



# R流



## 職業プログラマー向け 生産性向上の工夫

2007年6月2日 R・田中一郎

<http://blogs.wankuma.com/rti/>

# 自己紹介

名前: **R**・田中一郎(略してRさん)

職業: IT関連の、いろんなこと。

まとめ

果てしなく広く、浅い知識を持った人？

# 今回のテーマについて

## R流

Rさんは無駄に長い経験がある。

## 職業プログラマー向け

ソースコードを作成することで収入を得ている人。

## 生産性向上の工夫

ソースコードの生産性を向上するための工夫。

# 生産性向上の秘訣とは？

— あなたは神様を信じますか？ —

# 会議や外出は 15時を中心に

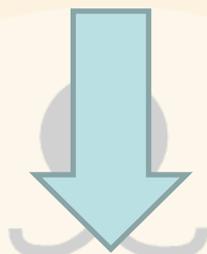
仕事時間中の効率

| 9  | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20            | 19 | 20 |
|----|----|----|----|----|----|----|----|----|----|----|---------------|----|----|
| 最高 |    |    | 休  | 普通 | 最低 |    | 高い |    |    |    | 普通<br>→<br>低い |    |    |

そして、最も複雑な仕事は、午前中に。

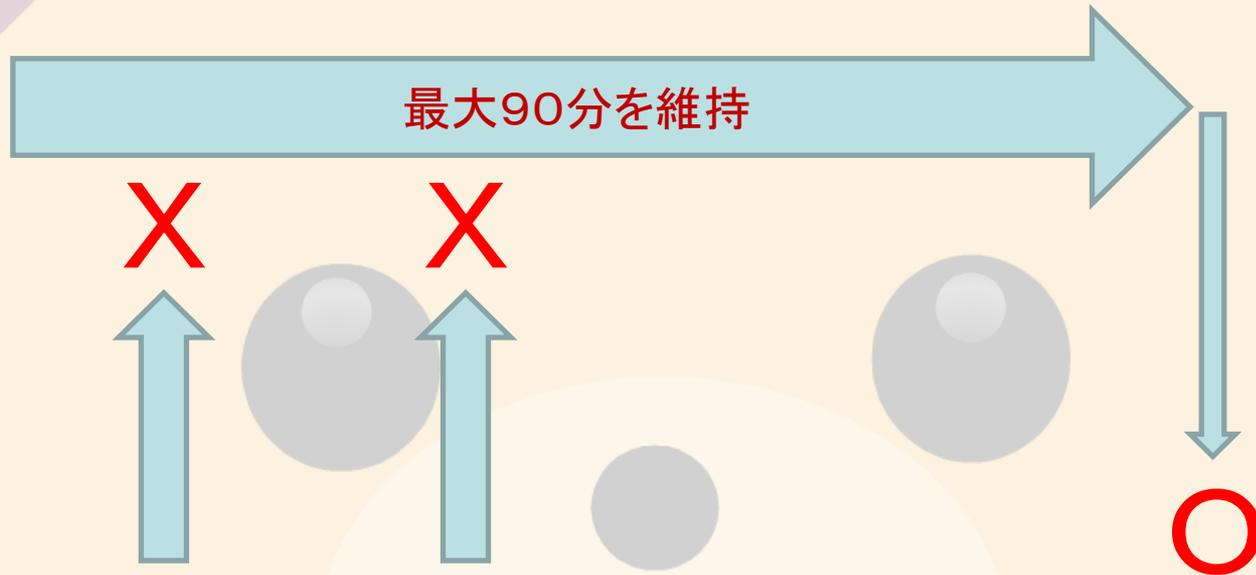
# 休日明けにすぐ仕事開始

長期休み明けの午前中を無駄にしない。



休み前に調整しておく。

# 持続させる環境を作る



相手からの要求をできる限りブロック  
連絡は、自分のタイミングで行う。  
音楽(歌なし)を聴くのも良い。

# 2つのバランスを考える

- 最初に苦勞すべきか？
- 毎回苦勞すべきか？
- 大多数がそうであるか？
- 小数に問題が生じるか？

# 頭の中の情報を素早く出力する

- 考え込むより、とにかくコーディングする
- ブラインドタッチをマスターする
- キーボーダーになる
- 高速なPCで開発する

# できる限り入力しない

Q.最も早くソフトウェアを開発するには？

A.できる限りコーディングしないで完成させる

- デザイナの利用
- Microsoft .NET Framework の利用
- 汎用クラス・メソッド・モジュールの管理
- 類似コードのコピー

# 直感的な表現を使う

未来の自分は、他人と同じ。  
他人にわかるように書けば、  
未来の自分にもわかる。

…この表現は直感的ですか？

他人が見てもわかりやすいコーディングをするように普段から心がけましょう。  
何故なら、他人が見てもわかりやすいというのはもちろんですが、  
数日後、数ヶ月後に、継続して開発している自分が見ても、わかりやすくバグを未然  
に防ぐことができます。

# 日本語を有効活用する

```
if (Is税込み金額()) {  
    return 金額;  
}  
else {  
    return Calc税込み金額(金額);  
}
```

このコードに説明やコメントが必要ですか？

# 今の開発環境に応じた表現

- 古い慣習なんて、捨て去ろう！
- 今開発しているソフトウェアは、今の開発環境以外で開発することがありますか？
- あるとしたら、どれくらいの比率ですか？

# 全体を見据えて確実に終わらせる

- 考え事は大雑把に全体を把握。
- 行動は、処理の単位を細かく分けて、一つ一つを確実に完了させる。
- 必要に応じて全体を再度調整する。

ご清聴ありがとうございました。

皆さんも、月曜日から、

是非、

実践してみてください。

<http://blogs.wankuma.com/rti/>