

第一章

プログラマの呪文

普段使っているインターネットエクスプローラやアウトLOOKエクスプレス、ワードやエクセル、はたまた年賀状ソフトなど。これらはみんな「プログラム」だ。そして、これを作るのがプログラマ。ここでは、プログラマとプログラム、それを実行するコンピュータの知られざる世界をご紹介します。

第1章
プログラマの呪文

ADDRESS	00	01	02	03	04	05	06	07	08	09	0A	0B	0C	0D	0E	0F
0000210	03	E8	00	4C	CD	21	50	53	51	52	57	56	89	16	04	00
0000220	89	16	02	00	8B	36	02	00	AC	22	C0	74	25	FF	06	02
0000230	00	E8	2E	00	72	0A	8A	E0	AC	FF	06	02	00	E8	03	90
0000240	B4	80	E8	5A	00	E8	88	00	8A	26	0A	00	22	E4	75	0E
0000250	EB	D2	E8	2F	00	8B	16	04	00	EB	C5	5E	5F	5A	59	5E
0000260	58	C3	50	53	51	52	57	56	3C	20	72	0C	3C	DF	77	08
0000270	3C	80	72	08	3C	9F	77	04	F8	EB	02	90	F9	5E	5F	5A
0000280	59	5B	58	C3	50	53	51	52	57	56	B9	03	00	E8	40	81
0000290	E8	0C	00	E8	3A	00	E2	F5	5E	5F	5A	59	5B	58	C3	50
00002A0	53	51	52	57	56	80	FC	80	74	03	E8	12	00	8B	D0	B4
00002B0	14	8C	D8	B9	0B	01	CD	18	5E	5F	5A	59	5B	58	C3	02
00002C0	E4	2C	1F	78	04	3C	61	14	DE	05	A1	1F	25	7F	7F	C3
00002D0	50	53	51	52	57	56	A1	0B	01	80	FC	01	74	09	C8	06
00002E0	01	00	0F	90	EB	07	90	C8	06	01	00	07	90	A1	0B	01
00002F0	80	FC	01	74	15	A0	01	00	E8	35	00	E8	68	01	A0	01
0000300	00	E8	2C	00	E8	5F	01	EB	13	90	A0	01	00	E8	63	00
0000310	E8	53	01	A0	01	00	E8	5A	00	E8	4A	01	80	3E	01	00
0000320	00	74	06	FE	0E	01	00	EB	C4	5E	5F	5A	59	5B	58	C3
0000330	50	53	51	52	57	56	C6	06	00	00	00	90	8A	26	00	00
0000340	BA	0D	01	E8	70	00	73	0D	8A	26	00	00	80	C4	04	E8

0~9の数値とA~Fのアルファベットの羅列がプログラム。タダの暗号にしか見えない

プログラマは魔法使い

「プログラム」は、パソコンにいろいろな作業をさせる手順書だ。しかし、普段パソコンを使っているぶんには、この手順書を目にすることはまずない。そもそもこの手順書は、どのようなものなのだろうか？

ワードやエクセル、年賀状ソフトなどのソフト。ウィンドウズや周辺装置を追加したときに必要になるソフトもプログラムだ。便宜上「アプリケーション」や「ドライバ」、「OS」などと呼ばれている。もちろん、銀行のATMやおいしいご飯が炊ける電子ジャーも、すべてプログラムという手順書があるからこそ、機械の中に入る小びとさんの行員も炊飯職人も必要ない。

これらプログラムの実体は、図のような数字とアルファベットの羅列である。そしてこの数字ひとつひとつには意味があり、パソコンの頭脳であるCPUがこの数字を読み込んで、計算を行っている。

CPUの中には、これらの数値と処理の内容を意味付ける一覧表がある。数値を読み込んで「この数値は、今ある数値に次に続く値を足し算するのだな。ふむふむ」と理解して、計算しているのだ。実際には、こんな悠長な話ではなく、

何マイクロやナノ秒という短時間で計算している。

人間が反射的に動作したとしても、目でモノを確認してから実際の動きをするまでに0.5秒かかるといわれている。この間にパソコンは、700,000,000（7億）回の計算を行っている。世界一を誇る国産地球シミュレータのスーパー、いやウルトラコンピュータにいたっては、20,000,000,000,000（20兆）回だ。

さてこの数値の羅列だが、人間がポツポツ考えながら入力している、次のウィンドウズが発売されるまで何十年もかかってしまう。それはそれで、ユーザーとしては金がかからなくていいのだが…。

そこで考え出されたのが、ほんのちよつとだけ人間にわかりやすい呪文、アセンブラ言語だ。

かなり無理があるが、英語っぽい

数字とアルファベットの羅列

日常使っている数字は10進数だ。0~9まで数えると位が上がって10になる。ところがコンピュータの世界は16進数が基本。0~9まで数えると、次はA、B、C、D、E、Fと続くのだ。16進数のAは10の意味、Fは15の意味。Fまで数え終わると、はじめて桁があがって10になる。16進数の10は、10進数でいうと16。この本のページ数を見てもらいたい。10進数のページの横にあるのが16進数での表記。慣例で最後にHをつけるが、¥×××と表記する。そのときの1と0だけの数値は、2進数表記だ。

0.5秒

運転免許を持っている人ならお分かりだろう。人が飛び出してきた「危ない！」と思ってからブレーキペダルを踏むまでの時間だ。時速60kmで走っているれば、この間に車は8m進むというアレだ。これに制動距離が加わって約27m。だから道路をポールが横切ったときは、その時点でブレーキを踏まなきゃ、後から飛び出してくるお子様はサヨナラというわけ。

コンピュータが理解できる数値に変換

変換だからConverter（コンバータ）かと思いきや、資料を集めるという意味のCompiler（コンパイラ）と呼ぶソフト

第1章
プログラムの呪文

```
Private Sub Command1_Click()
    Dim a As Integer
    Dim b As Integer
    Dim c As Integer
    a = 1
    b = 2
    c = a + b
    Print c
End Sub
```

初心者にも分かりやすいBasic言語。

```
#include <stdio.h>
#include <stdlib.h>
main() {
    int a, b, c;
    a = 1;
    b = 2;
    c = a + b;
    printf("%d", c);
    return 0;
}
```

Basicよりすこし複雑だがアセンブラよりはるかに人間らしい

これ以外にも、J A V A や C G I、
コンピュータに計算手順を教える
のである。
うにプログラマは、ある言語を使って、
が、これらの言語なら簡単だ。このよ
先の10倍以上を書かなくてはならない
アセンブラで計算結果を表示するには、
この10倍以上を書かなくてはならない

ともいえる。この1文が、先の数字ひとつひとつに対応しているのだ。このように、言語のようなものをプログラマが書き、最終的にはコンピュータが理解できる数値に変換している。
たとえば、ある2つの値を足し算する場合は、`mov ax,A mov bx,B add ax,bx` `mov C,ax`と記述する必要がある。そこで人間が理解しやすいように `A + B = C`と記述すればよい、さまざまな言語が開発された。それが `Basic` 言語や `C` 言語である。

一時はジャバジャバ言っていた。ところが、マイクロソフトが勝手に機能強化するなど、方言ができてしまい、本家の `J A V A` を「100% Pure Java」として区別するようになったのだ。サンマイクロに言わせれば、マイクロソフトの `J A V A` は海賊品というわけ。今は、大型コンピュータのデータを `Web` で公開するためのシステムなどで人気を盛り返している。

H T M L
HyperText Markup Languageの略。できることは限られているが、`Language`なのでやっぱプログラム言語の一種。
`Web`でおなじみ。

```
-----#
:      文字列を表示する#
:      #
:      in    dx      文字列が格納されているオフセットアドレス#
:      #      なお文字列はASCII文字列#
:      out   none#
:      break none#
:      -----#
put_string proc#
            push_all#
            mov   org_str_ptr,dx#
#
#_003:      mov   str_ptr,dx#
#
#_002:      mov   si,str_ptr#
            lodsw#
            and   al,al      ; 文字列の最後か?#
            jz    _001#
            #
            inc   str_ptr#
            inc   str_ptr    ; 次の文字にポインタを進める#
            #
            xchg  ah,al#
            call  get_sys_font#
            call  put_pattern#
            #
            mov   ah,stop_count ; STOPキーが押されていたら終了#
            and   ah,ah#
            jnz   _004#
#
#      jmp   _002#
#
#_001:      call  scroll_out#
#
            mov   dx,org_str_ptr ; 文字列の先頭にポインタを戻す#
            jmp   _003#
#
#_004:      pop_all#
            ret#
            endp#
put_string
:      -----#
:      画面に残っている文字をスクロールアウトする#
:      #
```

で変換する。これはプログラム全部を一気にまとめて、変換するタイプのものだ。おもに `C` 言語や `アセンブラ` など。逆に1句1句変換しながら実行するタイプは、通訳者の意味の `Interpreter` (インタプリタ) と呼ばれる。`Basic` がその代表例だが、`Visual Basic` などではコンパイルしても実行できる。なぜ `Translator` (翻訳者) じゃなく、`Compiler` なのかは定かでない。

J A V A (ジャバ)
サンマイクロシステムズという会社を作った、インタプリタ方式のプログラム言語。マックだろうと `UNIX` だろうと何でも動くということで、

HTMLやVBA (Visual Basic For Application) などと呼ばれる言語があり、プログラマと名乗る人は、必ずなにかしらの言語をマスターしているのだ。

コンピュータの知能は小学生レベル

3Dグラフィックがグリグリ動くゲームや高度な科学技術計算。さぞかしコンピュータは、素晴らしい計算ができるに違いないと思うだろう。ところがどっこい、コンピュータのCPUは泣きたくなるぐらいバカだ。

CPUができることといえば、四則演算(+-×÷)に、メモリの読み書き、値の大小の判断ぐらいしかない。ホントにこれだけなのだ。フロップの読み書きもできなければ、インターネットにつながることすらできない。ウィンドウやマウスポインタを画面に表示するなんて、高度な技はCPUは一切持っていない。乗除(×÷)も怪しいモンで、CPUの中では足し算と引き算の繰り返しをしているに過ぎないのだ。

まさにCPUの知能は、小学生レベルだ。いや、CPUは小数点の計算がとっても苦手なので、小学校低学年ぐらいかもしれない。これで3Dのゲームができ

たり、ムービーが見られるんだから驚きなのだ。

こんな貧弱な知能なのに、なぜ高度な計算ができるかといえば、計算速度がべらぼうに速いためだ。先に紹介したとおり、パソコンは1秒間に14億回の計算ができるから、多少効率の悪い方法でも速さにまかせてゴリ押ししている。

たとえば、 $125 \div 3$ を考えてみよう。何秒で答えが出せるか？ CPUは、この計算をするとき125から3ずつ引いて、何回引いて最後にいくつ余ったかをはじき出す。答えは41とあまりが2。わざわざ41回引き算したって、41/41億秒しかかからない。あなたが計算するより、ずっと速い。

これは極端な例だが、プログラマはこの小学生程度の知能を相手にしなくてはならないのだ。とはいえ、プログラム言語のおかげで「125から3ずつ引いて、引けなくなったら…」と教えなくても、 $125 / 3$ と書けばいい。

コンピュータがスペースシャトルを飛ばしたり、高度な科学計算ができるのはひとえに、プログラマの先人たちが開発し、進化したプログラム言語と絶え間ないプログラマの不健康と不摂生の賜物なのだ。

VBA

エクセルやワードで繰り返し行っていることをプログラムしちゃって自動化しようというもの。Visual Basicというプログラム言語に似ているので、分かりやすいとあちこちのメディアで言っている。でも、実際プログラムしてみると、マイクロソフトが作ったえらく難しい概念に阻まれて、本家のVisual Basicより難しいじゃん！ マイクロソフト以外では「マイクロ」と呼ばれている。また、マイクロソフトだけ暴走かよ。

大小の判断

値の大小の判断も基本的な引き算で行う。AからBを引いて0になったら

II(イコール)。AからBを引いたら、引ききれなくて上の位から1借りてこなければならぬ(桁あふれ)ときは、 $A \wedge B$ (AはBより小さい)。AからBを引いても桁あふれもしないし、ゼロでもない場合は $A \vee B$ (AはBより大きい)として判断している。まるで小学生…。

1秒間に14億回の計算

これは Pentium 4 2GHz程度の場合。実際には、メモリやハードディスクの読み書きがあるので、そんなに計算できない。実際にはFLOPS(フロップス)という単位で表され、14億回の場合140MFLOPSと表記される。

本を読むのが仕事

コンロや包丁、オーブンの使い方を知っているコックさんがいても、料理の味やその手順（レシピ）を知らないのと、料理は作れない。プログラマもまったく同じで、いくらプログラム言語に精通していても、麻雀のルールを知らなければ、麻雀ゲームは作れない。だから、とにかくたくさんの本を読む。

医者もパイロットも、それは分厚い専門書を読んでいるが、それ以上に読む必要がある。麻雀ゲームを作るなら麻雀の本、經理のシステムを作るなら經理から簿記の本、電子ジャーのプログラムなら「新米奥さん！ おいしいお米の炊き方」なる本まで読むのだ。

この程度の本ならまだいいが、レアな技術のプログラムを作るときは大変だ。防衛庁から弾道計算のプログラムを依頼されたら、弾道計算に関わる本が必要だ。「つて、そんな本どこで売ってるんだ！」と、叫びたくなるような本が必要になるときが、往々にしてある。ホントに。こうなると、まずはインターネットで本にあたりをつけて、東京は神田まで赴き、本屋の軒を何度もくぐるハメになる。

ときには、依頼元から「これ資料ですから」とコクヨの10cmバインダをダンボ

電子ジャーのプログラム

たかが電子ジャーとタカをくくっていられない。内蔵されているマイコンのほとんどは、16ビットCPUだ。10年前のパソコンと同程度の賢さ。一時期、日本国内はPC-9801シリーズというパソコンで埋め尽くされていたが、アレが電子ジャーの中に入っていると思えばいい。

プログラムも並大抵のものではなく、それこそ呪文のようなアセンブラで作られている。これに加えて、各社が実験を重ねてはじき出した、各時間帯での温度や加熱時間はまたまた新米やおかゆモードなど、機能は盛りだくさん。プログラムを印刷したら、10cmバインダ

ールいっぱい渡されることもある。これはもう拷問。とりあえず、主語と述語らしきものは分かるが、用語の意味がサッパリ。こんなときは、用語辞書をやって神田まで買いに行く。しかも、辞書に乗ってない数文字のアルファベットまであって、依頼元に質問したら社内だけのスラングだったり..。

加えてコンピュータは、マイクrosoftが変な新しい概念を打ち出してきたり、プログラム言語が改良されたりで、こちらの専門書まで目を通さなくてはやっていけない。この世で一番本を読み、勉強しているのはプログラマなのだ。

本を読むことに慣れてしまうと、こんどはクセで読んでしまう。コンピュータの周辺装置に付いてくるマニュアルはもちろん、仕事とは関係ないビデオデッキやオーディオ機器のマニュアルなど、一字一句漏らさず読む。すごい小さな文字で印刷されている「もっと便利な使い方」という小さな記事でも読む。最後の仕様一覧もとりあえず目を通して、消費電力や重量にうなづく。

とにかくプログラマは、装置やシステムに知らないことがあつてはならない！という恐怖心だか、強迫観念に駆られたように読み漁るのだ。もし、知り合いにプログラマがいるならば、その人と同じAV機器や装置をそろえろといひ。マニ

1冊は軽く越えるだろう。いまやビデオのリモコンだって16ビットCPUが使われているのだ。

コクヨの10cmバインダ

プログラマの必須アイテム。キングジム製を好んで使う人も多い。最大15cmバインダもあるが、そこまでいくとめくりにくく、留め金が多く曲がってしまう。だいたい1cmあたり100ページぐらいなので、10cmものだと1000ページ。開いてみたら両面印刷されて「200ページもあるのかよー」というオチがしばしば。

世界で一番本を読む

本を作った筆者や編集者以上に読み込んでいる。ワシはコンピュータの技

第1章
プログラマの呪文

- 「フタを半分まで開けてお湯（熱湯）を注ぎ、3分待つてよくかきまぜてからお召し上がりください。」
- さてこれをプログラムにすると、こんなに面倒になる。
- ① フタを5cm開く。
 - ② ポットのお湯は何℃かを確認する。
 - ③ もし90℃なら熱湯とみなし⑩へジャンプする。
 - ④ やかんに水を400cc入れる。
 - ⑤ 右手前のコンロにやかんを置く。
 - ⑥ コンロを着火する。
 - ⑦ 水温が90℃になるまで待つ。
 - ⑧ コンロの火を消す。
 - ⑨ やかんのお湯をポットに入れ②に戻る。
 - ⑩ ポットから350ccのお湯を注ぐ。
 - ⑪ フタを閉める
 - ⑫ 3分待つ。

ユアルに小さく書いてあった、ウラワザ的な使い方を教えてくれるだろう。逆にいえば、本を読むのが大好きで、技術系のNHKスペシャル万歳！ あまつさえディスカバリーチャンネル最高だせ！ という、好奇心旺盛で「世界にオレの知らないことはあつてはならん！」という人は、プログラマとしての素質をじゅうぶんに備えていると思っていいたいだろう。

どこまで分解できるかが勝負

これから作るシステムに関しての本や10cmバインダとの格闘が終わったら、これをプログラムとして書く。ところが、これらはあくまでも人間様が理解できるように書いてあるだけで、小学生なみの知能のコンピュータに手順として教えるのは、かなり噛み砕かなければならない。

割り算を引き算まで噛み砕くように、どんなに複雑な計算式や手順でもコンピュータがわかるようにプログラムするのだ。たとえば、カップラーメンを作るプログラムを作るとしよう。

人間様向けの作り方はこうだ。

術系出版社でプログラム向けの本を編集していたので、それをよく知っている。なぜなら、翻訳書などで自分が理解できなかった箇所は、原文をそのままを直訳して出版してしまう。すると後日、必ずユーザースポート電話や質問ハガキが届き「この意味がわからないぞ！」とお叱りを受けるのだ。でも「こつちも分からネーから原文直訳なんだ！ ゴルア！」とも言えず「サポート範囲外です」と、丁寧にお断りしてしまうのだ。プログラマの皆さんごめんなさい。とくに凶器になりそうな分厚い翻訳書の後ろの方になると、高度な内容に反して体力と納期の問題でだんだん手薄になってくる。

ディスカバリーチャンネル
ハイテクトイズや驚異の機械、スーパー・ストラクチャーやパワーゾーンなどを好んで見るようだ。モノを作るといふ点でチャレンジのジャンクヤードウォーズも仲間内で大好評。新番組のフルメタルチャレンジとサンダーレースもイイ感じ。

カップラーメン

お分かりと思うが、日清のカップヌードルだ。かやくやスープが小袋に入っていたら、さらに面倒な手続きが必要。というか、これをコンピュータに判断させるのは難しい。

第1章
プログラマの呪文

- ⑬フタをすべて取り去る。
⑭カップを固定し右手にハシを持つ。
⑮ハシをカップのフチから10cm差し込む。
⑯ハシを毎分60回転で5秒間回す。

そう。人間では常識の「お湯がなかったら沸かす」ということを手順に加えたり、熱湯とは何℃以上のお湯を示すかを教え、かきまぜるのに不都合なフタをあらかじめ取ることも教える。もちろん、「かきまぜる」とはどういう動きかも正確にプログラムにする。

ここまで懇切丁寧なプログラムでも、実際にロボットにカップラーメンを作らせたら問題だらけだ。フタは何kgで引っ張ればいいか？ 水の出し方は？ 左手前コンロは、何10cm前か左か上か？ ポットにいった後のやかんは、どこに置くのか？ などなどだ。

今度はカップめんを10個作らせてみよう。先の手順を10回繰り返しさえいいだけだ。人間ならなんのこともない。ところが、コンピュータで先の手順どおりに作らせると、最後のラーメンができるまでに30分以上かかる。そう、律儀に①～⑫

を10回繰り返し返すと、3分待つては次を作り、また3分待つて3個目……。最後のラーメンができた頃には、最初のラーメンはカップからはみ出して悲惨な結末を迎えることになる。

このように、ありとあらゆるシーンを想定し、それぞれに対応する手順をこと細かに記述する。これがプログラムなのだ。もし想定しないシーンがあれば、それはテレビや新聞などで耳にする「システム障害」（通称、バグ）となる。記憶に新しい、みずほ銀行のATMのシステム障害は、想定した以上に中継コンピュータにデータが殺到したことが原因だ。

前記の場合も、コンロの電池切れで着火できないことは想定されていない。たまたま電池が切れていたら、お湯が90℃になるまでコンピュータは待ち続ける。いわゆる「システムダウン」状態なのだ。航空管制のコンピュータがシステムダウンしたのは、想定されない自衛隊からの誤った運行データが、民間航空の管制コンピュータになだれ込んできたためだ。みんな「想定されていない」不測の事態で、コンピュータは止まる。

プログラマは、人間向けに書かれたシステムのルールや仕組みの行間を読み、

カップラーメン10個
カップラーメン10個であれば、一気にお湯を注いでそれぞれ3分まてばいい。ところが100や1000個になると、お湯を注いでいるそばから3分たってしまうので、さらに難しいプログラムが必要になる。原理としては、シューティングゲームのようには、あちこちでいろいろな操作が発生し、たくさん敵キャラを動かすのと同様のプログラムが必要だ。すべてのラーメンを常に監視し、なんらかのアクションがあった

らそれを優先して処理するというイベントドリブン（イベント駆動）方式のプログラムが必要になる。

バグ

プログラムのミスのこと。英語では虫（Bug）の意味。日本語はなぜかラ行変格活用し、バグらない、バグりました、バグる、バグれば、バグろう、となる。また形容詞としても使われ「なんじゃこのバギーなプログラムは！」といったように使われる。修正作業は、虫にちなんで「虫取り」するなどとも言いが、現場はかなり切羽詰った状態なので、緊迫感の高い「デバッグ」が好んで使われる。

そこで問題が発生した場合はどうするか？ ということをひとつひとつ想定しながら、プログラムを書き連ねていくのだ。

こんな事情もあって、できるプログラマの作ったプログラムは、細かくモジュール分けされていて、ひとつのモジュール（プログラム）が小さい。最大でも一画面に表示しきれられる程度だ。こうすることで、ひとつのプログラムを作るときに考えることを極力減らしているのだ。何画面にも渡る長いプログラムは、それだけ考えることも多く、うっかりミスが発生しやすい。それ以上に、短いプログラムならキリが着きやすく、うっかり終電を逃すなんてことも少なくなる。まさに一石二鳥なのだ。

それ以外にも気を使うことがある。たとえば、データベースを検索して何件見つかったかを表示するプログラム。「データは必ず見つかるもの」という前提でプログラムを書く。「0件見つかりました」となる。そりゃあんた「見つかりませんでした」でしょ。数学的に合っているからOKとしているのか、はたまた面倒からなのか？ この手のメッセージは非常に多い。気の利いたプログラマなら、0件だった場合だけ「見つかりませんでした」というメッセージを表示するよう

にする。こういう細かい気配りで「このソフトが使い易いだの」「難しい」だの言われてしまうので、神経も使う職業なのだ。

物理や数学・英語が得意なのか？

プログラマであることを教えると、必ず言われるのが「頭いいんですね」やら「英語が得意なんですね」とか「じゃあ数学か物理を専攻されたんですか？」などだ。これは「日本人がちょんまげ結ってる」とか「中国人はみんな麻雀できる」と同じぐらいの誤解。

確かに、英語がべらべらで外資系のメーカーでプログラマやってるヤツもいるし、応用数学を専攻していて「MP3ぐらいなら簡単に作れますよ。へっへっへ」という大学生バイトプログラマもいる。でも、ほとんどが普通の人たち。ただ行動や言動、社会に対する適応性にかけた人が多いのも事実。

学歴は、一切関係なし。プログラムを作るとなると本や資料を読まざるを得なくなるので、高校や大学を卒業していなくても、専門家にも勝るとも劣らない知恵がつく。コミケで一緒にバカやっていた連中が、証券取引場のシステム（電光掲

モジュール
ひとつのプログラムをいくつかの機能に分解したもの。プログラムをパッチ化して、最後にそれをまとめるプログラムを書くことで、どんなに複雑で手間のかかる処理でも分解できる。たとえば、ユーザーからの入力を受け、計算処理、画面への出力を3モジュールにして、最後にこれをもとめるプログラムを作れば、一連の動作になる。モジュール化すると考えがまとまるのだが、いざくつつけてみたら「ありや、かみ合わない」というデメリットもある。これについては、別の章で。

ほとんどが普通の人
別の章で明らかにするが「仕事は普通に見える人」といった方が正しい。なにせプログラマは、頑固オヤジの江戸前ずし職人じゃないが、それにかなり近いものがある。コレに加えて、みんな大きな夢を（ガンダム作りたいとか、光子魚雷ってどう作るんだろうなあみたいなど）持ったりして、付き合い方には独特のテクニクが必要。

示板に出るアレ)を開発して、妙に株式に詳しくなって個人投資を始めたり、巨大企業の基幹系システムとインターネットサービスを連動させて大型コンピュータにえらく詳しくかつたりとさまざま。

仕事によっては、インターネットキャバクラ(女の子の映像が表示されてチャットできる)システムを作って、風俗にえらく詳しくなったヤツなどいろいろだ。こいつらは、ワシも含めてみなキャンパスライフとやらで、女子大生たちと酒池肉林の宴を持ったことがない。

反面、国立大学の院卒で「頭のいい」プログラマは、アイデアに貧困だったり、あらゆるシーンを想定できない。カチカチ頭で、すぐ何でも常識として考えてしまうので、先に書いたラーメンの例ではないが、ある条件の上に立つことが前提のプログラムになってしまいがちだ。ユーザーが予定外の値を入力したり、他のプログラムから範囲外の値を受けたりすると、途端にシステムダウンするプログラムをよく作ってくれる。ありがとよ。

最も強烈だったのは、これも国立大の院卒プログラマ。社内の女の子に一目ぼれしたらしく、ストーカーになってしまった。挙句の果ては、レイプ事件を起す

始末で速攻クビ。仕事以前に犯罪はいかんよ。しかも強姦は…。世の中には、高学歴プログラマで、国の研究施設が何かで働いているスーパープログラマもいるんだろが、身の回りには見かけない。

先にちよつとだけ触れたが、コンピュータは小数点の計算がとつても苦手である。これをカバーするべく、最近のCPUには「浮動小数点演算ユニット」なるものが組み込まれている。いわばエンジンのターボやスーパーチャージャーといったところ。しかし、地球の周回軌道計算でもするわけでなし、小数点以下はせいぜい3桁まで。これなら最初から1000倍して計算して、表示するときに小数点をつければいいだけだ。プログラマは、数学ができなくてもちよつとしたアイデアが思いつけばいいのだ。

外資系企業に勤めたり最先端を扱うのであれば、英語もそんなに必要ない。さすがに先端技術になると、英語のマニュアルや論文まがいの本を読まざるを得ない。ただ普段プログラムで使う英語は、ほとんど中学生レベルでOK。中には難しいスペルの単語もあるが、ヘルプからコピー&ペーストしてくりゃいいのだ。プログラムで使うある値を記憶するためにメモリに名前をつけることがあるが、

インターネットキャバクラ

何分接続してナンボという課金なので、女の子はなかなか脱がないらしい。なぜ、ボイスチャットじゃなくて文字だけのチャットなのかと聞いたら、恥ずかしがり屋さんが多いからだという。とくに日本人のパソコンユーザーは、ボイスチャットだと黙っちゃって、会話にならないらしい。テキストのチャットの方が、より発言して長時間接続してくれんだと。それにボイスチャットでガンガン会話する人は、本物のキャバクラ行くでしょだった。うひゃー、よく研究しているなあ。心理学と民俗学に統計学を研究した上での、インターネットキャバクラなのだ。素晴らしい。

身の回りには見かけない

確かにいるはず。純国産ロケットが打ち上げられるんだから。MPEGなんかの難しい理論を実際にプログラム化する人とかいるわけだし。単純に類は友を呼んでいるだけなかも知れない。でも、米国のマイクロソフトで働くプログラマも、回りにいる連中と同じような人間だった…。つっかか、会社にサーフボード持って来てるは、犬連れてるは、プログラムにつまるとボーイングフィールド(工場)まで行って、気晴らしにセスナ飛ばして、いろいろな強烈な人種だ。やっぱり、毎日わらじみたいいなステーキ食ってる民族は豪快だわ。

これは無理に英語で書かなくても大丈夫。だからプログラムをよーくみると、DB_SaidaiKensuやMisuketa、Kazuなどとローマ字で書いてあるものがたくさんある。少女漫画の背景によくある、ローマ字で書かれた作者からの隠しメッセージみたい。

たまーに英語が必要になるのが、ヘルプを読んでいてジャンプした先が英語だった場合だ。マイクロソフトのヘルプには、よくあることだ。こんなときは、翻訳ソフトを使えばいい。技術書なので文学的な言い回しもなく、翻訳ソフトも素直に訳文を表示してくれる。たまに、訳してはいけない用語を無理に訳してくれて、爆笑川柳が出てくるときもあるが、カタカナと英語を照らし合わせれば、サッパリ意味がわからないということはずまない。

物理にいたっては、普通のプログラムを作っているぶんには、まず必要ない。ロケットを打ち上げる人や、ニュートリノだのドップラー効果で星間の距離を計算しなきゃならんのであれば必須だろうけど。でもそんなプログラムなんて、まず外注して作るものではなく、学者さんたちが自分たちでシコシコ作っているので、慌ててニュートン力学やらをおさらいすることもないだろう。

プログラマは開発のたびに新しい分野を勉強する必要がある。好奇心旺盛な人なら学歴なんかなくて誰でもなれる職業なのだ。

徹夜が多いのは本当か？

これまでに紹介したとおり、プログラマはありとあらゆるシーンを想定する。さらに計算式や仕様書の行間を読んで、コンピュータに分かるように手順をプログラムとして書いていく。「あゝ、こんな場合も考えられるなあ」なんて思ったら大変、それに付随してネズミ算式に「あんなシーン」「こんなシーン」があれよあれよと思いついて、さらにそれに対応するプログラムを書いていかなきゃならない。

考えることが倍々に増えていくのだ。ムーアの法則がここにも適応されるとは…。大事なことはメモしておくのだが、メモしている間にも、いろいろ考えることが出てくる。こんなときは、メモするより、頭にあることをすべてプログラムに書いてしまうのだ。だから、ものすごい集中力。「今日は頭の中にあることまでをプログラムせねば、明日になったら忘れてしまうっ！」とばかりに、キー

先端技術

だいたいマイクロソフトがぶち上げて、コンデモない構想、いや妄想だったりする。でもOSを握られているから、アプリケーションソフトはそれに従わざるを得ない。だから勉強する。マイクロソフトもよく知っているもんで、会社の経営陣にコラボレーションプログラムなどをアピールしてくるもんだから、トッパウンで最新技術を使わされることになる。「別にそんなの、今の技術やソフトでできるじゃん！」といういいわけは許されない。革新的で新しいが、バギーで複雑怪奇なものを使わねばならない苦悩もあるのだ。

ジャンプした先が英語

ユーザーがよく参照しそうなところは日本語化されている。ところが、ポイント的なトピックにジャンプした瞬間や、難しくてもわらにもすがる思いでヘルプを調べると、めぐるめく英語ワールドに…。最近OSのヘルプも以前のバージョンを流用してたりして、XPのヘルプなのに「ウインドウズ9500では云々…」と。それに従って操作していると「そんなボタンはねーぞ！」ってなタイプログボックスに直面する。お願いだから、ウインドウズ9500のヘルプそのままコピー&ペーストしないでくれ。

ボードを叩きまくる。

そして誰もが最初に戸惑う独り言だ。「これがこーなると、あーなって」「くそっ!」「おお!」そーゆー手があるか!」「ゲイツ殺す!」「なぜだっ!」などの小言が多い。非常に多い。開発現場を見た新卒の女の子なら、とてつもなく恐ろしい光景だろう。声だけ聞いてると、機動戦士ガンダムのアフレコでもやってんじゃないか? というほど魂の入ったセリフだ。でも彼らは、独り言を言っていることさえ気づいていない。それだけ集中しているのだ。また、自然に声に出して、プログラムの動作などが正しいか? 考えはあっているかを耳から確認しているのだ。だから、会社も独り言禁止にはできないのだ。

そこまで集中していると何が起るかは明白。朝になっちゃうのだ。ほとんどの場合、頭にあることをプログラムしているうちに気づいたら朝になっていた。あるいは「ありゃー終電逃した〜!」というケースがほとんどだ。中には「えー、また仕様変更ですかあ」とイヤイヤ徹夜せざるを得ないときもあるけど…。

こうした生活が続くと、日本にいながら時差が出てくる。朝の5時まで仕事して自宅に帰ると朝7時。ここから寝て昼すぎに起きて、夕方に出社すると、今度

はまた翌朝までという勤務体系になる。「寝なけりゃいいじゃないか」という意見もあるが、意識が朦朧としていたら「あらゆるシーン」なんて考えられない。結局、徹夜で書いたプログラムは、ミスだらけで寝た以上の時間を修正に費やすことになる。だから寝る。そして、その時差ほぼ8時間。ハワイの時差は、とつてもきついというが、まさにその時差と同じだ。日本時間と正反対の生活なのだ。一度こうなると通勤ラッシュに巻き込まれないので、かえって楽チン。若い連中は面倒くさいから自宅に帰らず、会社の床に寝袋を広げたりして、オフィスがキャンプ場になる。中にはシャンプーやら洗面器、ガスコンロ（セブンイレブンでナベを買って作っているらしい…）までデスクにあって「お前、住民票会社に移した方がいいんじゃない?」というツワモノもいるのは確かだ。

ただ大企業になると会議やら打ち合わせが多く（しかも、狙ったように午前中）、無理やり起きていると「ボンヤリしてるだの」「何も意見をいわない」などの小言を上司にいわれるハメになる。そりゃ寝てないからっすよ。勤怠表をしつかり見てくれよ上司。

「プログラマは夜が遅い」というのは、タクシートの運転手さんが一番分かってい

ムーアの法則

1965年、インテルの名誉会長 Gordon Moore が予測した「LSI（チップ）のトランジスタ数は18ヶ月で2倍になる」からきている。かなり拡大解釈されて「コンピュータは倍々に進化する」という意味合いで使われる。一時期はやった、ムーアの法則にも数多くうなづいたが、ムーアもそれ以上だ。パソコンのメモリがいい例、64M、128M、256Mがお手ごろ価格の今、次は512Mが2000円の時代はすぐソコ。ハードディスクも、2000年になる前は40Gバイトが主流だったが、今じゃ80Gバイトが主流。

キーボードを叩きまくる
こういうときは一切口を利かないのがお約束。どんなに急ぎの件でも、必ずメールすること。さもないと「バカヤロウ」呼ばわりされて、納期が遅れた恰好の原因に仕立てられるのだ。普段ウオークマンを聞いて仕事しない人が、突然イヤフォンをしたときも、かなり危険な状態。タバコの本数、瞬きの回数などを見て、危険度を見極める。これが大切。

なぜだっ!

ガンダムであまりにも有名なセリフ。ギレンのジオン国民に向けた長い演説にある力こもった「なぜだっ!」。まさに、あの絶叫が聞こえる。

るかも知れない。終電が終わった時刻になると、会社の前には数台のタクシール路肩に止まっている。流しのタクシールを拾うと、開口一番「お客さん、コンピュータ関係のヒト？」なんてよく言われるものだ。運転手さんいわく「コンピュータ、出版、放送」関連は、みなさんの多くがハワイ時間で暮らしているらしい。

情報処理試験なんてクソ喰らえ！

世の中こんなご時世で、資格試験を受験して少しでも就職率を稼いだり、高給取りを目指している。コンピュータの分野も、ご多分に漏れず試験がある。一番メジャーなのが、情報処理試験だ。

ところがこの資格、そーとークセモノなのだ。国家試験なので試験作ってるのはお役人。だから、いま旬の技術に関する問題なんてありやしない。試験問題は、一昔前の技術が多く、試験問題を見ると「こんないい方、いまだきしねーよ」的なところがあるのだ。とくにパソコンの技術の進歩は早いので、ものすごく役に立たない試験に見えてしまう。ただ、大型コンピュータの場合は、受験してもソソはない。

溶接技師や危険物取り扱い技師、大型2種や中小企業診断士のように、持つてりや飯食える資格には程遠い。それが今のところの情報処理試験の権威だ。下手をすると、マイクロソフトやオラクルが片手間にやっている資格の方が、国家試験より権威があるという…。

少なくともパソコン中心の開発をしている会社の面接では、たとえ情報処理試験に合格していようと、自慢したらダメだ。あくまでも、履歴書にこじんまりと書き、それを聞かれたら「とりあえず受験しました」と謙虚に行こう。

ましてや、初級シニアド試験は、プログラマたちに「アフォの姉ちゃんが取る資格」とまで言われる始末なので、男のあなたなら受験したことさえ内緒だ。バレたら「イヤ、ナンパ目的っすよ」と答えよう。間違っても本当の「匂いをかぎに行った」が禁句。

大型コンピュータ中心の会社や、社内で情報処理試験を取得する風潮がある会社なら、もつとアピールしてもいいだろう。資格によっては、手当てを支給してくれる場合がある。まあ数千円程度〜1万円前後が相場らしい。ただ1年間続くと結構な額になるので、ぜひ受験してみよう。

「坊やだからさっ」と合
いの手を打ってあげよう。

仕様変更

詳しくはあとの章で述べるが、プログラムの設計図ともいえるのがこの仕様書。どういうデータを入れて、どういう計算をして、どのような値を表示すればよいか？ そのシステムは、何人で使うのか？ など、さまざまな情報が書いてある。仕様書を書くのはSEの役割なのだが、下手をすると仕様が甘くて、苦勞させられる。

会社の床に寝袋

これは万国共通らしい。家のなかでも土足のアメリカ人プログラマでも、床に寝袋を広げていた。

すごいヤツだと、自分のパーティションの中にベッド（サマーベッドじゃない本物）を持ち込んである輩も。おそるべしアメリカ人。

情報処理試験

経済産業省が音頭を取って年に2回開催しているプログラマ向けの試験からシステムを監査する技術者向け試験、データベースエンジニア向けの試験、システム管理のシニアド試験まで、細分化されている。お薦めは、初級シニアド試験。若くて可愛い女の子たちがたくさん受験しにくるので、気分は女子高の教師。試験中は、甘い香水のおいで集中できないことウケアイだ！ 唯一権威が

ゲームソフト開発をメインにやっているような会社では、資格をもっていることすら伏せたほうがいい。履歴書にも書かない。面接で聞かれたら、しようがないから答えるしかない。もし入社して資格を持っていると手当てがもらえるようであれば（ほとんど皆無だ）、あとから人事にこっそり耳打ちしてもいいだろう。

どうにもゲームプログラマは、情報処理試験を受験することすら恥なのだ。小学校でウンコするのと同じぐらい恥ずかしいらしい。大型コンピュータは未知の分野なので、自分が受かりそうもないから、あえてバカにするのかは定かでない。とにかく、彼らはすごいゲームが作れてなんぼの世界なので、極秘事項扱いだ。

あるのは、システム監査技術者試験。こいつは、難関なので持っているだけでも結構ハクが着く。

面接

下手な情報処理の資格よりは、自分で作ったプログラマなどを持ち込んで、こんな凄いの作りまいたと売り込んだほうがいい。かくゆうワシも、アセンブラとBasicで書いたプログラムを全部打ち出して、持っていったら現場の人がしげしげと見ていた。プログラマ同士ってのは、言葉話さなくても相手の書いたプログラマを見ると大体の性格やスキルが分かるものなのだ。まさに以心伝心。