Pietのコード自動生成

NoNamea 774

August 17, 2015

1　なぜこんなものを……。

ここのサークルの主がコミケに申し込むというので(初サークル参加)、とりあえず何か出してみたいなぁ(初サークル参加)と思って「じゃあペーパー書いてみます」と言ったのが2月の中旬。これを書き始めたのが8/12。反省している。出ない神ペーパーより出る謎ペーパー、ということで書いています。ブログ[1]のいつもの記事とどう違うんだ、って話はあるけれど、コミケで頒布してみる って事にはそれはそれで意味があるんじゃないのかな、ということで……。というわけで、NoNameA774(Twitter:@nonamea774)です。よろしくお願いします。こんな文章を読んでくださってありがとうございます。

2　Piet

2.1　Pietとは

Pietというお絵かきプログラミング言語があります。Wikipediaによると(一部略) [1]、

> Pietは、難解プログラミング言語のひとつである。いくつかの難解プログラミング言語を開発した David Morgan-Mar がピエト・モンドリアンの作品に影響を受けて考案した言語で、文字ではなく色を組み合わせて記述する。ソースコードを一見すると、抽象画のように見える。

とまぁそんな感じです。大変申し訳無いのですが、紙面の都合上、Pietについては各自調べてください。非常に簡単に表面的な説明をすると、ドット絵で書かれたソースコードの上をプログラムカウンタが縦横無尽に動き、移動の際の色の変化によって定められた命令を実行するプログラミング言語です。うちのサークル[2]の部員が書いたおそらく現時点で日本語資料の中で一番わかりやすいPietについての資料をご覧いただくと便利です。「Pietのエディタを作った話[3]」on SlideShare.

2.2　テストツールを書く。

こんなスライドが書かれるぐらいなので、うちのサークルでPietを書く(描く?)のが流行っています(わたしは手では全然書いていませんが……)。部員が手でPietを書いて手でそれが正しく動いているのか適当に入力を与えてテストしているのを見て、せめてテストぐらいは楽にできたらいいな、という考えから、テストツールを書きました[4] [5]。これがすごくバギーで、全然他の人に使われていないのだけれど、これのお陰で自動生成がだいぶ助かっている……。

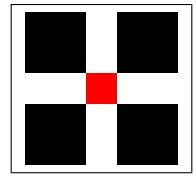
2.3　Pietを生成する。

2.3.1　方針

Pietの言語仕様を見てみれば[6]、こんなのはまともに手で書いていられないことがすぐにわかります。まず実行される命令列を擬似言語のような形で書いて、それを頑張って画像に落としていくようにみんなすぐになります[要出典]。それの様子を見て気づきます。「これはこの擬似言語っぽいところから自動生成できるのでは?」実際、条件分岐等の起こらないPiet[7]なら、幅を1Codelにして一直線に書いていけばすぐに書くことができます。条件分岐の無いものに関しては、順番に命令が実行されるように適当に置いていけば簡単に自動生成できそうです。でも一直線に命令を置いていくだけでは、構造化定理の「順次」しかない状況、チューリング完全とはなりません。チューリング完全にするためには‘switch’とか‘pointer’とかのよくわからない命令を実行する必要がありそうです。しかしただまっすぐ自動で置いていくだけだと、どのように分岐した後合流するかを考えたりするのが大変だったり、そもそも置いて行く時に交差が発生してしまって変な命令が実行されたりしてしまわないかを管理したりしないといけないので大変です。

2.3.2　部品化

ここで役立ったのが、Pietの仕様の白色のマスの動きです。適当に要約するとこんな感じです。「現在の向きに沿ってそのまま滑って行って、次のマスへと行く。そしてその際、なんの命令も実行されず、移動だけが起こる。」これを利用して、7x7[8]のサイズの部品を命令ごとに作り、それを組み合わせてプログラムを構成するようにした。それによって、置いていく際の大きさの予測が簡単になり、また、これが大事なのですが、交差を実装することができるようになった。以下のようにすれば、交差が作れるのは明らかであろうと思います [9] [10] (5x5で書いています)。



ここにIfとGotoが入れば、チューリング完全となります。Ifについては、‘not’、‘pointer’と連続して実行することによって、スタックの先頭の値が0でないなら何もせず、0ならば進む向きを時計回りに90度回す事ができます。それによって、以下の様なものを作ることによって、Jump Equal Zeroのようなものが作れるので、それを使います [11]。

[1] https://nna774.net/blog/

[2] この場合のサークルは大学のサークルのほう。KMC、京大マイコンクラブです。以下出てくるサークルも全部こっち。

[3] http://www.slideshare.net/KMC_JP/piet-46068527. ここに出てくるエディタは現在クローズドで、部員にしか配られていない。需要があればビルド済みのものは公開してもいい とは言っているが、できればOSSにして欲しい……。世界に対する損失だ。

[4] https://github.com/nna774/piet-testutils

[5] というのを適当に書いたのがブログのhttps://nna774.net/blog/2015/07/08/piet.htmlの記事。

[6] Pietの言語仕様をよく理解していただけないとこの先何言ってるかわからなくなりそうです。全く説明していないのに要求してごめんなさい。「Pietのエディタを作った話」をよく読んで下さい……。

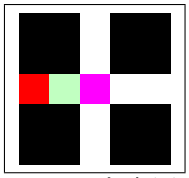
[7] Hello, World! とかの定数出力を行うものとか。

[8] 最初は5x5だったけど、いくつかの命令を安全に実装できなさそうだったので大きくした

[9] Pietの仕様に十分詳しければ。ごめんなさいごめんなさい。

[10] piet/cross.png

[11] piet/notbranch.png



これができれば、あとはgotoですが、うまい感じに繋いでやって、プログラムポインタを誘導するだけでできます。以下の様なパーツたちを作ればできます。ファイル名は、レポジトリでの同じ意味で使われてる画像です。

- piet/nop_h.png 左から来たのをそのまま右へ(or 逆)。
- piet/nop_v.png 上から来たのをそのまま下へ(or 逆)。
- piet/curve5.png 上から入り回転して左に出る。

- piet/curve7.png 左から入り回転して上に出る。

- 他にも同じような回転がある。
- piet/join.png 左、下からを合流して右へ。
- piet/rjoin.png 右、下からを合流して上へ。
- piet/ljoin.png 左、下からを合流上へ。

2.3.3 組み立て

というわけで、「順次」、「反復」、「分岐」の3つが揃い、これで任意のプログラミングが書けそうな感じがしてきました。あとはこれを実装するだけなので、パーツをPidet(「Pietのエディタを作った話」スライドで作られていたエディタ)で作り、そのパーツをうまいことjsのCanvasにぺたぺた貼り付けてコードを組み立てます。この、"うまいこと"というのがすこし難しいのですが、現時点では、ジャンプ系の命令がひとつ出る度に、一列の通路を上記のnop_h(horizontal nop)で作ってやって、JumpEqualEeroやcurveシリーズ、joinを使ってそこに誘導していくような感じで生成しています。最初からn番目のジャンプ系命令で使う通路を、上からn番目であると決め打ちしてしまうことで、通路が重なったりすることを考えたりしなくて済むので、必要以上に縦長になってしまう可能性はあれど、生成は楽になります。https://github.com/nna774/piet-automata/blob/07d387/app.js#L272-L366のあたりです。一枚例を入れておきます。

```
INN
PUSH 1
DUP
ADD
MOD
JEZ zero
PUSH 1
OUTN
HALT
LABEL zero
PUSH 1
NOT
OUTN
HALT
```

こんな感じのコードが生成されます。

二本横向きの線が見えると思いますが、上がメインの命令列で、下のほうが、ジャンプの時の通路です。うーんわかりづらい……。

現状ある程度いいのが生成されますが、人間には明らかな変形で小さくできるような時もあったりするような感じです[12]。

[12]例を載せるのムズイから、https://twitter.com/nonamea774/status/622858091365711872とかを見てください。セルフリプライで解説してるやつです。

3 あとがきの国

3.1 Piet

とまぁだいたいこんな感じで生成できるようなものを作っています。レポジトリはnna774/piet-automata/にあります。今レポジトリのREADMEに書かれている命令セット(最近サークルではPasって呼ばれている。Pascalとかぶってることに最近気づいた) でも、かなり楽になってるとはいえ、Ifをそのまま書いたりできないし、だいぶつらいのでこれに落とす上位言語を書きたいなぁと最近は思っています。PUSHの数字の生成とか結構雑だし、いい感じのPullRequestをぜひ待っています!! 何か進展があったりしたら、https://nna774.net/piet/にどんどん書いて行こうと今思いました。まだページは404なので、帰ったら作ります。

3.2 さいごに

まぁなんか適当に書いてきましたが、そろそろ印刷してもらわないといよいよヤバいので、とりあえずこのへんで一旦終わりとしようと思います(現在2015/08/15 2:53 JST)。こんな文章でしたがここまで読んでくださったことを非常に感謝します。まともな文章を書くのは苦手だけれど、何か書くのは結構好きなので、もっとなんかだしてけたらなぁ という感じです。冬コミの申し込みセットを勢い良く明日買うとかしてみる? 何はともあれ、また(Twitter以外で)活字で会えることを祈っています。それでは。

References

[1] Wikipedia. Piet — Wikipedia. [Online; accessed 12-August-2015]. 2013. URL: https://ja.wikipedia.org/w/index.php?title=Piet&oldid=46762504.

最後に色々書いて埋めようとしたものの余ってしまったこのスペースを埋める雷(せっかくカラーだし)。

奥付

2015/8/15	初版発行
hash:	cc67d54e1a8de9dc1e7c6(の次)
著作・発行	NoNameA 774 (nonamea774@nnn77)
メールアドレス	nonamea774@gmail.com
Web	https://nna774.net/
Twitter	@nonamea774
GPG Key	0x0C3E3AB2
fingerprint	674A 287A 21D2 2431 AD8F D328 AEF3 C3C7 0C3E 3AB2