



# プログラミングの 世界の歩き方

## 「リバースエンジニアリング」

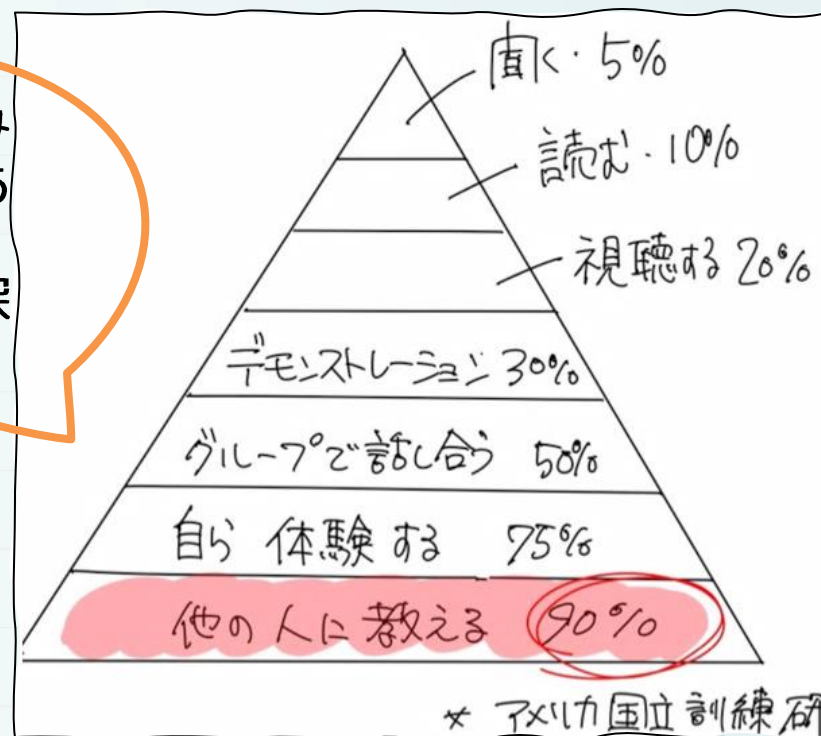


# プログラミングの世界を歩こう！

プログラミングの世界を知るにはその世界で使われている  
ことばを知ることが大切だ。

これはプログラミングに限らず、スポーツでも音楽でも何かを  
習得するには、その世界のことばを知ることから始まるよ。

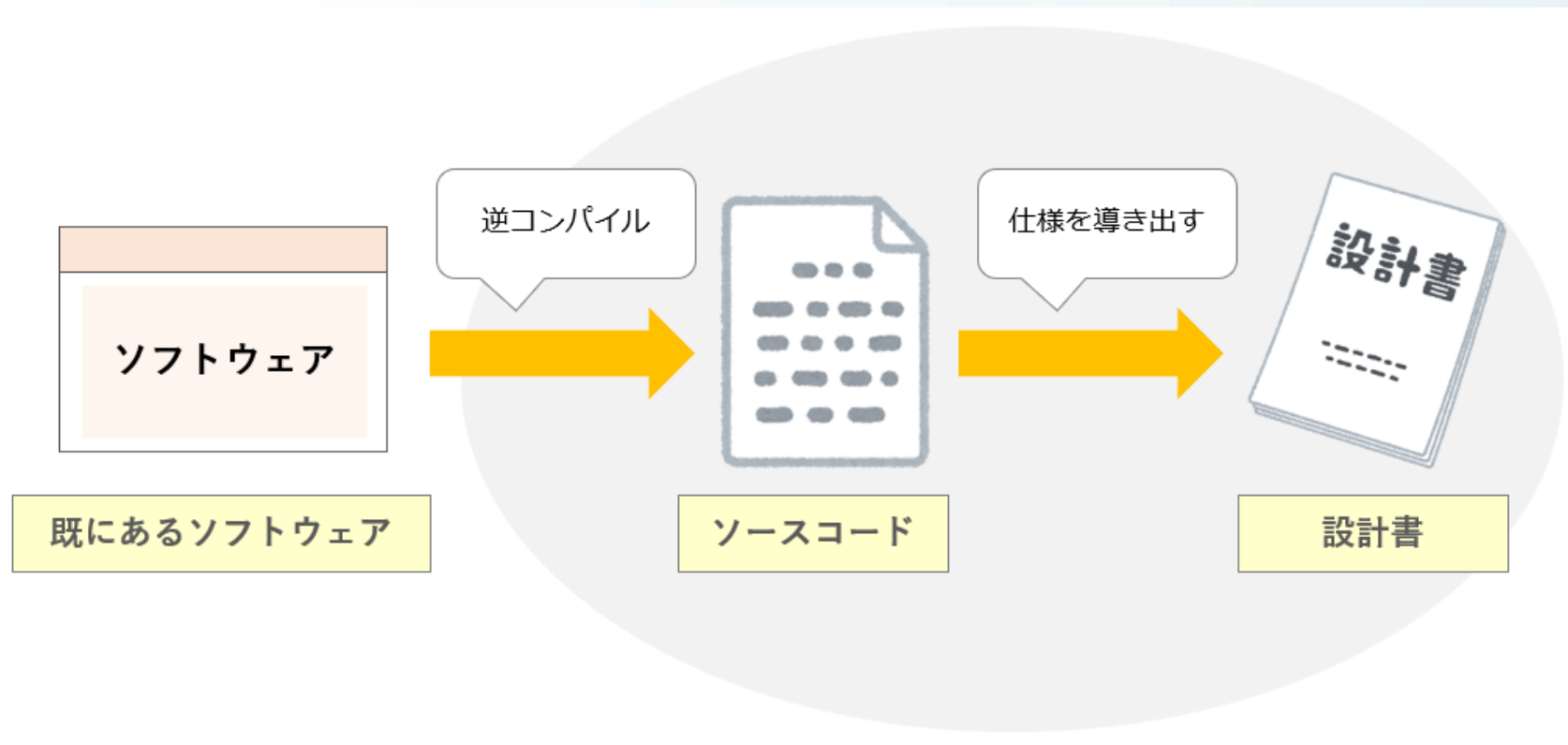
ラーニングピラミッド



学んだことはみんなに教えてあげよう！  
もっと理解が深まるよ！

# リバースエンジニアリングって？

リバースエンジニアリングとは、動いているプログラム(製品)から仕様と設計を探ること。「リバース(reverse)」には「逆」という意味があり、「システム開発の手順」を逆にすすめていく方法。製造業の世界でも行われている手法。



プログラムをソースコードに戻すことを逆コンパイルという。



# リバースエンジニアリングって？



本来のシステム開発の手順は

1. 外部設計や内部設計で、「どんなプログラムにするのか？」を決める
2. 設計した通りに、ソースコードを書く
3. 実際に動くプログラム（製品）が完成する

リバースエンジニアリングは、これの「逆」の手順をふむので

1. 実際に動いているプログラムをもとに
2. プログラムを逆コンパイルしてソースコードを出す
3. ソースコードをもとに、外部設計や内部設計をする

「ドラクエを作りたい」というのを決めて  
(要件定義)  
プログラムで「どう実現するのか？」を決める。  
(基本設計・詳細設計)  
ソースコードを書いてプログラム(製品)にする  
テストして、世に製品として出す。



世に製品として出されている「ドラクエ」をみて  
ドラクエのプログラムをソースコードに戻す。  
ソースコードを見て、「どう実現されているの  
か？」の設計を探る。  
「ふむふむ。こうやって『ドラクエ』が作られてる  
んだな」ってなる。

# リバースエンジニアリングのメリット

リバースエンジニアリングのメリットとしては、以下の通り。

## ■開発にかかるコストと時間の削減

プログラムを一から作らないためコストと時間の削減が可能。

通常はしっかりと設計図を起こしてから開発を進めても、開発段階で問題が見つかることも多く、何度もトライ&エラーを繰り返す必要がある。

しかし、既製品の分析によってトライ&エラーを繰り返した結果の技術を参考にすることができ、開発にかかるコストと時間の削減が実現できる。

# リバースエンジニアリングのメリット

## ■製品の文書化

古い製品の場合、設計図が手元に残っていない可能性がある。

古い製品に使われている技術をそのまま応用し、新たな製品として開発を進めたい場合、リバースエンジニアリングによって分析することで、内部回路や設計が把握できる。

調査した結果を文書化し、さらなる改良や今後の維持管理のために役立てられる。

他の部門

# メモ



# プログラミング教室の テクノロ

なまえ：