

D <sup>♥ H25</sup> 言語

群馬高等専門学校

Gunma National College of Technology Computation club



# 付録 CD について

CD には以下のファイルが含まれています。

- 部誌 PDF 版
- プログラム作品
- 使用した画像等
- 2012 年以前の配布物

収録されているプログラムは Windows vista Business SP2 32bitd で動作確認を行っております。なお、全てのコンピュータで動作を保証するものではありません。また、各プログラム、画像の著作権はそれぞれの製作者にあります。バグの報告、感想等ありましたら下記のアドレスにお願いいたします。

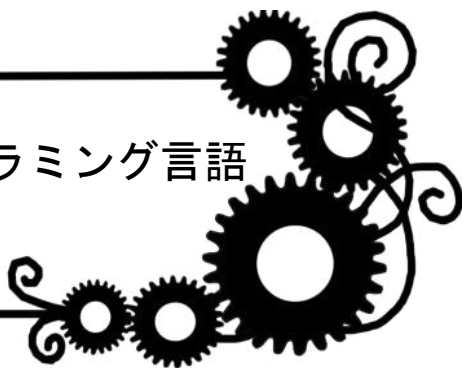
[gnct.densan@gmail.com](mailto:gnct.densan@gmail.com)



# 目次

1	付録 CD について	1
2	サウンドプログラミング言語の紹介	3
3	東方銀符律ノ手引き	7
4	マイコン・マメコン・オワコン	14
5	ArchLinux のアレ	17
6	systemd で Gentoo を使う	27
7	ソーシャルゲームについて	31
8	プチコンで靈異伝を作ってみる	35
9	日本人の日本人による日本人のためのプログラミング	39
10	自作 PC を組んでみた	46
11	ミクさん召喚計画	50
12	部員紹介	53

# サウンドプログラミング言語 の紹介



著者 NaGa

## 1 はじめに

こんにち。NaGa  
(なーが) と申します。

突然ですが皆さん、サウンドプログラミングをご存知でしょうか。私も最近知って興味が湧いたので少し調べてみたのですが、汎用のプログラミング言語でできるみたいですね。当然だね。しかし、サウンドプログラミングのために開発された専用のプログラミング言語もあるということなので、本記事ではそれを紹介したいと思います。あくまでも「こんなプログラミング言語もあるんだよー」という紹介に過ぎませんので、ご了承ください（私自身ほと

んど理解していないというのは内緒）。少しでもサウンドプログラミングに興味を持っていただけたら幸いです。

## 2 サウンドプログラミングとは

先程から「サウンドプログラミング」という単語が並んでいますが、「それって何？」と思う方もいるのではないのでしょうか。簡単に説明すると、sound（音、音響）programming（プログラム作製）、つまり音響処理に関するプログラムを作ることです！  
……そのままだの意味だな？言われなくてもわかる？すみません、私もこ

れくらいしか説明できません…。サウンドプログラミングでは音を作る、加工する、鳴らすなど様々な処理を行うことができるのですが、それをするためには音に関する理論を勉強する必要があります。大変かもしれませんが、サウンドプログラミングをやってみたいという方、一緒に頑張りましょう。

### 3 専用言語

第1節で述べたとおり、サウンドプログラミングにはそれ専用のプログラミング言語が存在します。様々なものが提案されているみたいですが、今回は2種類紹介します。

#### 3.1 Pure Data

まず1つめは、Pure Data（ピュアデータ）です。可愛らしい名前です。Pure Dataは命令や数値などをそれぞれ1つのブロックにして、そのブロックをつないで処理を書いていくプログラミング言語です。視覚的に見やすいという印象を受けまし

た。普段プログラミングをしないという方はPure Dataから始めてみるのもいいかもしれません。

#### 3.2 SuperCollider

2つめは、SuperCollider（スーパーコライダー）です。強そうな名前です。SuperColliderはテキストによって処理を書いていくプログラミング言語です。普段からプログラミングをする方にとっては馴染み深いと思います。文法は汎用のプログラミング言語と大差はありませんが、少し難しそうです。

### 4 おわりに

本記事を急遽書くことになり、知識不足な部分しかないと思いますが許してください。これから勉強します。Pure DataやSuperColliderについて詳しく知りたい方は、各自調べて下さい。と言っても、日本語でサポートしているところは少ないですけどね。もっと増えてほしいですね。以上、サウンドプ

プログラミング専用言語の紹介でした。

## 5 おまけ

以下、音楽ゲームをやる人向けのお話。また、おまけが本編。

サウンドプログラミングでは音に関する様々な処理を行うことができます。エフェクターとかも作製できるみたいですね。エフェクター・エフェクトといえば、何を想像するでしょうか。

……某音楽ゲームたちを想像したあなた。私と同類です。やったね。お皿やつまみ、ギターなどがあると思いますが（これで伝わるよね？）、ここではお皿（以下「忒寺」）のお話をします。

忒寺をやっている方、エフェクター機能を利用しているでしょうか。筐体の effect ボタン (effect on/off) を押すと、エフェクターが ON になります。その状態で VEFX ボタン (VEFX change) を押すと、エフェクトモードが切り替わる仕様になって

います。モードは 3 種類あり、VEFX モード、EQ ONLY モード、PITCH モードです。

### 5.1 VEFX モード

VEFX モードでは、VEFX スライダー (VEFX) を上下させることで「ECHO 1~4」「REVERB 1~3」の 7 種類のエフェクトをかけることができます。echo は「反響」という意味、reverb は「残響」という意味です。音を響かせたいときに VEFX モードを利用するといいでしょう。

### 5.2 EQ ONLY モード

EQ ONLY モードでは、エフェクター効果をなくしてイコライザとフィルタのみの効果を出力することができます。VEFX スライダーによるエフェクト効果をカットし、音質のみを調節することができます。low-EQ スライダー (low-EQ) を上下させることで低音出力の増減量を、high-EQ スライダー (high-EQ) を上下させることで高音出力の増減量を変化

させることができます。また、filter スライダー (filter) によってローカット (低い音をカットする) フィルタのカット量を調節することができます。つまり、いずれのスライダーも中心で変化なしです。

### 5.3 PITCH モード

PITCH モードでは、VEFX スライダーを上下させることでピッチを変更することができます。pitch には「音高」という意味があります。つまり、音の高低を変化させることができます。スライダーを下げると音が低くなり、スライダーを上げると音が高くなります。中心で変

化なしです。

### 5.4 まとめ

簡単な説明になりましたが、各自いろいろと試してみてください。そして、自分好みのエフェクトを発見してみてください。1 クレジットを有効に使うか使わないかは、あなた次第です。

### 参考文献

- [1] 青木直史、サウンドプログラミング入門 – 音響合成の基本と C 言語による実装、pp.273-274、2013 年 3 月 10 日、技術評論社

# 東方銀符律ノ手引き

著者 幻馬

皆さんこんにちは！幻馬と申すものです。さて、今日は「東方銀符律」というカードゲームについて書いていこうと思います。

まず、東方について語っていこうと思ったのですが、尺が足りないので全く東方を知らない人は私のブログを読んでください。  
<http://blog.livedoor.jp/galvashighwind/>(決してステマでない)

## 1 東方銀符律とは



東方銀符律とは、「東方



project」を題材にした対戦型カードゲームです。しかし、カードゲームと言っても色々な種類がありますよね遊戯○やデュエルマ○ターズ、○アイスシュヴァルツに cha ○ s。この東方銀符律は「Lycce」に似ています。え？ Lycce を知らない？大丈夫だ、問題ない。一から説明していきます。なんせこの東方銀符律の全国競技人口は1万2千人くらいしかいませんので……(ちなみに遊戯王は全世界500万人くらい)

## 2 ゲームの説明

まず大雑把なルールから。お互いに60枚のデッキを用意し、そのデッキを先に削り切ったら勝ちです。ね？簡単でしょ？

次に、ゲームの進行について。お互い自分のデッキをシャッフルします。そして、デッキの上からカードを7枚ドローします。じゃんけん等で先攻後攻を決めて、ゲームスタートです。

まずカードを『2枚』ド

ローします。(先攻最初のみ1枚)そしてメインフェイズを行いエンドフェイズを挟んで相手のターンです。これを繰り返します。各フェイズに行くことは後に説明します。

っと、まだカードについて説明していませんでした。カードの例として、可愛い諏訪子ちゃんを用意しました。



いやあ可愛いっすね〜で、このカードの見方ですが、

1. カードの属性。  
花、日、月、雪、宙、無の6種類。ちなみにこのカードは宙(そら)です。

2. EX.

このカードが捻出できるコスト。このカードが捻出できるコストは宙を2点分です。

3. キャラの名前。

東方銀符律を始める  
とマイナーなキャラ  
の名前も覚えられ  
ます。……マエリベ  
リー・ハーンとか。



4. このキャラが登場するのに必要なコスト。  
つまり、この諏訪子が登場するのに必要なコストは宙3点と無2点、ということです。

5. このキャラが登場可能な位置。

このキャラは中央列にしか登場することができません。ちなみに盤面はこんな感じ



6. キャラの状態ス。  
左上がAP、右上がD-P、下がSPとなっています。詳しい説明は後々。

7. キャラの基本能力。  
様々な基本能力があります。

● ダッシュ

このキャラが参加しているバトルの結果は、このキャラの攻撃を優先して判定します。このバトルで対戦キャラがダウンする場合、このキャラはダウンしません。

- アグレッシブ  
このキャラは登場したターンから攻撃・特殊能力の使用ができます。
- ステップ  
このキャラを前後左右に1マス移動します。自分のターン中、このキャラが未行動状態の間だけ、1ターンに1回だけ使用できます。バトル中には使用できません。
- サイドステップ  
[ステップ]と同じです。ただし、左右にしか移動できません。
- オーダーステップ  
[ステップ]と同じです。ただし、前後にしか移動できません。
- ジャンプ  
好きな自分のフィールドに移動できます。使い方は[ステップ]と同じです。
- エスケープ  
このキャラを破棄します。自分のターン中、このキャラが未行動状態の間だけ、1ターンに1回だけ使用できます。バトル中には使用できません。
- ボーナス  
このキャラが参加したバトルの結果、対戦キャラがダウンしたとき、併記された効果进行处理します。(バトルによるダウンのみです。)
- デッキ・ボーナス  
このキャラが参加したバトルの結果、相手のデッキがダメージを受けたとき、併記された効果进行处理します。(バトルによるデッキへのダメージ『破棄』のみです。)
- ペナルティ  
このキャラが自分の場から離れたとき、併記された効

果を処理します。  
(バトルによるダウン[破棄]だけでなく、手札に戻されたりゲームから取り除かれた場合にも有効です。)

# 8. 特殊能力。

条件があった時に使えます。このキャラは登場時しか使えませんが、いつでも使えるキャラなどもあります。



ちなみに、このキャラの特殊能力を使用するためには書かれているコストが必要

です。

# 9. キャラのイラスト。

このゲーム一番の魅力と言っても過言ではありません！全てのカードに色々な絵師さんの可愛い、カッコいいイラストが描かれています！

これでキャラカードの説明は終わりです。

次にイベントカードの説明です。



イベントカードは条件が指定されていない限り何時でも使用することができます。コストの支払い方、カードの見方もキャラ

と同じです。手札から突然飛んでくるので奇襲性が高いです。

これらのカードの他にも『エリアカード』と『アイテムカード』があるのですがめんどくさ……一度に多く覚えるのは難しいので、説明は放棄します（白目

では、フェイズの説明に戻りましょう。メインフェイズに行えることは、『キャラを登場する』『キャラの能力を使う』『イベントを使う』『攻撃をする』の4つです。キャラの登場、キャラの能力、イベントの使用は前述の通りです。なので、攻撃についての説明をします。

攻撃の手順はまず未行動状態の AF キャラで『攻撃宣言』をします。そして、相手は同列の未行動状態の DF キャラで防御するか否かを選択します。ここでバトル突入。お互いにイベントや特殊能力を行使してキャラを支援したり相手の攻撃を止めたりします。防御キャラがいる場合はお互いの AP と DP をくらべて、AP より DP が劣っていたらそのキャラを破棄します。

両者のステータスを同時に比べるので、両者ダウンや攻撃側のダウンなどもあります。防御キャラがいない場合はそのキャラの元々の AP 分だけ相手のデッキを削ります。

フェイズの説明はこんなものでしょうか。わからなかったらツイッター辺りで私に聞いて下さい（決してステマでない） @galvashighwind

### 3 デッキの構築

デッキの組み方です。まず嫁を4投するところから始めましょう。……なにか？嫁を入れると士気も上がるのでなんの問題ありません！

さて、次にファッティの投入です。ファッティとは登場に5コスト以上を必要とするカードのことです。多くの手札を必要とする代わりにどれも協力的な効果を持っています。ただし、多く入れると事故になってしまうので4～8枚くらいが適正でしょう（デッキによる）

残りは構築済みデッキなどを参考にするといいと

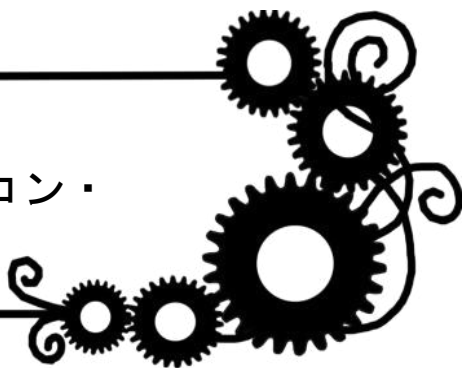
思います（投げやり……だって私もまだまだ下手ですし。

#### 4 終わりに

さて、ここまで読んでいたたきありがとうございます。これを読んだら、早速カードを買って銀符律を始めましょう！……？相手がいない？仕方ないです。競技人口少ないですし。そこでtwitter

ですよ。twitterの銀符勢と交流して、Skypeで銀符をする。そうすると始めはボコボコにされると思いますが、プレイングの指導などもしてもらえるのでどんどん上手くなります！また、ニコ生などに凸するのも手です。ネットって、便利ね(KONAMIではさいごに一つルールとマナーを守って楽しく銀符律！！  
ありがとうございました。

# マイコン・マメコン・ オウコン



著者：FreeBEAN

これは群馬高専電子情報工学科（以下、J科）のマイコン講義内容変更に関する物語風解説である。

## 1 終わりの始まり

それは 2010 年半ばのことであった。J 科では 1987 年の設立以来、S 大先生がマイコンに関する教育を担当していた。使用するマイコンは Z80 である。実習は PC98 シリーズ上で動くアセンブラを用いて機械語をプリンタ（ドットプリンタ!）から出力し、その内容を一人が読み上げ、別の一人が RAM に打ち込むという手法が用いられていた。これは、プリントアウト時のフォント上の問題による、0（ゼ

ロ）と O（オー）や 1（イチ）と l（エル）のような機械語の読み間違い、入力ミス（最初からやり直し）といった問題点があり、学生実験が苦行となっていた。さらに、Z80 の生産が終了し、秋葉原ですら入手困難にありつつあった。だが、S 大先生は J 科設立の大恩人。猫の首に鈴をつける鼠はなかなか現れなかった。

このような状況で Z80 による教育が続けられていた。しかし、さすがの S 大先生も 2012 年に定年を迎えることになり、授業科目の継承等を話し合うことになった。その場で、某教員が「マイコンの内容もアップデートする必要がある」と発言したことによ

り、マイコンに関する教育内容を変更することになった。当然のごとく「言いだしっぺの法則」が発動されたのだった。

このようにしてJ科のマイコン教育アップデートが始まった。

## 2 Z80 を継ぐもの

問題は、某教員がマイコンの専門家ではないことであった。マイコンに関しては学生時代の実験でゼッパチを使っただけである。まず、どのマイコンを使うかを決めなければいけない。このとき、某教員が考えたのは教育用としての適性である。できる限りシンプルな構造を持つマイコン、教育用実習装置がつかえるマイコン、なおかつある程度広く使われているマイコン。ついでに、なるべく国産品にしたい。

一番重要なのは実習装置として何を使うべきかであった。教育用としては一人一台を購入できる価格であり、できる限り多様な実習を行うためのオプションがあるものが望

ましい。困難な探索の末に、サンハヤトより発売されているマイコンボードを採用することになった。搭載マイコンはルネサステクノロジ（当時。現在はルネサスエレクトロニクス。以下、どちらもルネサスと表記）のH8シリーズ H8/3062BF である。2010年度末のことである。

しかし、H8マイコンに関する本は多数出版されているが、H8/3062BFに特化した本はなかった。（そして、現在も存在しない。）こういう場合はどうするか？ 「なかったら作ればいいじゃない？」そう、担当者自ら書くしかないのである。このようにして某教員は約1年間H8マイコンの勉強をすることになった。市販の書籍を買い込み、自分でプログラミングとデバッグをし、ときにはルネサスのマニュアルを読んで動作解析をした。そして、月日は流れ2012年2月に数々の不備を抱えながらもテキストが書き上がり、某印刷会社が発注することができた。

「Z80を継ぐもの」の誕生



である。

### 3 そしてオワコンへ

あらたなマイコン H8 が登場し、2012 年 4 月より J 科では新たな内容のマイコン講義が始まった。一部学生は某教員の名前からこの講義をマメコンと呼ぶようになった。

プログラムの開発は MS-Windows マシン上の統合開発委環境で行い、アセンブリ言語はもちろん C 言語でもプログラミングができるようになった。ロードモジュールは USB 経由で H8 マイコン内のフラッシュメモリに書き込むので機械語の打ち間違いはない。以前より良い環境でマイコンの教育ができるようになった。

しかし、光もあれば闇もある。講義開始後まもなく衝撃的な報道がなされたのだった。

ルネサスが H8 マイコンをサポートプロダクト入りさせ、新規採用非推奨という状態になってしまった。事実上の生産中止予告である。これに伴い、2012 年 7 月 10 日リリースの製品情報でサンハヤトがマイコンボードの販売終了を決定した。

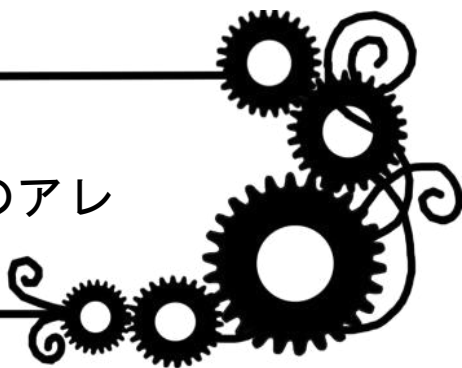
学生がソースファイルのデバッグのたびにメモリに書き込むという無茶な使い方をする。やがて、フラッシュメモリの耐用回数を超えて壊れてしまう。だが、そのときには代わりの H8 マイコンは既に存在しない。

このようにして、アップデートを目指したはずのマイコンの教育計画は早くもオワコンとなってしまったのであった。

その後、J 科のマイコン教育がどのようなものかを知るものは誰もいない……。

the END.

# ArchLinux のアレ



著者：礫御飯

## 1 はじめに

非電算部員の礫御飯と申します. 外部からの投稿として本記事を掲載して頂いています.

この記事のねらいは, Linux を使うことの楽しさを味わってもらうことです. インストールや環境構築の手順の紹介はドキュメントや解説サイトにお任せします. 「Linux とはなにか」よりも「Linux をどう使うか」に重点をおいています. 基本的な用語しか用いませんが, わからないことがあればその都度調べてみてください. Linux を使っていく上でも重要になってくると思います.

想定読者は, 初心者～初級者程度です. 私自身初級者程度のため, それ以上の内容は期待しないでください. 特に, 開発に Linux を用いたい人向けの内容ではないため, ご注意ください.

誤解しないでいただきたいのは, この記事は他の OS から Linux への乗り換えを推奨するものではないことです. あくまでもお遊びでの使用を想定しています. また, Ubuntu など他の Linux ディストリビューションが悪いと言いたいわけでも, それらの環境の構築が簡単と言いたいわけでもありません. ArchLinux が Linux を

楽しむための良い選択肢として示しているだけであることを念頭に読んでください。

## 2 Linux を使う利点

Linux の利点の話題になると、すぐに「開発で云々」「オープンソースが云々」などという話が出ます。もちろんそれも多くの人が Linux を使用する理由の 1 つもなっていますが、お遊びで使う上でもっとも重要なのは「使っていて楽しい」ことです。すべての作業を Linux に移行する必要はありません。仮想マシンでもデュアルブートでも、お遊びで使うためにも多くの選択肢があります。

また、Linux で遊ぶことで得た知識は、サーバなどの実用的な環境構築にも役立つでしょう。ただし、完全に流用してはいけません。ディストリビューションの選択や、各種設定などにおいてその環境では慎重になる必要があるかもしれません。

## 3 ArchLinux を勧める理由

### 3.1 Linux の楽しみ

Ubuntu などのディストリビューションで、簡単な操作でインストールが完了し、多くの設定が自動で済まされ、初めから多くのソフトウェアがインストールされている環境を使用して、ブラウザやメール、ゲームなどの動作を確認したとします。これだけで自分が何をしてこうなったのかを理解できる人はいないでしょう。これでは、Windows よりも不便で面倒なものという認識で終わってしまい、Linux の楽しみを感じられません。ここで言う Linux で遊ぶことは、ウェブブラウジングやゲームではないのですから。

ArchLinux の環境の構築はすべてコマンドラインで行う必要があります。初めてでは少し戸惑うかもしれませんが、大丈夫です。Installation Guide や

Beginners' Guide に基本的なことはすべて書いてあります. どの設定ファイルが何を意味しているか, ひとつひとつ理解しながら進めることで, 自分の手で環境を作り上げる楽しみを感じられるでしょう. GentooLinux でも良いのですが, ArchLinux に比べて要求されるハードウェアの知識が高いため, ArchLinux のほうがお手軽です.

もちろん, ためしに Ubuntuなどをインストールしてみたり, LiveDVDなどで遊んでみるのも十分有力な選択肢です. ただ, 少しでも面白いと感じたら, 手動で細かい設定を試してみたり, 別のディストリビューションをインストールしてみたりすることで, 楽しみも広がるのではないのでしょうか.

また, 明確な用途や使いたいアプリケーションが定まっていて, それを実現する手段として Linux を使用する場合は話は別です. そのときは, 用途に適

した Linux ディストリビューションを選択するべきです.

### 3.2 ローリングリリース

ArchLinux はローリングリリースであるため, 明確なバージョン区別がなく, 面倒なアップデート作業は必要ありません. あなたは必要なときに後述の pacman でパッケージを更新するだけで良いのです.

### 3.3 パッケージ管理

パッケージマネージャ pacman と, AUR (ArchUserRepository), ABS (ArchBuildSystem) の存在は偉大です. pacman は高速に動作するパッケージマネージャで, ストレスのない使用が可能です. パッケージは常に新しいものが維持され, 種類も豊富です.

### 3.4 ABS

ABS (ArchBuildSystem) は, ソースコードか

ら `pacman` で管理可能なバイナリパッケージを作成するシステムです。ソースコードのダウンロード先 URL や、コンパイル等の手順が記された `PKGBUILD` というファイルを用いて `makepkg` コマンドを用いて読み込むことで、自動的にコンパイル、パッケージングを行います。

設定したリポジトリにあるすべてのソフトウェアの `PKGBUILD` が `ABS` ツリーに存在します。`ABS` ツリーは `abs` コマンドでいつでも更新が可能です。

`ABS` を使用することで、必要なパッチを当ててコンパイルしたり、あなたのシステムに最適化してリビルドすることが可能です。

これは絶対に必要なものではありませんが、環境を自分色に染める楽しみを味わうことができます。

### 3.5 AUR

AUR (ArchUserRepository) は、コミュニティ

により運営されている上述の `PKGBUILD` が集められたリポジトリです。ユーザがこれを投稿することもできます。これにも `makepkg` コマンドを使用しますが、`ABS` とは別物です。

AUR には非常に多くのソフトウェアが登録されているため、公式のリポジトリにないものを使いたい場合に非常に便利です。多くの人が Linux を使いたいと思う理由の 1 つである `mikutter` もこれに登録されているため、簡単にインストールが可能です。

`Yaourt` など、AUR へのシームレスなアクセスが可能な `pacman` のラップを用いれば、もはや `pacman` でパッケージをインストールするのと手間は変わりませんが、公式にサポートされていないパッケージであることを忘れないでください。`PKGBUILD` に悪意のあるコマンドが記述されていれば `makepkg` をそれを実行してしまいます。ほとんどの

パッケージは安全ですが、使用には注意を払ってください。

## 4 ArchLinux を使う

さて、前置きが長くなりましたが、それでは ArchLinux の世界へ入ってみましょう。インストールする前に ArchLinux を体験してみたい方は、mikutter LiveDVD や、派生ディストリビューションの ArchBang を試してみてください。

### 4.1 インストールする

まず、あなたは物理マシン・仮想マシンのどちらにインストールするかを決める必要があります。もしあなたのマシンがある程度パワフルなものなら、まずは仮想マシンにインストールしてみるべきです。パーティションの選択を間違えると、最悪あなたの大切なデータを失うことになってしまいますが、仮想マシンであればその心配はありません。大丈夫、仮想マシ

ンであっても、十分に楽しむことはできます。

インストールについては、ArchWiki の Installation Guide と Beginners' Guide、各種解説サイト等を参照してください。私が改めて書くよりも格段にわかりやすく、最新の情報が手に入ります。ここで注意することは、必ず英語の Installation Guide, Beginners' Guide の情報をメインに使うことです。日本語のガイドや、外部の解説サイトでは、情報が古いことがあります。ドキュメントはわかりやすい英語で書かれています。コマンドをメインに見て、周りの文を参考にする程度で問題ありません。

また、初心者の方が一番躓くのは、パーティションの選択でしょう。インストールの失敗は主にこれが原因であることが多いです。`/dev/sda1` `/dev/sdb2` や、`(hd 0,0)` など Windows と異なる表記に戸惑うかと重います。本当に自分がインストールした

い場所なのかどうかを確認してください. これを間違えると, 物理マシンではデータを失いかねません. 慎重に行ってください.

おっと, ブートローダの設定を間違えてしまいましたか? 大丈夫, インストールメディアから起動して再度 chroot して設定を変更してください. 多くのソフトウェア的なトラブルは, この方法で解決が可能です. あなたの ArchLinux の環境が完全に壊れたときは, ハードウェアの故障を意味するでしょう.

Welcome to ArchLinux. 正常に起動したようですね. おめでとうございます. 引き続きドキュメントを参照して, 各種設定や一般ユーザの追加を行ってください.

#### 4.2 時計が合わない!ここは日本だゾ!??!?

ArchLinux をインストールした知人で, 時計がずれる方が多くいまし

た. 無闇にコマンドを叩くと, ハードウェアクロックをずらしてしまうかもしれません.

ここで, Linux の時計について簡単にお話させていただきます. 時計は, ハードウェアクロック (CMOS 時計) とシステムクロックの 2 種類があります. 起動時に 1 度だけハードウェアクロックを読み, 移行はそれをもとに定められたシステムクロックが使用されます. システムは, ハードウェアクロックが UTC か localtime どうかを判定する必要があります. /etc/adjtime にこの設定を記述します.

時計がずれている方の多くは, 9 時間のズレとなっているのではないのでしょうか. おそらく多くの方のハードウェアクロックは, localtime(UTC + 9:00) になってるかとおもいますので Beginners' Guide に従って, hwclock -systohc -localtime を実行してみてください. もしこれでもずれているの

なら,/etc/adjtime を見てみてください.cat コマンドでも,テキストエディタで開くのも構いません.1 行目が 0.000000 (数字) (数字) ,3 行目が LOCAL となっているか確認してみてください.(もちろん (数字) の部分は何か数字が入っています.) 0.000000 は時刻の補正量です. ハードウェアクロックと同様にするため,0.000000 にします.LOCAL は localtime を表します.UTC を使用する場合,ここが UTC になります.もしこれでもずれているなら,ハードウェアクロックを確認してみてください.

#### 4.3 好みの環境を構築する

多くの入門ガイドでは,この後 X のインストールを行っておしまいです,あなたが X を必要としないのなら無理にインストールする必要がありません.X がなくても fbterm などを用いれば日本語の表示もできますし,画像・動画の再生が可能なソフ

トウェアもあります.

しかし,多くの人は X をインストールすることになると思います.常用しているユーザを video グループに追加するのを忘れないでください.

忘れがちなのが,KMS (KernelModeSetting) です.ArchWiki の KMS の項目や,グラフィックドライバ (intel や nvidia) の項目を参照して設定してください.

デスクトップ環境・ウィンドウマネージャの選択は,大きな楽しみの 1 つでもあります.豪華で派手なデスクトップ環境を選択しても,シンプルで軽量のウィンドウマネージャを選択しても構いません.ウィンドウマネージャは,スタック型 (フロート型) とタイル型の違いに気をつけてください.どちらも利点がありますが,私はタイル型がお勧めです.とくにこだわりがないのなら,LXDE や xfce4 が良いでしょう.また,興味があ



れば是非タイル型ウィンドウマネージャも使用してみてください.AwesomeやXmonadがタイル型です.

WikiのALSAの項目を参照して,サウンドの設定をしたり,iBusをインストールして日本語の入力ができるようにしてもいいでしょう.必要な情報の多くは,ArchWikiに記されています.

AURやABSを使いたいのですか?大丈夫,すべてArchWikiの通りの操作で大抵の場合は使用可能です.makepkg.confの編集には,Gentoo WikiのSafe CFALGSなどが役に立つでしょう.

#### 4.4 情報を集める

まずはmanコマンドでマニュアルを読んでみましょう.内容をすべて理解する必要はありません.マニュアル中から必要ところを探しだして読むのは慣れが必要ですが,長く使っていれば問題なくなるはずです.

もしあなたが ArchLinux を使っているとき,ArchWiki だけでは解決できない問題に立ち向かう必要が出てくるでしょう.しかし,それはあっさり解決してしまうかもしれません.

検索エンジンで,エラーメッセージなどで検索してみてください.多くの場合,様々なディストリビューションの Wiki やフォーラムがヒットするかと思います.違うディストリビューションの場合,設定ファイルが置かれている場所が異なる可能性があります,内容はおおよそそのまま使用可能である場合が多いです.このとき注意が必要なのが,それが本当に使用しているソフトウェアの設定なのかということです.例えば,ArchLinux では systemd が採用されていますが,他のディストリビューションの情報は OpenRC など違うものかもしれません.

また,pacman は常に新

しいパッケージをインストールすることが可能ですが、稀にそれに不具合がある場合があります。特に、システムにとって重要なパッケージや、よく使用するソフトウェアのメジャーアップデートに注意してください。これらの情報は、ArchLinux のニュースや、フォーラムで得ることができます。メーリングリストに登録しておくのが最も確実な方法です。

#### 4.5 ArchLinux を直す

もしパッケージの更新や、その他の理由で正常に起動しなくなってしまうても大丈夫です。まずはフォーラムなどで同様の現象が発生していないか確認しましょう。

インストールメディアから起動して chroot すればすぐに修復作業を行うことができます。

## 5 ArchLinux と付き合う

使いたいハードウェアやソフトウェアが使えるようになったら、もしかしたらあなたが日常行っていた作業のほとんどが Linux で済んでしまうかもしれません。自然と Linux を常用することになるのではないのでしょうか。

ここまでできれば、ユーザとして大抵のことはできるはずです。バグの報告をしたり、AUR にパッケージを投稿したり、プロジェクトに貢献したいと思うようになるかもしれません。すべてはあなたの自由です。

## 6 諦めてしまった人へ

インストールや環境の構築がうまくいかなくて、嫌になってしまうこともあるかもしれません。大丈夫です、無理に ArchLinux に固執しなくても、他のディストリビューションでも十分に楽しめます。それに、それまでの作業はすべて

が無駄になってしまったわけではありません. いかもう一度挑戦してみてください.

いものとなってしまうました. 英語のドキュメントを翻訳したような文章にしようと思ったのですが, これではただの上から目線な文章では……??

## 7 あとがき

インストールや環境の構築についての方法の解説は山ほどあるので, できるだけ解説を省いた文章を心がけましたが, 内容の薄

筆者自身Linuxをまともに使えているか疑問なレベルですので, 間違えている箇所もあるかと思います. 気がついたら末尾の連絡先にこっそり教えてください.

## 8 参考文献

最終閲覧確認は, 初版発行日です.

- ArchLinux  
<https://www.archlinux.org/>
- mikutter ユーザ会  
<http://yuzuki.hachune.net/>
- Man page of HWCLOCK  
<http://linuxjm.sourceforge.jp/html/util-linux/man8/hwclock.8.html>

2013/08/19 礫御飯 reki.rice@gmail.com



# systemd で Gentoo を使う

著者 1\_k\_\_

## 1 はじめに

私の開発用 OS は Gentoo Linux です。が、最近 systemd が流行っているのは事実。Gentoo にはオリジナルの init である OpenRC が採用されています。しかし Archlinux の爆速起動を見ていると試してみたくなってしまう。これはいままで使っていた環境に systemd を導入するまでの作業記録です。

## 2 systemd をインストール

### 2.1 emerge!

まずはさっくりと

```
1 #euse -E systemd
2 #emerge -uDN world
```

16 個ほどのパッケージが(再)インストールされました。

### 2.2 カーネルの設定

Gentoo Wiki を参考に作業を行います。

```

1 General setup --->
2 [*] Control Group support
3 [ ] Enable deprecated
   sysfs features to
   support old userspace
   tools
4 [*] Networking support
   --->
5 Device Drivers --->
6 Generic Driver Options
   --->
7 [*] Maintain a devtmpfs
   filesystem to mount at
   /dev
8 File systems --->
9 [*] Inotify support for
   userspace
10 [*] Filesystem wide access
   notification

```

この他に、

```

1 File systems --->
2 <*> Kernel automounter
   version 4 support (
   also supports v3)
3 Pseudo filesystems --->
4 [*] Tmpfs virtual memory
   file system support (
   former shm fs)
5 [*] Tmpfs POSIX Access
   Control Lists
6 [*] Tmpfs extended
   attributes

```

## 2.3 ブートローダ

GRUB の設定。私は宗教上の理由で未だ GRUB Legacy を使っています。

```

1 title Gentoo Linux-3.9
   with systemd
2 root (hd0,0)
3 kernel /boot/vmlinuz
   -3.9.11-gentoo-r1 root
   =/dev/sda1 init=/usr/
   lib/systemd/systemd

```

grub.conf にこのようなエントリを追加。再起動しましょう。

## 3 システム設定

### 3.1 ロケール

うまく起動出来ましたか？とりあえず root でテキストログインして設定をしましょう。ロケール、キーマップの設定には `localectl`、ホストネームの設定には `hostnamectl` コマンドを使用するようです。

```

1 # hostnamectl set-hostname
   dellxps
2 # locale set-locale LANG
   =ja_JP.UTF-8
3 # localectl set-keymap
   jp106
4 # localectl set-x11-keymap
   jp

```

### 3.2 xdm

私は gdm を使っています。そのまま systemctl enable をしても.service が無い！と怒られてしまうので、systemd-units を emerge します。systemd オーバーレイにあります。ingnome3 フラグをお忘れなく。

```
1 # layman -a systemd
2 # USE=ingnome3 emerge
  systemd-units
3 # systemctl enable gdm
```

## 4 課題

現状の問題点。いくつかの問題があるためこのままでは常用できません。

### ネットワークデバイス名

systemd の Predictable Network Interface Names のせい。当然のネットワークインターフェース名・・・？ってなんじゃ。見た感じデバイス名が enp なんちゃらや wlp なんちゃらに変わっ

ています。いままで認識された順(?)であったネットワークデバイス名を、固定しようというものらしい。OpenRC の環境にも合わせるため、とりあえずこれを行わないようにします。/etc/udev/rules.d/内に空のルールファイルを作成。ArchWiki に則り、/dev/null にシンボリックリンクを張ります。

```
1 ln -s /dev/null /etc
  /udev/rules.d
  /80-net-name-
  slot.rules
```

これで元通り。

うるさい起動画面 起動画面がなんだかうるさい。メッセージがたっくさん。/etc/systemd/system.conf に

```
1 loglevel=error
```

ブートオプションもいじります。


```
1 | kernel /boot/vmlinuz  
  | -3.9.11-gentoo-  
  | r1 root=/dev/  
  | sda1 init=/usr/  
  | lib/systemd/  
  | systemd systemd.  
  | log_target=null
```

## bumblebee

NVIDIA Optimus テクノロジーに対応した GPU を積んだノート PC で、外部 GPU を使った 3D レンダリングを行うためのもの。vgl を利用しているため設定をどこに書けばいいのかわからない & .service ファイルがない。未解決。

## 5 まとめ

起動時間は短くなりましたが、正直ここまでやって systemd に移行する必要があるかは疑問です。Gentoo はオリジナルの init である OpenRC を採用しています。まだ systemd に対応していないパッケージがいくつもあります。依存関係が解決できていない場合もありました。



# ソーシャルゲームについて

著者 ホリヤコロヌ

## 1 ソーシャルゲーム とは

ソーシャルゲームとは SNS（ソーシャルネットワークキングサービス）上で提供されるオンラインゲームまたはブラウザゲームである。本来は SNS の中のサービスとしてパソコン上で行うゲームのことを指していたが、日本では携帯電話向けのブラウザ向けものをさして使うのが主流である。また、スマートフォン向けの SNS と関係のないネイティブアプリのようなゲームもソーシャルゲームと呼ばれることが多いが、厳密に言うとソーシャ

ルゲームとは異なる。この記事ではスマートフォン向けソーシャルゲームについて扱う。

## 2 主なゲームの内容

主要なブラウザ向けゲームでは主にスタミナを消費して進むモード、他のユーザと対戦をするモードがある。その他にも様々な内容でユーザを楽しませる。

### 2.1 スタミナを用いるモード

スタミナを消費して進むモードでは、ただ画面をタップするだけでゲームを進行できる。各ステージやエリアが用意されしばらく進むとステージク



リア、ステージをいくつかクリアするとエリアクリアとなる。その過程ではゲーム内通貨やカードとよばれるもの入手でき、あるいは他のユーザとの対戦の時に奪い合うことになるアイテムを入手できる。また、ステージの一定数クリア毎にボスが出現しカードを用いて対決することになる。勝利すると便利なアイテムやゲーム内通貨がもらえる。しかし、ブラウザ向けでボスに負けることは珍しい。

## 2.2 対戦機能

他のユーザとの対戦では主にアイテムの取り合いをする。対戦にはカードを用いるが他に対戦用のコストを必要とする。カードにはコストと攻撃力と防御力と効果があり、単純にお互いに攻撃側の攻撃用コストの範囲でのカードの組み合わせの攻撃力と効果による補正の値と、防御側の防御用コストの範囲でのカードの組み合わせの防御力と効果による補正の値の大小で戦う。取り合いをするア

イテムにはシリーズがあり、あるシリーズをコンプリートすると報酬アイテムが貰える。対戦の勝利時にはゲーム内通貨を貰える場合が多く、わざと対戦に負けてくれるユーザもおり道場と呼ばれることが多い。

## 2.3 ガチャ

ブラウザ向けソーシャルゲームの課金対象はカードを引くガチャ、定期的開催される課金を促すイベント等である。ガチャでは主に課金ガチャと友情ポイントガチャの二種類がある。課金ガチャで出現するカードは基本的に友情ポイントガチャの性能をはるかに上回る。主にレア度の低いカードが弱く友情ポイントガチャで出現し、一定のレア度から下は課金ガチャでは出現しない。課金ガチャでのよりレアなカードの出現率はゲームによって異なるが、最もレアなカードの出現率を 1.0 % 以上にするというソーシャルゲームを運営する諸企業の中での取り決めがある。

課金ガチャには期間限定のカードが取り入れられ、一定期間ごとに更新される。課金ガチャにはさらに、「コンプリートガチャ」「STEP ガチャ」「BOX ガチャ」など、ユーザの目当てのカードより入手困難にさせより多くの課金を促す種類がいくつかあり、この内のコンプリートガチャについては消費者庁により違法であるとされ禁止されている。

## 2.4 イベント

定期的に行われるイベントには多くの種類があるが多くの課金ユーザの目的はイベントのランキング報酬である。このランキング報酬というものは主に上位に対しての強力なカードであり、課金ガチャの同様なレアカードに匹敵する。また、ランキング戦のためユーザの多いゲームではより多くの課金が必要になる傾向があり、より人気の高いゲームほど多くの収入を見込める。

## 2.5 トレード

大抵のソーシャルゲームにはトレード機能が実装されている。ユーザ間である程度自由にカードやゲーム内通貨やアイテムのトレードが行える。目当てのカードをそのカードが欲しいユーザにむけてトレードの条件を出し、他のユーザがその条件に同意してトレードをすると成立する。レア度の高いカードはゲーム内の課金アイテムや同様なレア度の高いカードとトレードされることが多く、レア度の低いカードについては同様なレア度の低いカードや少額のゲーム内通貨とトレードされる場合が多い。ソーシャルゲームの初期ではかなり自由にトレードが行えたので、RMT（リアルマネートレード）と呼ばれる現実の世界で使われる通貨とカードを交換する行為が横行したがソーシャルゲームの運営の

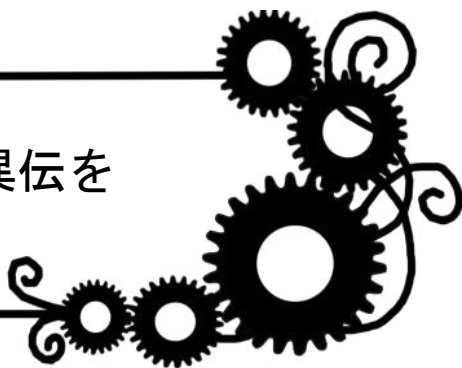
利益にならないために禁止されている場合が多い。

### 3 まとめ

このようにソーシャルゲームでは課金ユーザが優位に立てるゲームの設定や、通常のゲームに加えて定期的な課金ユーザのための企画が用意されている。このほかにも最近では「美麗」ということが推しのゲームが増え、その多くが萌えキャラのような絵柄をカードに取り入れ我々を釣り上げようとしている。このような

物に課金をすることを否定はしない。自分のお金をどのように使おうがそれは他人の関わる所ではないからだ。しかし、課金をする前に考えてほしいことがある。それはなんの目的を持って課金をするのかということである。例えば「美麗」なカードの絵柄が欲しいなら課金を剃る必要はない。しかし〇〇のカードがほしいと強く願うならそれは課金をするしかないだろう。課金をする前には一度冷静になって目的を再確認して本当に必要かどうかを考えるべきである。

# プチコンで靈異伝を 作ってみる



著者 しーきゃ

## 1 はじめに

DS で BASIC が書ける  
プチコンというソフトを  
ご存知でしょうか？昨年  
の D 言語にもプチコン  
ネタを書いたので、知っ  
てる人は知ってることと  
思います。今年は、東方  
project の第一作である東  
方靈異伝をプチコンで作  
ろうと挑戦中です（未完  
成です、記事のタイトルが  
進行形なのもそのためで  
す）。現時点でのプログラ  
ムの QR コードが CD に  
入っているのでぜひ実行  
させてみてください。

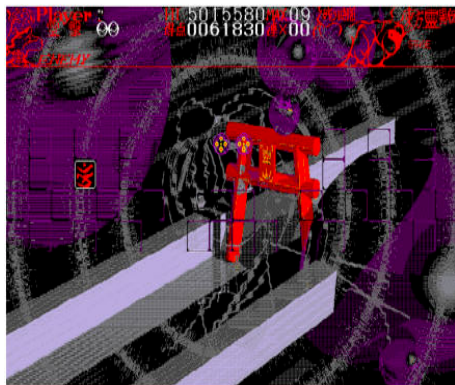
## 2 作り始めたきっかけ

1. 画面端で円が反射するだけのプログラム作った
2. 重力をつけてみた。
3. 「あれ、これって靈異伝っぽくない？」
4. あとは勢いに任せた。後悔はしてない。

## 3 リソースの作成～グラフィック

陰陽玉とかのスプライトはスクリーンショットを見ながらひたすらドット打ちをするだけだったので根気さえあれば何とかなる。（靈夢のドットが打ち終わってないけど。）問

題はステージ背景のほうだ。スクリーンショットを撮ったところでスプライトが邪魔で想像以上に作業がしづらい。で、手動で補正してみた。ペイントで。こんな感じになりました(右の画像)。あとは、これをトリミングするなり縮小するなりしてDSの画面サイズに合わせたわけですが、256×192のドット絵を2枚も打ち込むのはさすがにきつい。この作業だけで数か月かかった。なんやかんやでタイトル画面と前半ステージの背景はできたので、グラフィックに関してはここでひとまず終了ということに。



#### 4 リソースの作成～B

##### GM

東方といえば、BGM も大きな魅力の一つですね。

無音でプレイしているのでは東方をやっている感が全然でません。ところで、プチコンmk2にはMML再生機能がついています。(MMLとは

MIDI をテキストファイルで書いたようなものです) プチコンの最大同時発音数は8パートなのでFM音源ぐらいなら再現できるはず! というわけでとりあえず永遠の巫女を作ってみることにしました。靈異伝といえはやっぱりこの曲でしょう。(最終面なんてなかった。サリエルなんていなかった) とは言ったもののう p 主の耳コピ力はかろうじてメロディが探れる程度です。さて困った。しょうがないので怪綺談でもやりながら考えていたら気が付いた。怪綺談の MusicRoom には鍵盤が表示されるじゃないか。で、あとはひたすら目コピタイムです。背景のコピーですでに疲れているのに、今度は動体視力まで酷使する羽目になるとは。ブルーベリーください。ともかく永遠の巫女に関してはそれっぽくなったので5面まではこれでokでしょう。タイトル画面は適当です(え

## 5 プログラムの作成

リソースだけ作ったところでプログラムを書かなくてはそもそもゲームになってない。今回のプログラムはゲームなので前回と違って状態遷移が必要になってくる。これがCのプログラムだったらメインループから switch 文で各関数に飛ばさせるところだが BASIC でそれをやるのは割とめんどくさい。どうせ BASIC なんだから goto を使いまくってみた。つまりは各状態をモジュール化してそのループをぬけたら別のモジュールの頭に付けたラベルへ goto! ややスパゲッティ気味になってるけど飛ぶタイミングは規則的だし、これはこれで BASIC らしいからあえてこの方針で組んでみた。


実際は各サブルーチンとかにもいろんな Tips を使っているのだけれどもソースを読みながらそれらを探すのも楽しみの一つだと思うので、ここでの説明はこれぐらいにとど

めておきたいと思います。  
文字数の都合もあります  
しね。別に書くのがめん  
どくなっただけではない  
ですよー（棒

## 6 あとがき

はじめににも書いた通  
りこの project はまだ製作

途中で発展途上です。プ  
ログラムの更新がある際  
は、GNCT 電算部のブロ  
グでお知らせする予定で  
す。最後に身内の人たち  
へ。プチコンを持ってい  
て、かつ暇でしょうがない  
方、スタッフ募集中です。



# 日本人の日本人による日本人 のためのプログラミング

著者 莢

## 1 はじめに

「こんにちは。」と表示。  
「初めましての方は初め  
まして、莢です。」と表示。  
「え？挨拶がおかしい？」  
と二択。

もし、それがはいならば  
「これは、なでしこ  
というプログラミング言  
語です。」と言う。

もし、それがいいえなら  
ば

「結婚しよ」と言う。

どうも改めましてこんに  
ちは。最近、みなさんは日  
本語ときちんと向き合っ  
ていますか？あ、こちら  
は群馬高専電算部の部誌  
で合っていますよ。ご安  
心を。「俺は情報系だから

日本語なんて使えなくて  
も構わない！」という貴  
男、「Jamaneseなんてs-  
mart じゃないわ！」とい  
う貴女。のんのん、日本人  
が日本語使わなくてどう  
するの？

こほん、先ほどのなにか  
がおかしい挨拶、こちらは  
プログラミング言語、「な  
でしこ」の一例です。ち  
なみにあちらを実行する  
と... おっと続きは次以降  
のセクションで学んで行  
きましょう。めくるめく  
なでしこの世界へ、よう  
こそ。

## 2 なでしことは

なでしこは、クジラ飛  
行机（くじらひこうづく  
え）が制作したインタプ



リタ方式のスクリプト型プログラミング言語である。動作確認済みの OS は、Microsoft Windows 98/Me/2000/XP/2003/Vista/7/8。(wikipedia より引用)

なでしこ、というところ... 日本代表の女性サッカーチームが思い浮かびますね、お花が先に思い浮かんだ貴女、莢までご連絡を。あ、10代以下で。すいません、話がそれましたね。プログラミングのなでしこ、こちらのなでしこジャパンが名前の由来だったりします。確か、開発時に作者の方がなでしこジャパンの試合を見ていた... とか。古い記憶なので信じないで下さい。

さてこちらの言語、プログラムをかじったことがある方なら名前を聞いたことがあるかもしれませんが。プログラマの中では「エグい」「怖い」「プログラミング言語の歴史を感じる...」と好評なこちらなでしこ。どうされました、何か問題でも？なでしこに魅了されてしまう人も多いでしょう。是非、着物美

少女にプログラミングしてもらいたいですね。全国 73 億人の着物美少女の皆様、是非この記事を読んでですね...

### 3 ようこそなでしこへ

さて、早速なでしこの構文をいくつかご紹介します。プログラミング経験のない方向けに書きますので物足りないと感じる方もいらっしゃるかもしれませんが、お付き合い下さい。

#### 3.1 コード例

一つのコードを辿る形でなでしこの基本をおさえていきましょう。

今回は、京都のお座敷遊び、「とらとら」を簡単に書いてみました。ルールは簡単、ジェスチャー表現を使ったジャンケンで

す。わ とう ない和唐内(やりでつく)、虎(よつんばい)、お婆さん(杖をつく)の組み合わせで、和唐内は虎に強い、お婆さんに弱い。虎は、お婆さんに強い、和唐内に弱い。お婆さんは、和唐内に

強い、虎に弱い。掛け声は  
「とらとらとーら、とーら

とら♪」。ただのじゃんけ  
んとか言った人、怒らない  
から先生の部屋に来なs...

1 「おこしやす。」と表示。

2  
3 続行はそれ。

4 「とらとらしはりますか？」と二択。

5 もしそれがはいならば、

6 続行に1を代入。

7 もしそれがいいえならば、

8 続行に0を代入。

9  
10 「とらとらとーらとーらとら♪」と表示。

11  
12 勝ちはその。

13 勝ちに0を代入する。

14 負けはその。

15 負けに0を代入する。

16  
17 続行 $\neq$ 0の間

18 「どれを出しはりますか？ {改行} 1:和唐内 2:虎  
19 3:お婆さん」と尋ねる。

20 自分はそれ。

21 自分 $>$ 3または自分 $<$ 0の間

22 「どれを出しはりますか？ {改行} 1:和唐  
23 内 2:虎 3:お婆さん」と尋ねる。

24 自分はそれ。

25  
26 相手はそれ。

27 相手に3の乱数+1を代入。

28 「うちが {相手の手}、お客様が {自分の手}、

29 どすえ。」と言う。

30 自分と相手の勝敗判定。

31 勝敗はそれ。

32 勝敗で条件分岐

33 1 ならば、

34 勝ち = 勝ち + 1。

35 2 ならば、

36 負け = 負け + 1。

37 「もういっかいしはりますか？」と二択。

38 もしそれがはいならば

39 続行に 1 を代入。

40 もしそれがいいえならば

41 続行に 0 を代入。

42  
43 「{勝ち} 勝 {負け} 敗どすえ。」と表示。

44 もし、勝ち > 負けならば、

45 「お客様お強いどすなあ。」と表示。

46 違えば、

47 もし、勝ち < 負けならば、

48 「ほな、さいなら。」と表示。

49 違えば、

50 「気が合いますなあ。」と表示。

51  
52 ●手 (人の)

53 もし人 = 1 ならば、

54 「和唐内」で戻る。

55 もし人 = 2 ならば、

56 「虎」で戻る。

57 もし人 = 3 ならば

58 「お婆さん」で戻る。

59  
60 ●勝敗判定 (自分と相手の)

61 判定はそれ。

62 判定に自分-相手を代入。

64  
65  
66  
67  
68  
69

もし判定 = 2 または判定 = -1 ならば、  
「お強いどすなあ。」と言う。1 で戻る  
もし判定 = -2 または判定 = 1 ならば、  
「ごめんやす。」と言う。2 で戻る  
もし判定 = 0 ならば、  
「あいこどすえ。」と言う。0 で戻る

### 3.2 解説

これ、プログラムのコードですよ?! え g... 綺麗でしょ! 私は全角英数字アレルギーなので使って

いませんが全角英数にももちろん対応しています。12 行目のような全角記号も使用可能です。さて、以下の表を御覧ください。(複数用いた構文は一番最初の行数を記載します。)

行数	なでしこ	C	説明
1	「〇〇」と表示	<code>printf("〇〇")</code>	〇〇に書いてある文章を表示する
1	。	;	処理の区切り
3	<code>a</code> はそれ	<code>int a</code>	<code>a</code> を定義 その他後述
4	「〇〇」と二択		〇〇というメッセージのポップアップが出る 選択肢は「はい」, 「いいえ」
5	もし～ならば 処理	<code>if(～)</code>	条件式～が真 (＝1) ならば処理を行う
5	,	{	処理を一まとめにするもの
6	<code>a</code> に <code>b</code> を代入	<code>a=b</code>	<code>a</code> に <code>b</code> を代入する
17	～の間 処理	<code>while(～)</code>	条件式～が真の間処理を続ける
17	≠	<code>!=</code>	不等号
18	「〇〇」と尋ねる		〇〇というメッセージのポップアップが出る 入力ボックスに数字, 文字等入力する
18	{ 改行 }	<code>¥n</code>	改行する
28	「〇〇」と言う		〇〇というメッセージのポップアップが出る ボタンは <code>ok</code> 一つ
28	変数	<code>%d</code>	変数の値を出力
30	<code>a(と b と...)</code> の <code>function</code>	<code>function(a(,b,c...))</code>	関数呼び出し 助詞は種類によって異なる
32	<code>a</code> で条件分岐 <code>n</code> ならば 処理	<code>switch(a)</code> <code>case n: 処理; break;</code> : :	<code>a</code> の値や条件式によって処理が多分岐する
53	● <code>function(a(と b と...))</code> の	<code>function(int a, int b...)</code>	関数定義
55	<code>n</code> で戻る	<code>return n</code>	関数から値を返す

わかりやすいようにCで書いた時のものと比較(複雑な処理は省略)しています。いかがでしょうか、だいぶ優しいですよ、易しいではなく、優しいですよ

ね(目に)!

なでしこはフリーフォーマットではないので、や。は使わなくてもいいのですが、使ったほうが見やすいですかね?ただイン

デント (字下げ) がとても大切で、もし～ならばなどの下に条件を書く時はしっかりインデントして下さい。処理が一行の時は隣に書けばよいのですが。その他いろいろな文法がありますので気になった方は調べてみて下さい。

いかがでしょう、これで冒頭のすこしおかしいあいさつもうまく読めるのではないのでしょうか。だんだんなでしこが書きたくなってきたでしょう？

ゲーム自体のアルゴリズムに関しては、とらとらの手を数字に置き換えて (和唐内が 1、虎が 2、お婆さんが 3)、その差を出して勝敗を決めています。少し計算してみてください。

余談ですが、勝ち負けの表を配列で作ってそれに手の数字の要素数を入れて判定する方法もありますね。今回は簡潔にしてみました (別に上の表を増やすのとか配列の説明が面倒臭かったわけじゃないんだからねっ！)

## 4 おわりに

日本語プログラミング言語「なでしこ」、いかがで

したでしょうか？

経験がない人はプログラミングって、英語がものすごい羅列して、見ても意味がわからない！というイメージがあると思いますが、意外と簡単なことしか書いていないかもしれませんよ。少しハードルが低くなったのではないのでしょうか？

経験者の方。これで人間、やめてみませんか？

さて、ここまで読んでいただいてありがとうございました。日本語プログラミング言語「なでしこ」のインストールは下記 URL を参考にして下さい。またお会いできるのを楽しみにしています。

ナデシコの花言葉は、無邪気、才能、大胆... 等。あなたの才能も花開くかも。

## 5 参考文献

日本語プログラミング言語「なでしこ」公式ページ  
<http://nadesi.com/>

# 自作PCを組んでみた

著者 高速人

## 1 はじめに

みなさん、初めまして。高速人です。初めて部誌を書きます。今回、自作PCを組んでみました。って言っても、もう組んで半年以上経つんですけどね。ということで、自作PCについてダラダラと書いて行きたいと思います。

## 2 構成とか

とりあえず、まずは構成とか書きます。値段とかはめんどいので省略するのです（え

今回組んだ PC の構成

名称	規格
CPU	Corei7-3770K (3.5Ghz TB 時 3.9Ghz)
RAM	DDR3-1333 8GB(4GB × 2)
GPU	GTX650Ti
HDD	Seagate 1TB
マザーボード	GIGABYTE H77-D3H-MVP
電源	玄人志向 KPRW-L4-600W
PC ケース	IN WIN IW-EC002
OS	WindowsServer2008R2Datacenter

・CPU 今回の自作PCを組む上で一番こだわったところかもしれない。CPU だけはこの 3770K を使いたかった！しかし組んで、使っててわかったが、なぜ自分はこんなにこだわったのだろうか。まあでもそこは気にしたらいけない。あと、Ivy-

Bridge だからがダブルグリスバーガーだけど、そこも気にしてはいけない。組んだ時は IvyBridge が最新だったけど部誌を書いているときは Haswell が発売して最新じゃなくなってしまった・・・。

- ・RAM 本当は 16GB にしたかったけど、予算の都合上 8GB になってしまった。マザーボードもあと 2 つメモリ刺せるし、そのうち 16GB にしよう。

- ・GPU 当初は「IntelHD-Graphics4000 でいいんじゃないね？別に普通に使うなら問題ないでしょ」とか思っていたが、マイクラをやるし、いざ組んでみるとグラボを買う余裕があったのでグラボをちゃんと積むことに。ゲームやるのも GTX650Ti で十分？？？まあマイクラの影 MOD をいれても普通にできるから問題ないだろう。

- ・HDD SSD なんてなかった。

- ・マザーボード実はマザーボードがこの PC の欠陥である。なんと、OC できる CPU なのにチップセットが H77、つまり

OC できないのである。あ～まさか Z77 のマザーボードと間違えて H77 のマザーボードなんて買っちゃうとはなあ・・・。ちなみに、夏休みに Z77 のマザーボードに変えるかもしれない。

- ・電源 2012 年冬コミに行った帰りに秋葉原で買った電源

- ・PC ケース一番安いのを頼む

- ・OS 組んだ当初は Windows7Professional だったが、Dreamspark で WindowsSeverR2 をもらったのでそっちに変更。実はもう一つ、某 LinuxOS がインストールされているが、ブートローダーがちゃんと動かないせいで起動しない。UEFI めんどい。

### 3 ベンチマーク

組んだ自作 PC の性能がどれくらいの性能なのか確かめるため、ベンチマークをしてみます。比較用として、自分のサブのノートパソコンのほうのベンチマークも載せておきます。まあスペックの差が



けっこうあるので参考になるかは微妙ですけど…。

比較対象 PC のスペック

名称	規格
PC	eMachines E732-F32C
CPU	Corei3-380M(2.53GHz)
RAM	DDR3-1066 8GB(4GB × 2)
GPU	Intel HD Graphics
HDD	HITACHI 320GB
OS	Windows7Professional 64bit

今回やるベンチマークは、CrystalMarkR2、P-SO2 キャラクタークリエイト体験版 ver2.0、CINEBENCH です。自作PCのほうについては、GTX650Ti でやったときと IntelHDGraphics4000 でやった時のスコアも載せます。

ベンチマークスコア

ベンチマーク		スコア		
		自作 PC(GTX650Ti) (GTX650Ti)	自作 PC(IntelHDGraphics) (IntelHDGraphics)	ノートパソコン
CrystalMarkR2	ALU	61682	63205	32509
	FPU	59275	59655	32646
	MEM	53392	52276	23527
	HDD	20851	20025	7740
	GDI	20791	19763	10357
	D2D	21571	9671	2202
	OpenGL	40019	10244	3548
PSO2	描画設定 1	317860	30517	1451
	描画設定 2	37762	2166	149
	描画設定 3	21272	877	91
CINEBENCH	OpemGL	52.77fps	21.20	1.32
	CPU	7.44pts		2.08
	シングル	1.64pts		0.84

IntelHDGraphics でのスコアと CPU が同じなので割 CPU スコア (CINEBENCH) は GTX650Ti でとったときと CPU が同じなので割愛します。

### 3.1 考察

PSO2 のほうのベンチマークで自作 PC のほうの GTX650Ti と IntelHDGraphics4000 を比べてみても桁がまず違います。初代 IntelHDGraphics からかなり進化して IntelHDGraphics4000 がありますけど、ミドルクラス以上のグラボにはやっぱり勝てないようです。ゲームをするのには向かないのでしょうか。まあ所詮は内蔵グラフィックですからね。

## 4 まとめ

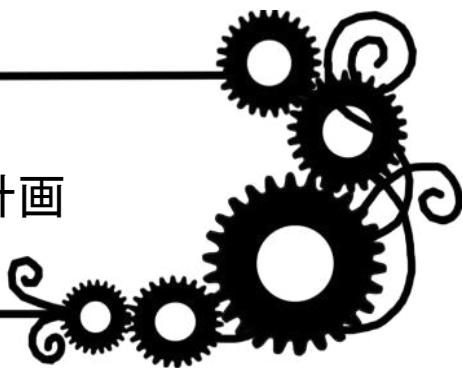
今回はじめて電算部の部誌を書かせてもらいましたが、適当に書いてしまった部分もちよっとあったりしてます……。一応書く前に過去の部誌とか見て参考にさせてもらった部分はあるんですけどね。

ベンチマークは PSO2

の映像を同じやつ 9 回も見ることになってめんどくなったりしました。さすがに 9 回も見ると飽きる……。自分はこれでも前にノートパソコンで PSO2 をやってたところがありましたけど、考えてみればこんなベンチマークのスコアでよく自分は PSO2 ができたなーって思ったりしてました。このノートパソコンの弱点がグラフィック関連ですからね。まあこのノートパソコンはまずゲームをすることを考えて作られてるわけではないからそんなことで文句を言うんだったらちゃんとしたパソコン買えってことですね。

そんなことを思いつつも自作 PC を組む前はマイクラとか PSO2 をこのノートパソコンでやってたわけですけど、自作 PC を組んでからマイクラやったりしたらヌルヌル動いて感動したりしてました。かがくのちからってすげー。

# ミクさん召還計画



著者 でいらん

## 1 はじめに

部誌を手にし、読んでくださっているあなた、まずはこのような拙い文を読んでくださる事に感謝いたします。私はでいらん(19)、何故か群馬高専電算部に所属し始めた、化学系学科の3年生(あっ…)です。ちなみにプログラムは書けません。この、うっかり入ってしまった電算部という環境を利用して何か大きなことをやりたいと野心を抱き始めた所存です。その大きなこととして、タイトルにあります通り、かの有名な電子の歌姫、初音ミクさんと呼び出し、一緒に歌って踊りたいなーと思い、計画を

練っている所です。来年の文発でステージ使ってやりたいね、是非。応援者求む。

## 2 ミクさんを我々の次元へ

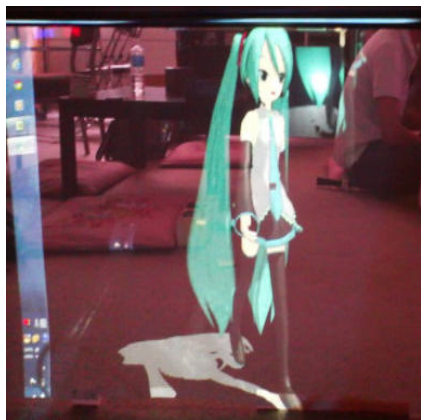
ミクさんの召喚には、透明スクリーンを使います。透明な等身大スクリーンにミクさんを映し、背景として実際のステージ、楽器、演奏者を配置すれば、あたかもミクさんが実際のステージの上にいるように見せることができます。画面の中ではなく、実際のステージの上に「召喚」するので。しかしこの透明スクリーン、製品で買おうと

するとかなりお高い。等身大サイズなんて、とても学生の懐から出せるもんじゃないですね(段々安くなってきてはいるのですが)。投影するためのプロジェクターも必要ですが、これは学校から借りられます。ここで満を喫して登場するのが、「網戸」と「農ポリ」(農業用ポリエチレンフィルム)です！(は?) は? と思っ  
た方は、プロジェクターでグレーの網戸に投射してみてください。世界が変わります。映るんですよ、綺麗に。農ポリは、まだやってませんがとある動画サイトで見ました。綺麗に映ってました。結果として、網戸と農ポリが透明スクリーンとして利用できる訳ですが、お二方の原理はまるで異なります。網戸さんは、暗いところから明るいところは見やすいよねー、っていうのと、光の回折とで、スクリーンとしての機能と透明性を両立しています。スイマセン原理理解ってないです。網戸の色とか、表裏でどうか、色々あ

るっぽいので、これから色々と実験してみたいと思います。一方、農ポリさんは自身が半透明なので、像がうつり、半透明なので透明性がありますよーって感じです。透明度が低い農ポリさんは像がはっきり映る、透明度が高すぎると映り難くなると予想できます。性能としては主に素材が持つ光の波長ごとの透過性なんかが関わってきそうですね。スクリーンについて現段階での私の理解度はこんな感じです。次にモデルとなって頂くミクさんの準備ですが、M-MD(MikuMikuDance)を使うのが最も手軽でしょう。モデルも付属ですし、フリーソフトとは思えないほど、使いやすく、機能性に溢れています。気をつける点は背景を黒にすることくらいでしょうか。あとは視野角の調整ですね。と言いつつ私も触っただけなので偉そうなことは言えないです。むしろ誰か教えてください。

### 3 ミクさん召喚

さてさて、早速モデル用意して、網戸買ってきて、プロジェクター借りてきてやってみました。こんな感じでした。



### 4 おわりに

いやー、何の計画性もなくやってみただけで映るもんなんですねー。

というわけで、今のところはまだまだこんな感じですよ。文化祭の方が忙しいので(副委員長ですし)中々、思うように進められないのが現状ですね。お金も無いし。では、来年の文発、ステージの上で会いましょう！(未確定)



## 部員紹介

著者 部長

### 1 NaGa

知的なオーラを放つ電算部 1,2 を争うイケメン。またそのスタイリッシュ力は53万にもなるという。が、猫耳をノリノリでかぶってくれる。音ゲー筐体の前では手の動きが音速を超えるらしい。

### 2 幻馬

電算部の3枚目俳優。その才能は多彩でピアノ、バイオリン、格闘技、家事等なんでもこなす。こいつにできないことがあるのだろうか。あっ進級か。新歓で新たにタンバリンの才能が明らかになる。

### 3 FreeBEAN

電算部内で部長よりはるかに権力を持った人物。名前からとあるOSを使っていることが予測されるが、その素性は謎に包まれている。というのは世を忍ぶ仮の姿で実は電算部の名誉顧問。

## 4 礫御飯

記事にもある通り部員ではない。多忙でなかなか僕と契約してくれないんだ (CV:加藤英美里)。ソフトウェアもハードウェアもこなしいつの間にか謎 Twitter クライアントが完成していた。詳細は本人へ。

## 5 1\_\_k\_\_

キチガイ (スペック的な意味で) ノートと式寺専コンを持ち歩くイケメン。彼の70%くらいがコーヒーと板チョコで構成されているらしい。わけのわからないハードを組むと思えば、わけのわからない文字列を打ち込んでいたりする。わけがわからないよ (CV:ry)

## 6 ホリャコロヌ

ソシャゲを愛しふわふわとした雰囲気をもつ部員。なんかPN もふわふわしてる。その愛らしさで油断させておいてかなりの秀才。脳みそも柔らかいようだ。

## 7 しーきや

かなりの実力派。DS で様々な謎ゲーを繰り出している。またそのゲームが面白いから困る。電算部の洗礼を受け、最近たまに茶目っ気がうかがえる。堕ちるのも時間の問題か。

## 8 莢

超絶美少女でありイケメンである電算部部长。その権力は山より高く海より深い(?)。リラックマを溺愛している。好きなフォントはTerminalで、これじゃないとプ

プログラミング能力が1/256まで落ちる。(・∀・) イジョ  
ウジサクジエンデシタ

## 9 高速人

電算部でかなりの小型化(1.56m)に成功したロボット。  
名前の通りとても動きが速い。ツンデレ機能を搭載し、  
豆腐を建築するようにプログラムされている。レミリア  
バージョンも存在する。

## 10 でいらん

でいらんお兄さんかつでいらんお姉さん。魔法を使っ  
て歌姫や巫女に変身する。歌姫のクオリティがかなり高  
い。記事のミクさん召喚計画も楽しみである。



イラスト、編集 茨

部誌「D 言語」・2013 年関東信越地区高専文化発表号

---

発行日 平成 25 年度 8 月 26 初版  
発行者 群馬高専電算部  
発行所 群馬工業高等専門学校  
〒371-8530 群馬県前橋市鳥羽町 580

---