

わたしはこれまで、お恥ずかしながら、あまり「料理をする」ということがなかった。飲食店でバイトをしていた頃はあったが、カラスカレイのムニエルを焼くだとか、パスタ用のトマトソースを仕込むだとか、ピザを焼く程度のルーチンワークであり、"料理"という感じではない。

一人暮らしになるので、当然であるが、食事が自動的に出てくることはない。もちろん床ドンしたとしてもダメだ。そこで、わたしは本格的に自炊を始めた。料理を始めたのだ。ここではハッカー文化的な立場から、「料理・自炊とは何か」を考察していこうと思う。

2. 自炊に向いてる人・向いてない人

合理性を追求すれば、自炊というのはあらゆる面で「悪」である。以下の点で自炊は無駄だらけだ。

- ◆ 一人分を作るのは、結果的に材料費が高つく。
- ◆ 水道代や光熱費もそれなりにかかる。特に熱器具の消費エネルギーが大きいので侮れない。
- ◆ 時間を浪費する。お料理を作り、洗い物などをしていると、あっという間に1時間は飛んでしまう。
- ◆ 体力を使う。特に、慣れないうちは効率のよい動きができず、全てを終えた後に妙な疲労感に襲われる。

ハッカー文化に傾倒する人は効率を重視するきらいがあるので（ハッカーとはそういう生物だ）、これらの無駄を許容できない人も多いだろう。そういう人は自炊向きじゃない。ただし、ここで考えてほしいのが、有名ハッカーの多くがコンピュータ以外の趣味を持っているということだ。たとえば、（真の）フリーソフトの教祖として有名なりチャード・ストールマン氏。彼は音楽やSF作品に傾倒していて、自身でSF作品を書いたこともあるそうだ。

だから、**料理を趣味にして、料理自体をハックすることを楽しめるような人が自炊向き**だといえる。自炊に慣れてくると、「もっと効率的にジャガイモを茹でるにはどうしたらいいか？」とか、「醤油の味が気に入らない。別のライブラリ（醤油）を使ってみよう」とか、そういう考えが頭にあふれてくる。これがとても楽しいのだ。

3. 自炊とはビルド作業である

自炊は、makeと本質的には変わらない。料理をしているあなたは、ズバリmakeコマンド自身だ。食材および調味料はライブラリやリソースであり、お皿はファイルシステム。鍋やフライパン、計量カップや包丁などはコンパイラ。レシピはソースコードやMakefileである。

- 1 あなたはソースコードやMakefile（レシピ）を読んで、ある料理をビルドする方法を理解する。
- 2 ライブラリ（食材）を調達する。残念ながら、料理におけるライブラリのほとんどは非フリーで、不自由なプロプライエタリだ。製造過程はブラックボックス化されているし、有償である。選ぶライブラリによってビルドの難易度が変わるし、ビルド結果のクオリティも変わる。
- 3 調達したライブラリを、コンパイラ（調理器具）を使って、炒める、煮る、焼く、揚げる、レンチンするなどの方法でコンパイルする。
- 4 コンパイルしたオブジェクトを、混ぜる、上からかける、一緒に炒める、一緒に煮るなどしてリンクする。
- 5 ファイルシステム（お皿）に、1-4の工程で生成された実食可能なオブジェクトをmake installする。余剰な実食可能オブジェクトは、その他のストレージ（冷蔵庫や冷凍庫）にアーカイブしておくこともあるだろう。
- 6 make cleanをする。洗い物などをして、キレイキレイするまでがお料理のビルドである。

4. 自炊の楽しみ方

前節の説明で、あなたがmake自身であることを実感してもらえたと思う。料理のビルド時間やビルド結果のクオリティは、makeであるあなたの手腕に依存する。

ハッカー的自炊の楽しみ方とは、**料理をハックをすること**である。これに尽きる。

スーパーマーケットで良質なライブラリやリソースを探したり、クックパッドでおいしそうなソースコードを発掘したり、自分でソースコードを作ってビルドしてみるのも楽しい。優秀なコンパイラを見つけるのもよい。経験を積んで、料理の勘を磨き、手際よくビルドできるようになると、もっともっと料理がエキサイティングな体験になってくる。

自炊とは、ハッカー向けの素晴らしい趣味なのである。

5. おまけ

わたしのビルド日記は <http://mozacafe.quellencode.org/> にまとめている。

gitでレシピを管理して、みんなでフォークしたりすると楽しそうだと思っている。