験をして給料をくれる会社とは、なんとありがたいところかと思ったものだった。

CALTECHに留学（1990年）して思ったのだけど、こんな優秀な奴ら在必死で努力しているのだから、のんびり日本の感覚でやっていたのでは絶対に勝てない。

出世のためのくだらない努力に費やす時間などあるはずない。

まー、特に化学系の仕事は、ある時期、本当にブラックに集中する必要があったのだと思う。それができたから今の自分があるのだと思う。定時に帰れって言われる研究者は、どうしても可哀想って思ってしまう。

留学から33年経てみれば、ウン、確かにゆとり教育やら働き方改革で、楽そうにはなった。でも世界の競争にはついて行かぬ、給料は全く上がらず、生産性は落ちる一方。その中で同じアウトプット出そうとしたらダイハツみたいにズルするしかない。そんなズルが色々噴き出してきているのだろうか。

いつの間にか、ゆとり教祖様を「裸の王様」と崇めたてまつる社会に変貌していた。ゆとり教祖様って、日本を崩壊させるための、外国の破壊工作員ではないかと思ってしまう。

pirika.comでは従業員に残業はさせない。定時以降の自己啓発には、本業の200%増しの手当を出そう。

まー、従業員は居ないのでなんとでも言えるが。経営者の私？ もちろん経営者には残業規制そのものがない。残業手当もない。ブラックそのものだ。

大学の授業

2024年にはやらないのも視野に入れていたが、横浜国大の春学期7コマ分だけ残ってしまった。データサイエンスの話は、もう若い人へと思つての引退だったが、結局は後任も現役を退いた方が行かうようだ。大学がコストカットしたいのはよくわかる。正規の先生方が超忙しいのもわかる。でも、学生のための授業に時間をかけられなくなったら、教育は崩壊するしかない。

まー、「ガクチカ」に力を入れたい学生と、時間をかけられない先生の馴れ合いで事が進んでいくのだろうな。勉強なんてタイパの悪いものAIに任しておいて人間はもっと人間にしかできないことをやるべきだ。映画なんてタイパの悪いもの、AIに要点だけ3分にまとめて貰らって見ればいい。

こんな事言うと映画好きには怒られるかもしれない。「そんな見方してなんの意味があるのか？」

間違っではいけない。「意味がないからそんな見方」しているんだ。同調圧力で見たくもないものを見なければいけないんだ。

我々サイエンティストだってずーっとそうやってきた。

論文読むときにアブストラクトと結論だけ読んでポイした論文はどれだけあるか考えてみればいい。間のストーリーがどんな優れていようが関係ない。特許だって同じようなものだ。

大学の授業だって、オンデマンドを早送りして何が悪い？

そんな授業でAが取れてしまうなら先生の方が悪い。

他の授業は早回ししても、この授業だけはちゃんととらないとマズイと思わせられないなら、すでにそこで負けだ。

結局、自分だけが「こんなの教育じゃない」って一人カリカリしているのがバカらしくなったのが大学をおしまいにした理由なんだろうな。

それでも4人に1人ぐらいはやる気のある学生はいるし、それはそれで、可愛い。

大学ビジネスにとってはどうでもいい事だろうけど。

書籍のEditor

若い人は、可能であれば、コースDr.できっちり作法を習っておいた方がいいよ。

薬関連のChemo-informaticsに関する洋書のEditorになっている。

自分でも2つのパートを書いている。

1つは「初学者のためのChemo-Informatics手法を用いた医薬品のスクリーニング法」というタイトルだ。

Abstract :

化学の分野でinformatics (情報学) という言葉が使われ出したのは、1998年ごろのChemo-informaticsが初めであると言われている。本書にもあるようにChemo-Informatics用の新しい手法が数多く開発されている。しかし、化学の領域は多様性が高くビッグデータが存在せず、データの信頼性が低いという問題がある。また、開発の目標が不明瞭であることも多い。さまざまなツールを体系的に利用した例が少なく、研究にどう役に立つのかが分

かりにくい。そこでスクリーニング研究にChemo-Informaticsが多用されているとは言えない状況である。本章では、吸入麻酔薬の分子設計を例に、候補化合物をスクリーニングする方法を解説する。まず、分子の構造から物性を推算する。その結果を用いて候補を絞り込む方法とコンピュータに候補化合物を半自動で逆設計させる方法を解説する。

の話だ。

もう1つは「ハンセンの溶解度パラメータ(HSP)を使った製剤」に関するものだ。

Abstract

ハンセンの溶解度パラメータ(HSP)はDr. Hansenによって1967年に発表された。分子間の溶解性に関するエネルギーを、分散項、分極項、水素結合項の3つに分割する。その3つの項を3次元ベクトルと見做す。そしてベクトルの類似度に従って溶解性を判断する。ある化合物(薬)がどの溶媒に、どのくらい溶解するかがHSPによって判断できる。ここでは、医薬品の製剤に対して、HSPという溶解性の指標を利用する方法を解説する。医薬品を合成する際には、目的とする薬剤以外にさまざまな副生成物の混合物になる。これを精製し目的物質を得るには、抽出、蒸留、昇華の技術が必要になる。これらの技術は溶解性の差による分離精製法であるので、HSPの考え方は有用である。また、最近の医薬品は分子が大きくなり、溶解性が低くなっている。そこで結晶化のコントロールがHSPを用いて行われている。シクロデキストリンやポリマーとの相互作用をHSPから予測することが行われている。

両方合わせると90ページ近いボリュームだ。

はっきり言って悩ましい。

僕は論文Dr.だ。コースDrではない。

大学の今でいう、博士課程の前期(昔は修士といった)で卒業して、博士は47歳の時に論文博士としてとった。論文に書くような内容はいくらでもあったが、いわゆるフォーマットには慣れていない。フォントのサイズはこれこれで、一行何文字で、表や図はこういうフォーマットというのを訓練しなかった。

そしてpirikaのページで好き勝手に書いているだけなので、こういう形式ばった文章を書くのはとっても苦手だ。

英語はものすごく楽になった。元々英語の論文はよく読む方だったので、Deep Lの翻訳を読んだときに違和感を感じれば、日本語を直す。で乗り切れたと思う。

では、Springerの格調高い書籍にふさわしいかという、難しいな。

書籍はこれで最後にしよう。

若い人は、可能であれば、コースDr.できっちり作法を習っておいた方がいいよ。

こればっかは後付けではうまくいかない。

プログラミング：

「まだ見ぬ、やりたいことを発想し、形にする」

このところ、Excel上のプログラムでTypeScriptを使う事が多い。まーこれはJavaScriptとほとんど同じなので今後も使い続けるだろう。JAVAのプログラムは昔作ったプログラムを動かすのによく使うようになった。

差し当たって、HSPiPのMIライセンス相当を作ろうとしている。C#のexeファイルを作るか、JAVAで作るか、Nodeで作るか。悩んでいたらChatGPTがすぐに雛形を作ってくれた。これなら皆んな作って仕舞えばいいか。

プログラミングの動機ってなんだろう？

私の場合は「こんなこといいな、できたらいいな」と言うドラえものの歌ではないけど、必ず先にやりたい事がある。まだ大学生だった頃(1984年)はコンピュータで分子模型を描きたかった。そこあたりから計算機化学にのめって行った。

だから、プログラム言語の書籍を頭から勉強するってことはした事がない。

やりたいことを、その部分だけ学ぶ。

これまで、それで済ましていた。だから使うコマンドもほとんど限られたものだけだ。

それでもどうにかなっている。

さらに有難いことには、最近、足りない部分はChatGPTが補ってくれる。

まだ見ぬ、やりたいことを発想することだけに集中すればいい。

いいものを見て、あ、これを自分の系に取り込め、って感じればいい。

それを、今までは本を買ってきて読んだり、ネットを探したりしていたのに、生成系のAIのおかげで、とても楽にできるようになった。

GPTsのこともあるし、有料版に乗り換えた。

Visual Studio for Mac

統合開発環境 (IDE) 「Visual Studio for Mac」の提供を2024年8月31日に終了すると発表した。

C#で開発しようとした矢先にこれだ。悩ましいな。

Visual Studio for Macを使ってGUIありのアプリを作るには？

Windows用のVisual StudioでFormアプリを作る。一旦それがちゃんと動けば、後はVS for

Macでその.slnをひらけば、csのプログラムは変更しても、コンパイルし直せばexeファイルが作れる。作成したexeファイルはwindows11でそのまま動かすことができる。

parallelsでMac上でwindowsを動かしていれば、作ったプログラムはwindowsで動かせる。本来の使い方は、Macで動かすにはGUIはMac用のXcodeで作る。でもXcodeは対応するバージョンなどで制限が多く、VSから対応するするのは難しい。windowsだけで動けばいいなら、MicrosoftとしてはVS for Macを作る意義はないよな。

でも、こんな風にIDEをやめられてしまうと、安心して開発に使うことはできない。差し当たっては、.Net MAUIがあればiPhoneなどに開発は続けられるかもしれないが、何度も切り替えるのは面倒だし、自分の作るソフトはiPhoneはターゲットにしていない。JetBrainsのRiderにでもしようか？

GUIがなくいいなら、コンソール・アプリを作ることになる。
Windowsであればターミナルを起動して、C>からexeプログラムを走らせる。
これはVSでC#のコンソールアプリを選んでプログラムを作成すればいい。

Macからは、VS for Macで”Webとコンソール”、コンソール・アプリケーションC#を選ぶ。
XCodeなんかなくても、Unixアプリが作成される。
これをダブルクリックすると、自動的にターミナルが立ち上がり、プログラムが動作する。

同じソースコードから、Mac用とWindows用のターミナル・アプリが2つ作成されるので都合が良い。

結局、開発者はいいようにやられる。
Appleが潰れかけた1997年ごろには、Macを買えば、windowsやLinuxのプログラムが作れた。弱い方は強い方に寄り添わなくては生き残れない。それはそれでよかった。どんな環境でも動くプログラミングの勉強になった。

AppleがiPhoneやiPadで強くなると、さっさとJAVAを切り捨て、開発者はXCodeを使うことを強制される。作ったソフトはApple製品でしか動かない。強いところは囲い込みを始める。その分、AndroidやiPhoneにやられたMicrosoftはいろんなマシンで動く開発環境を作る。

このところ、copilotなども含め、Microsoftの取り組みが、Appleより光っていると感じる。するとすぐに、Microsoftの囲い込みが始まる。

その度に、開発環境が使えなくなる。AppleやMicrosoftはキラーアプリが欲しい。ちまちました、サイエンス系のソフトなんてゴミみたいなもので、作ってくれる必要はない。

やっば、JAVAに戻った方がいいかな。

まだ動くうちに、HSPiP ver.6に組み込まれたY-MB24をコンソールアプリにしてみた。

ChatGPTに雛形を作ってもらおう。

C# コマンドプロンプトプログラム

input.txtファイルを読み込む

Output.txtファイルを作成する。

input.txtファイル一行ずつ分割する。

分割された一行ずつDo_Analyze関数に送る。

Do_Analyze関数

受け取ったString変数をタブ区切りでMyData配列に格納する

MyData[1]をSmiles Ana関数に送り、処理した結果をMyResをStringで受け取る

Do_Analyze関数から戻る

Output.txtを開く

MyResを一行ずつOutput.txtに書き込む

Output.txtを閉じる

次の分割された行を繰り返す

全ての行の処理が終わったら終了する

input.txtとしてタブ区切りでSmilesの構造式を準備する。

benzene C1=CC=CC=C1

m-Xylene CC1=CC(=CC=C1)C

これは、ターミナルのプロンプトと同じ位置に置く。

Y-MB24のC#のプログラムを、SmilesAnaに置き、グローバル変数や配列の使い方を

ChatGPTに聞きながら必要な修正を行う。Windows版は同じプロジェクトをWindows用のVSで読み込み、ビルドするだけだ。

1時間ぐらいで、マックでダブルクリックすると、input.txtを読み込みY-MB24の計算をして結果をOutput.txtに書き込むアプリが出来上がった。

これでCLIライセンスと同じようにY-MBをバッチで計算させる事ができる。

後は、Y-MB24に搭載されなかったMI用のDescriptorを計算/出力するように変更すればP-MBが完成する。これで、MacでもWindowsでも動作するアプリが完成する。

普通に本で調べながらなら3日仕事だ。

C#版のP-MBが完成した。

最近読んだ本

最近、Copilot関係の書籍にハマっている。私は紙の書籍派だけど、ついにKindle本にも手を出してしまった。何をしたかったか？というと、Copilotの本質は自然言語を扱える生成

系AI, ChatGPT。このChatGPTに自分の持つ資料を別に付け加えると、自分独自のGPTsが作れる。「これまでの大学の授業の資料に沿って、今年7回分の講義資料を作って」と頼む事ができるようになる。そう聞いて、この1年間、昔のパワポなどの資料を整備した。一番古いものは1992年か。32年で1500個、9.4Gのパワポになる。年平均50個作ってきたのか。pptxになる前のpptを新しい形にしたり、特に、Macの古い画像、pict形式が見れないものを直し、Excelとの連携が崩れたグラフを直した。やっと準備できたところに、やっとCopilot登場!?

でも、月30\$えー。

どうしようか悩んでいたら、ChatGPT Plusの方でGPTsの形で対応された。PDFやらWordやら画像などをどんどん学習させてしまう事ができる。

こうなってくると、

「やりたいことを持っている」

「広い範囲に興味を持っている」

「過去の膨大な資料を持っている」

者は強いよね。

格差が広がるわけだ。

アマとプロの戦い

ゴルフなんかでもそうだけど、同時では無いにしろ同じ土俵でアマとプロが戦う。ルールも変わらない。1ホール目で大叩きしたら、後はどんなに頑張ってもリカバリーできない。貯蓄から投資へ。

それはそれで良いと思うが、アマとプロが同じ土俵で戦うのは同じだ。

20年持っていれば平均したらプラスは良い。

でも、病気になってお金が必要になったときに、たまたまその時にマイナスに凹んでいたらどうなるか？

施設に入ろうと思った時に、リーマンみたいなことが起きたら？

基本的に若い人は時間がある。必要となるのが先なら生活費まで投資しなければ大丈夫だろう。

でも自分ぐらいの年寄り微妙だよな。

ただでさえ、この頃、物忘れが激しい。施設に入ろうかと思う頃には、痴呆で自分のお金を下ろせないなんてことになりかねない。ましてやその時に暴落してたら？

欲の皮がつっぱったプロが市場を歪めているのだから、アマなんてひとたまりもない。

そのうち、一旦リセットが入るはず。（永遠に上がり続けることはできないので、誰かにババ掴ませて、プロは逃げる）

デイトレーダーみたいなプロは好きにさせれば良い。

でも、長期保有のアマにはプロにない特権を与えるべきだ。

10年以上持っていた場合には、10年以降は、前の5年の平均利率見たいのにするとか。さもないと、きっとババをつかまされる。

EVって本当に環境に良いか？

日本のように電気が火力発電で作られる割合が高いなら、EVにしたって炭酸ガスを排出する。太陽電池と風力だけで必要な電力が賄えるならどうだろう？ 本当に環境にいいのだろうか？

僕は、ガラス会社に勤めていて、ガラス設計の神様のような研究者と仕事をした事がある。僕はコンピュータによるガラス組成設計の担当だった。

その研究者が退職するときに言った。

「自分の40年の経歴の中で、とうとう窓ガラスの組成は変わらなかった」

1000年持つと実証されている組成を、強度が高くなるとか、透明性がどうのとかいくら言ってもそれは変わらない。

コンピュータで組成設計を狙うなら、携帯電話用の有機ELパネルとかを狙うとかしないとダメだと言われた。そんなものは10年も使う奴はいない。

カルフォルニア工科大学に留学していた時にも、スタッフに言われた。なんでアメリカ人がトヨタが好きかといえば、50年たっても問題なく動いていると言う実績があるからだ。

つい最近、20年使った冷蔵庫、エアコンがお釈迦になった。

電気店でははっきり言われた。今度のものは持って5-6年だと。

その当時のものと比べ、電気コストは半分ぐらいになるからすぐに元が取れますよとも言われた。

ウリをそこに置くと破綻する。5-6年ごとにエネルギー消費を半分にはできないからだ。

メーカーは売れなくなると困るから（昔は日本で型落ちしたものでも輸出の道があった）OSを入れ替えて古いものでは動かないとか言い出す。

100年持つ黒電話と、3-4年で買い替えないといけないスマホでどちらがエコだろうか？

50年持つトヨタの車と、7-8年しか持たないEVでどちらが本当にエコだろうか？

SteraのEVはiPhoneのような車とか言われているが、その製品サイクルで買い替える事ができる金持ちがどれだけいるのだろうか？

なぜ、EVが高価なのか？ 作るのに多くのコストがかかるから。そのコストは必ずエネルギーの消費を伴っている。iPhoneと同じで、EVはぶつけたりするとすぐに全損になってしまう。長持ちしない仕組みになっているので保険代が高くなる。

中古のiPhoneを買う人が少ないように、EVの中古市場が広がらなければ、EVブームは、一部の金持ちの道楽で終わる。2024年、そろそろ結果が見えてくるだろう。第一世代のEV車の電池の耐用年数は確か5年だもんな。

投資家の一部が、ゲームチェンジを図って、昔ながらの安く長く使える商品を、エコでないから使うなって言っているだけだ。いつまでもガラケー使ったり、古いテレビを使い続けられていると儲からないから、地球のためという錦の旗のもと高いものを買わせようとしているだけだ。株もそうだが、ある程度成熟してしまうと、一度リセットかけないと儲けがなくなる。昔のような大量生産、大量消費を相変わらず求め、他の国にも求めさせる。

そこで、ノリの悪い日本が気に入らないのだろう。寿命が短く高価なEVを欲しがらない。そもそも、ワーキング・プアーな若者が車を欲しがらない。車を持たないのが究極のエコだし。

でも、車のリコールとソフトウェア・アップデートの関係はよく学ぶことだ。

pirikaの研究会/サイエンスカフェ

セブンカルチャークラブというのが溝口にある。

僕がレンタル・オフィスしている神奈川サイエンスパーク(KSP)も溝口だ。

カルチャークラブで、いろいろな講座に混じって「こども科学実験」という講座がある。

その講座の先生が、なんと昔の会社の同僚の奥さんだった！

なんとという奇遇！

なんか面白いことできないかなー。

なんて思って、カルチャークラブ+プログラミング+講師で検索してみると結構いっぱいある。この角川ドワンゴ学園ってN/S高じゃん。楽しそう。給料もすごくいい。

無料で変なことやっていないで、働きに出ようか。。。。

高校の情報の教科書ガイドはI・II買ってきて読んでる。

難しいこと習っているけど、こんな事ができたってChatGPTには勝てない。

やらないよりはマシという程度。

自分は幸い、稼げるプログラマーだ。プログラミング技術が高いからではない。化学に特化しているから食える。そんな技術を社会人や大学院生に教える？ってやってきても限界を感じる。

でも、こどもまで戻ると、今度は化学が障害になる。

どうするか悩んでいても始まらないので、枠だけ確定してしまう。

月1回、ZOOM開催、(多分、夜21:00ぐらいから1-2時間)

月1回、Tech-Pot, リアル開催(多分、15:00ぐらいから1-2時間) 来られない人用にZOOM
接続

費用：学生、元学生0円、社会人5500円/月(6ヶ月分まとめて払い。いわゆるチケット制で参加できなかった場合持ち越し)、コンサル対象の会社2名まで無料参加。

内容：企業向けの講習会やテクニカルサービスを安く受ける場ではない。リスキルするため、生産性を上げるため身銭を切って勉強する場だ。山本が一方的に講義する場ではなく、皆が発表し合う場だ。所属はオープンにしてもらおうし、NDAは結ばないので、オープンにできる内容のみを扱う。

プログラミング、処方設計、HSP関連、(最近多いのが)ラジカル重合、気液平衡、物性推算など(他に希望があれば)

何時から？

参加したい人が現れたら。

参加者同士のバッティング：あまり直接的に同じ分野の方が参加を希望された場合は、お断りするか、時期をずらしていただく事がある。