

生成AI時代において、なぜ今も子どもたちに“プログラミング教育”が必要なのか

株式会社TENTO 代表

竹林 暁

takebayashi@tento-net.com

自己紹介

- 株式会社TENTO代表 竹林 暁
- 黎明期から子供向けプログラミングスクールを始める
 - 草野真一と共同で2011年にTENTOを創業
- 現在はスクール事業だけではなくソフトウェア開発にも注力
 - オンライン授業ツールnoiz、AI先生アプリChatStudyなど
 - Cursor、Claude Codeなどのおかげで現役プログラマ
- 大学院では言語学を研究していた。
 - メタ思考（人間の考えをどう考えるか）が研究テーマ

むかしばなし：2011年当時の話

- スマートフォンが普及してきたころ
 - 2011年に携帯電話の中でスマートフォンの割合が50%を超えた
 - iPhoneが2008年発売（日本）
 - アンドロイドが2009年発売（日本）
- 小学校のプログラミング教育必修化は2020年！
 - 正式に告知されたのが2017年

むかしばなし：2011年当時のプログラミングスクール

- まだ市井の子供向けプログラミングスクールはLife is tech!さんとTENTOしかなかった
- プログラミングがどういうものか一般の人には知られてなかった
- つまり、プログラミング自体ではなくプログラミングすると何が良いことがあるかを訴求するしかなかった
- **実は当時と現在は状況が似ている！**

プログラミング教育でAI時代に価値がなくなってしまうもの

- 具体的なプログラミング手順を覚えること
 - 現在ではもはや
 - Pythonの仕様書＝エアコンの説明書

プログラミングは実用性を売りにできなくなった

- 誰でもプログラムを作れてしまう

AI時代でも価値がなくなるもの

- プログラミング教育が子供たちに及ぼす良い影響
 - 試行錯誤を怖がらなくなること
 - ルールについての視座
 - 思考の明確化のトレーニング

プログラミング学習の良い影響：試行錯誤を怖がらなくなる

- プログラミング＝ミスをしながら覚えるもの
- このため試行錯誤が普通になる

なぜ試行錯誤を怖がらなくなることが重要か？

- AIが人間を代行するようになるので人間はますます安全になる
- 現代は「チャレンジ精神＞安定志向」

試行錯誤の要素をさらに伸ばすためのアイデア

- オープンな教材にする
 - 答えがある教材＝答え通りじゃなきゃダメというメッセージ
 - 教材にないことをコーディングするから間違う
 - 例：
 - 「Processingでヨーロッパの国旗を描いてみよう」
- 閉空間で学ぶ
 - 自分が自由にできる空間
 - テキストプログラミングを最初は学習しない
 - できることが限定的で目に見えているツールを使うべき
 - 例：ビスケット、スクラッチ、マイクラ、IchigoJam

プログラミング学習の良い影響：ルールへのメタ的な視座

- プログラムを作ること＝ルールを作ること
- 世の中のルールや法律も人間が作ったものであることに気づく
 - ルールや法律も改変可能と考えるようになる

ルールへのメタ的な視座をさらに獲得するためのアイデア

- 子供同士のコミュニケーション
 - 自分で作ったゲームをほかの子供にやってもらう
 - 「こんなのクソゲー！」
 - 共同で開発する
 - サーバ側を作る人とフロントエンドを作る人
 - マイクラのfunctionsを作る人とオリジナルブロックを作る人
- 生活をプログラミングで表現する遊び
 - 学校の1日をプログラムで表現する

プログラミング学習の良い影響：思考の明確化のトレーニング

- プログラミングとは
 - 頭の中にある思考を機械の思考に翻訳すること
 - 翻訳する過程で思考が明確化する

思考の明確化をさらに伸ばすためのアイデア

- 発表の時間を設ける
 - 「どうしてこのように作ったか」という言語化を鍛える
- 作る前に設計書を作る
 - まずは日本語で思考を整理する
- リファクタリングを重視する
 - プログラムの読みやすさを考えさせる

今後のプログラミング教育のありかた

- 水泳やそろばんに近いものになる
- プログラミングを通して精神を陶冶する

プログラミングとプログラムは違う

- プログラミング＝人間がプログラムを書くこと
- 人間が書かなくてもコード（プログラム）は大事

コンピュータサイエンスはもっと学習したほうが良い

最後に

変わるものもあれば変わらないものもある

私たち自身が変化を楽しみましょう！