

上田市マルチメディア情報センター

「春休みプログラムセミナー」

NINTENDO 3DSでゲームプログラムを作ろう！

2016年3月26日

講師：(株)スマイルブーム 小林貴樹

Version 1.0

■ 本日のスケジュール

10:00 プチコンの基本をおぼえよう（ちょっとだけ他の人の作品を動画で見よう）

11:00 「ゾンビハンター」ゲームのプログラムはじまり

12:00 <お昼休み>

13:00 「ゾンビハンター」ゲームのプログラム続き

14:00 眠くなってきたら「おえかきタイム」？

15:00 「ゾンビハンター」好き勝手に調整してみよう

16:00 プログラマーになるためには？（質問コーナー）

17:00 おしまい

だいたいこんな感じで進めます。

途中で分からない人が増えてしまったときは少し説明の時間を増やします。

いつでも分からなくなった時は声をかけてください。

TOPMENU (トップメニュー)

キーボードの使い方

はじめての **プチコン3号** SMILE BASIC



アイコンを
タッチ!

TOPMENU (トップメニュー) の機能

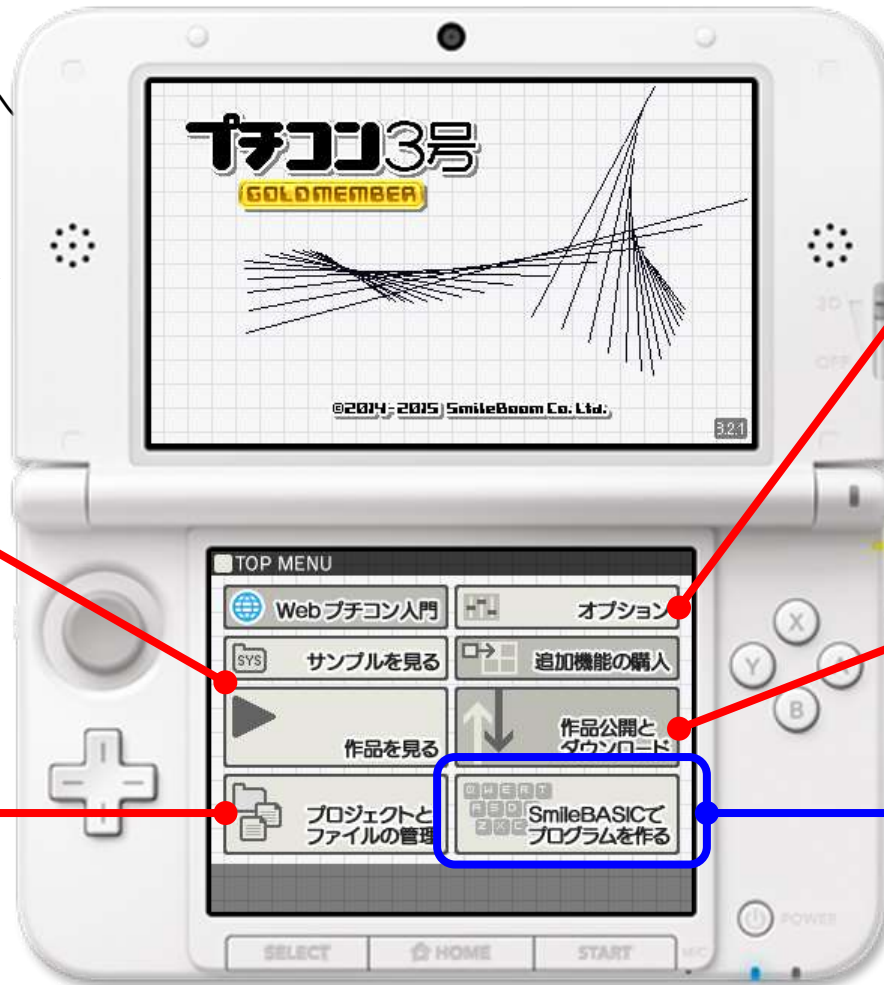
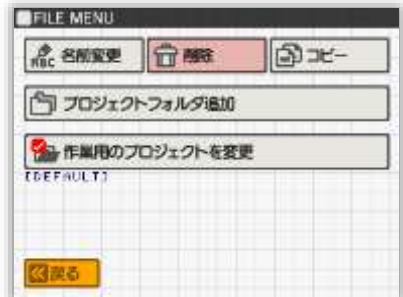
- 起動すると必ず最初に表示される画面 (やりたい作業にあわせてボタンをタッチ)

- Webプチコン入門
 - 公式初心者ページへ
- 追加機能の購入
 - 拡張サービスの購入

サンプルや作品を見る



ファイルの管理機能



色や設定の調整



公開とサーバー管理機能

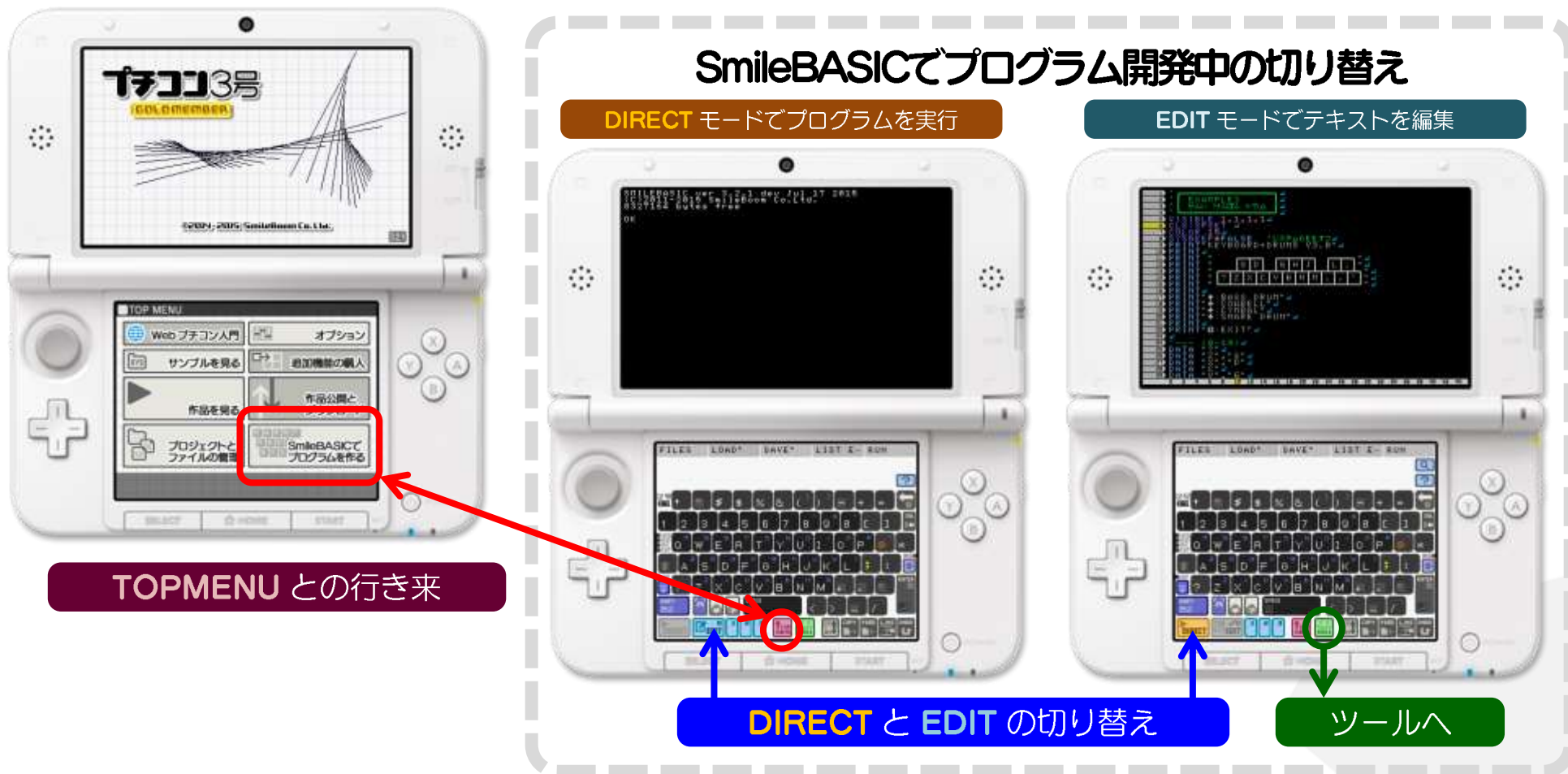


SmileBASICでプログラム作成



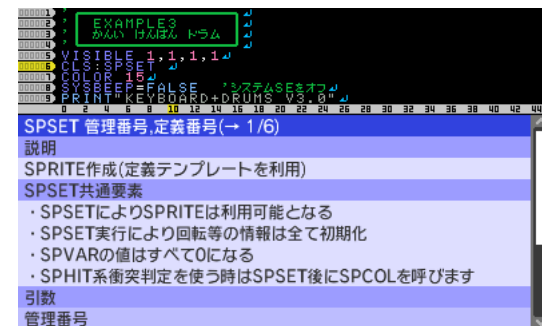
プログラム開発中の画面の切り替え

- プログラム作成中は、**TOPMENU** と SmileBASIC の間はそれほど切り替えない
- SmileBASIC 内での **DIRECT** と **EDIT** の切り替えは良く使う



キーボードについて

- SmileBASICの作業中は下画面にキーボードが表示されます
 - DIRECTモードとEDITモードで共通部分



命令候補エリア

- 通常は命令候補を表示
- #で始まる定数の確認
- 長い命令の人力回避
- 文字を増やし候補を絞る

ファンクションキー

- KEY命令で再定義可能なボタン

●CAP

- SHIFTキーを押したままの状態にする

文字種類の変更

- 英数字
- 記号 (欧州文字)
- ひらがな/カタカナ

スマイルボタン

- ツールを実行する

ヘルプ表示モード切替



Yボタン

- 1文字削除(BSキー相当)

Aボタン


- 改行(エンターキー相当)

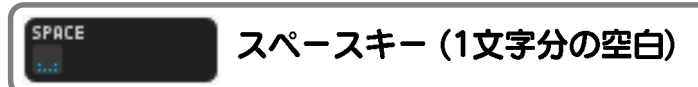
INS

- 挿入と上書きの切り替え

エンターキー (改行)



本資料内で、 という記号がある場合は、エンターキーを人力すると解釈してください。



その他のキーとステータス表示部品

- 主にテキスト入力時に利用します（SHIFTを押すと変化するボタンがあります）

ファンクションキー（KEY命令で定義可能なボタン）

FILES LOAD" SAVE" LIST E... RUN

SHIFTを押すと、SLOTのSAVEとLOADボタンとなる

SAVE LOAD *EX3KEYDRUM



バックスペースキー（カーソルの前を1文字消す）



SHIFTを押しながら押すと1行削除



デリートキー（カーソル位置の文字消す）



SHIFTを押しながら押すとカーソルから行末まで削除

DIRECTモードとEDITモードの切り替え（SLOT変更）



- COM（通信中）
- MOT（センサー利用中）
- MIC（マイク利用中）
- EXP（拡張スライドパッド利用中）



範囲選択（カーソルのある位置を始点とする）



コピー（選択範囲を取り込む、範囲が無い時1行）



ペースト（取り込まれた内容を貼り付ける）



1行挿入（カーソルの下の行に1行追加する）



やり直し（直前の操作を元に戻す）



カット（選択範囲を切り取る、範囲が無い時1行）



ペースト（取り込まれた内容を貼り付ける）



1行削除（カーソル行を1行削除する）



取り消し（UNDOを取り消す）

※CUT以降は、SHIFTキーを押した時に切り替わります



TOPMENUへ戻る










検索と置換モードへ

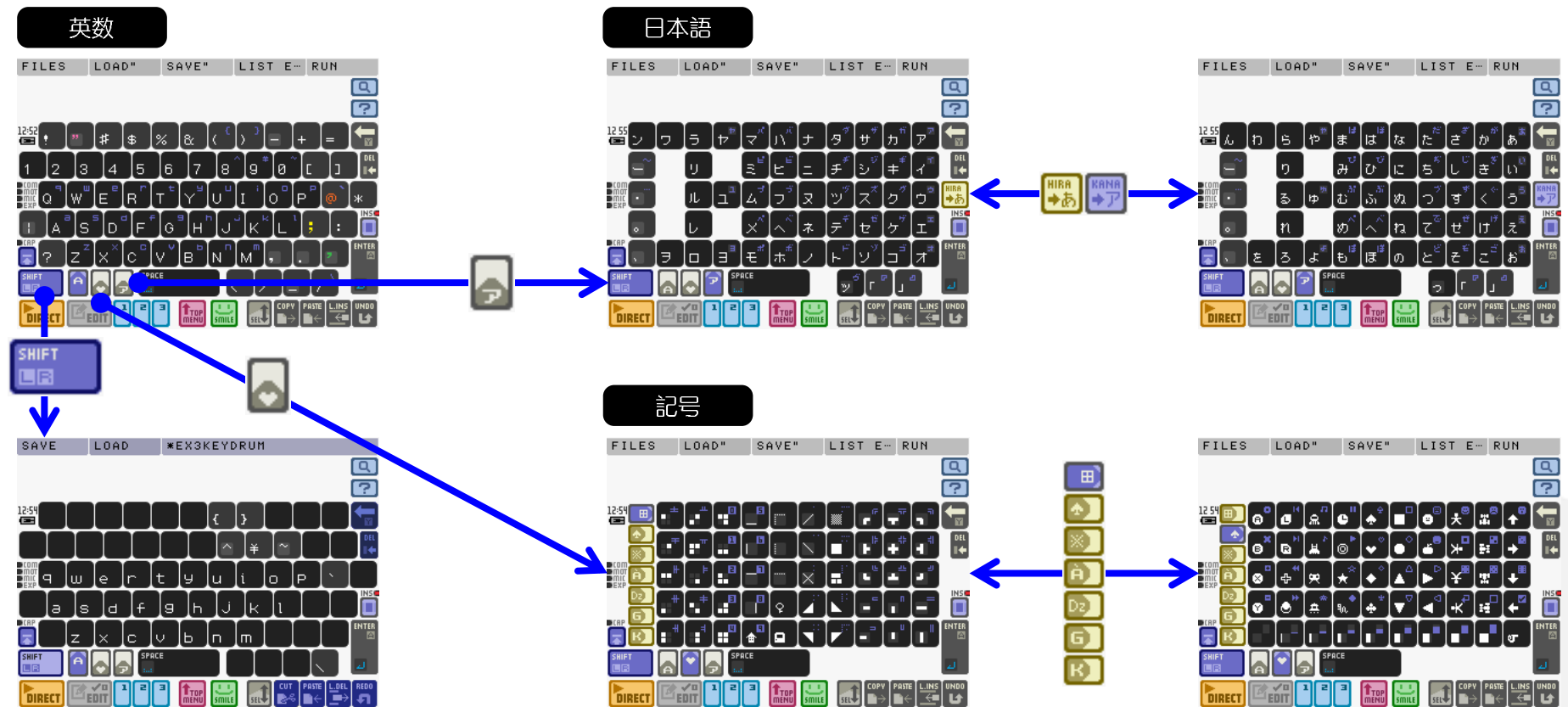
12:54



時計と電池残量

キーボードの種類を切り替える

- キーボードは、 英数・ 記号・ 日本語の切り替えが可能です
- 記号は、さらに6種類()の切り替えが可能です
- 日本語は、 平仮名・ 片仮名の切り替えが可能です
- 右上に紫色の文字が書かれているキーは、 SHIFTで別の文字に切り替わります



TOPMENUからSmileBASICに入ってください
DIRECTモードでコンピュータに直接指示
ダイレクト
DIRECTモードについて

ボタンを
タッチ!



DIRECT (ダイレクト) モードとは？

- 3DSの中に入っているコンピュータに直接指示を出すモードです
- 命令を1行分入力して実行することができます

DIRECT モードでプログラムを実行する場合の表記 (1行単位)

```
ACLS ↵
```

スティック

- 左右で行頭行末移動

十字ボタン

- 上下で入力履歴の再利用
- 左右でカーソル移動



画面に1+3の答えを表示する「 PRINT 」

- 単純な計算結果を表示させます

```
PRINT 1+3 ↵
```

```
4  
OK
```

まるで電卓のように計算結果を表示します（数学関数も搭載しているので関数電卓代わりにも使えます）

- ひき算 `PRINT 5-2 ↵`
- かけ算 `PRINT 2*4 ↵`
- わり算 `PRINT 6/3 ↵`
- わった余り `PRINT 6 MOD 5 ↵`
- 複雑な計算 `PRINT (1+3)*5-2/4 ↵`
- 小数を含む計算 `PRINT 1.25*2.5+5.77 ↵`
- 四捨五入 `PRINT ROUND(0.25*63.99)+15 ↵`
- 0~9までの乱数 `PRINT RND(10) ↵`

```
? 1+3 ↵
```

PRINT は、? (クエスチョン・はてな)記号で省略することが可能です

画面に線や円を描く

- 順番に実行すると画面に線や円が描画されます

```
ACLS ↵
```

まず、表示関係を初期状態に戻します（画面の文字などが消えます）

```
GLINE 0,0,399,239,RGB(255,0,0) ↵
```

左上から右下へ赤い線を引きます

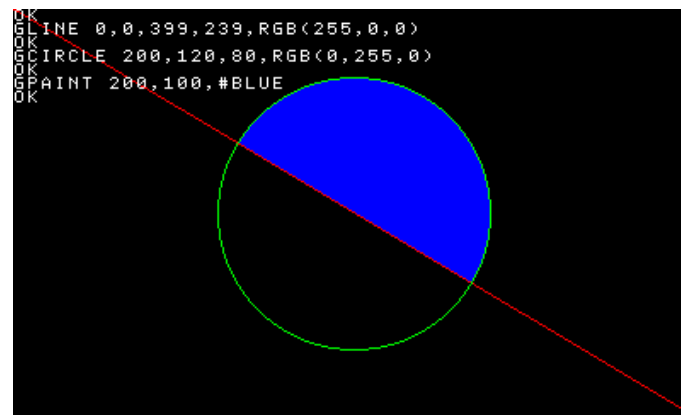
```
GCIRCLE 200,120,80,RGB(0,255,0) ↵
```

画面中央に半径80ドットの緑の円を描きます

```
GPAINT 200,100,#BLUE ↵
```

最後に画面中央より少し上から青で塗ります

#BLUEは、定数と呼ばれるシステムが予約した名前です



音を出す（効果音・BGM・声）

- どんな効果音だったか忘れたときに簡単に確認できます（134種類）

BEEP 8 ↓

- 標準搭載されているBGMについても確認できます（43種類）


BGMPLAY 41 ↓

- BGM演奏を止めたい時は以下のように入力して下さい

BGMSTOP ↓

- 音声合成によって会話を喋らせることもできます

TALK "オハヨウゴザイマス" ↓

- プチコン3号に搭載されている音声合成は日本語しか喋れません
- ひらがなカタカナの入力は、キーボードの  を押してキーの種類を切り替えてください

作ったプログラムを SAVE 命令で保存する

- 3DSの電池が切れるとせっかく作ったプログラムは消えてしまいます
- プログラムを書いたらこまめに保存する癖をつけましょう
 - 何か大きな変更を行う前には保存しておくのと元に戻るのが楽です
 - 大きな変更を行う前にはバックアップとして別の名前を付けて保存すると安心です
- ファイル名として使える文字はアルファベットの大文字A～Zと数字および _ です

```
SAVE" SAMPLE" ↵
```

- SLOTごとのSAVE専用の機能（SHIFTを押しながらファンクションキー）

SAVE LOAD *EX3KEYDRUM

- SAVE用のダイアログが開きファイル名を入力することができます
- DIRECTにSAVEコマンドを入力するよりも楽に保存できます
- 現在作業対象となっているSLOTの内容が保存されます



保存されたプログラムを LOAD 命令で読み込む

- 前回の続きからプログラムを作る場合はSAVEで保存されたファイルを読み込みます
- 作業中のプログラムがある状態でLOADすると作業中のものは消えてしまいます
 - 作業中のプログラムを後で利用する可能性がある場合は名前を付けてSAVEしておきましょう

```
LOAD" SAMPLE" ↵
```

- SLOTごとのLOAD専用の機能（SHIFTを押しながらファンクションキー）

SAVE

LOAD

*EX3KEYDRUM

- LOAD用のダイアログが開きファイルを選択することができます
- 現在の作業SLOTに読み込むファイルを選択し「決定」を押します
- 現在読み込まれているファイル名は **青い枠部分** に表示



まとめて命令を書くためのテキスト入力画面
エディット
EDITモードについて

ボタンを
タッチ！



EDIT (エディット) モードとは？

- コンピュータへ指示する内容をテキストとして登録するモードです
- 命令をたくさん並べて書くことで連続して指示を出すことができます
 - 編集用の行番号は実行時には意味がありません
- プログラム実行は **DIRECT** モードから
 - **RUN** ↓

EDIT モードでテキストを編集する場合の表記 (複数行)

```
000001 ACLS ↓  
000002 PRINT 1+2 ↓
```

実行するためには **DIRECT** モードで RUN (またはファンクション5)

```
RUN ↓
```

スティック

- 上下で先頭と最後尾
- 左右で行頭行末移動

十字ボタン

- 上下左右でカーソル移動

SHIFTを押しながら操作するとスティックと同じ動作となる

検索と置換


- テキスト内から文字を探すときや置き換えるときに利用します

編集画面

- カーソルのある位置に文字が入力されます



新しいプログラムを作る準備

- 新しいプログラムを入力する前に今まで入力したプログラムをすべて消します
-  DIRECTボタンを押して実行モードに切り替えます
- キーボードから「NEW」と入力してENTERを押して下さい

NEW ↵

- 編集作業対象が **SLOT0** になっていることを確認してください
 - プチコンには上級者向けに作業用SLOTが4個用意されていますが今回はSLOT1～3は使いません

このボタンが押されて
いれはOKです！




エラーが出てしまった人へ

- RUNした結果、エラーが出てしまった！！

```
Syntax error in 0:##
```

※##の部分はエラーが出ている行番号が表示されます

- ##行で書式に間違いがあるようです
 -  EDITモードに切り替えて、編集画面から該当行を見てみましょう
 - キーボードの上の方にあるファンクションボタンの右から2番目を押すとエラーのある行に飛んでくれます



- その行の中に必ずエラーがあります。元になっているプログラムと良く見比べてみてください
- その他のエラーが出た場合も、まずは編集画面でエラーが出ている行を見てみましょう
- 必ず何か入力間違いがあるはずです
 - 似ている記号 ' アポストロフィーと , カンマや、ゼロとオーなどが良く間違えます
 - 緑色のコメント（注釈）としてメモが書かれている行はアポストロフィーで始まっています

プログラムを作るという仕事は、

「プログラムを書いて」→「実行」→「エラーがあれば修正」→「再び実行」・・・

という作業を正しい動きになるまで繰り返す作業となります。



長いプログラムを入力する前の準備

**お待たせしました
ゾンビハンターの開発を始めましょう！**

準備

- 今回のゲームは大量発生するゾンビを倒すシンプルなシューティングゲームです！
- 画像を使ったプログラムなので最初に公開キーを1つ読み込みます

公開キー「DXEEQ3NJ」

- 読み込むと「UEDA2016」と言うプロジェクトが追加されます
 - カレントプロジェクトを変更しましょう
 - TOPMENUの「プロジェクトとファイルの管理」ボタンをタッチ
 - 「作業用のプロジェクトを変更」ボタンをタッチ
 - 「UEDA2016」を選択して「決定」ボタンをタッチ
 - TOPMENUから「SmileBASICでプログラムを作る」をタッチ
 - コンソール画面に、[UEDA2016]OK と出ていれば正しく設定されています
- BASICに入ったら、コンソール画面からLOAD"MON" (enter) と入力します
 - LOAD "MON" (enter)
 - これでプログラムテンプレートが読み込まれます
 - この先はこのプログラムに追加してゲームに仕上げていきます



MON_S1、MON_S2

ゲームのループ以外の処理を組み込む

①~③タイトル表示、その後④でゲームオーバー

①

```

0001  VAR I,X,Y,DY=100,HI=2525
0002  ACLS:BGMSTOP:XSCREEN 2
0003
0004  --- スプライト設定 ---
0005  TL LOAD "GRP4:SP",FALSE
0006  I=0 TO 15
0007  X=(I MOD 4)*128
0008  Y=(I DIV 4)*128
0009  SPDEF I,X,Y,128,128,64,127,&H01
0010  SPCLIP 0,16,399,239
0011  --- タイトル ---
0012  @TITLE
0013  SETUP "TIT",TRUE
0014  --- ゲーム ---
0015
0016  --- そうさばんしょきか ---
0017
0018  ---
0019
0020  --- メインループ ---
0021
0022  --- ゲームオーバー ---
0023
0024
0025
0026
0027
0028
0029
0030
0031
0032
0033
0034
0035
0000  0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44
    
```

②

```

0038  -----
0039  ; 名前しゅんぴ
0040  -----
0041  DEF SETUP NM$,BSW
0042  VAR P,T=30*1
0043  FOR P=1 TO 0 STEP -1
0044  DISPLAY P
0045  FADE 0
0046  FADE #WHITE,T
0047  NEXT
0048  WAIT T
0049
0050  IF NM$!="" THEN
0051  LOAD "GRP0:"+NM$,FALSE
0052  DISPLAY 1
0053  GOCOPY 0,0,256,319,495,0,0,TRUE
0054  DISPLAY 0
0055  ENDEF
0056
0057  FOR P=1 TO 0 STEP -1
0058  DISPLAY P:SPCLR
0059  FADE #WHITE,T
0060  NEXT
0061
0062  IF BSW==FALSE THEN RETURN
0063  WHILE 1
0064  IF BUTTON(2) THEN BREAK
0065
0066
0000  0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44
    
```

③

```

0045  -----
0046  ; #WHITE,T
0047  -----
0048  NEXT
0049  WAIT T
0050
0051  IF NM$!="" THEN
0052  LOAD "GRP0:"+NM$,FALSE
0053  DISPLAY 1
0054  GOCOPY 0,0,256,319,495,0,0,TRUE
0055  DISPLAY 0
0056  ENDEF
0057
0058  FOR P=1 TO 0 STEP -1
0059  DISPLAY P:SPCLR
0060  FADE #WHITE,T
0061  NEXT
0062
0063  IF BSW==FALSE THEN RETURN
0064  WHILE 1
0065  IF BUTTON(2) THEN BREAK
0066
0067  ZDY
0068  ZDY
0069  ZDY
0070
0071  -----
0072  ; たいりょくげしゅ
0073  -----
0000  0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44
    
```

RUN

④

```

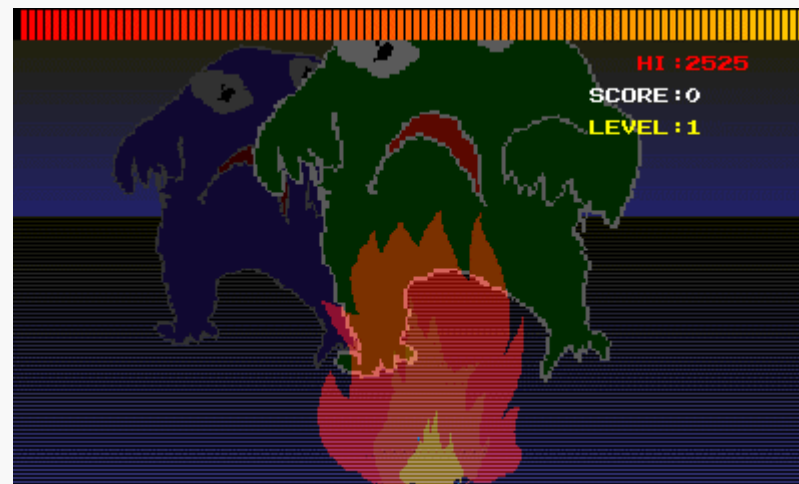
0016  SPCLIP 0,16,399,239
0017  --- タイトル ---
0018  @TITLE
0019  SETUP "TIT",TRUE
0020  --- ゲーム ---
0021
0022  --- そうさばんしょきか ---
0023
0024  ---
0025
0026  ---
0027
0028  ---
0029
0030  --- メインループ ---
0031
0032
0033  --- ゲームオーバー ---
0034  VAR T,DY=100
0035  X=0,Y=100
0036  GOV 400,400
0037  TITUP "GOV",TRUE
0038  BGMSTOP
0039  @TITLE
0040
0041  -----
0042  ; 名前しゅんぴ
0043  -----
0044  DEF SETUP NM$,BSW
0000  0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44
    
```

RUN

名前を変えてSAVEしておこう

MON_S3、MON_S4

ゲームのループを組み込む



①～③で体カゲージとスコアの表示

①

```
19 @TITLE
20
21 SETUP "TIT", TRUE
22
23 SETUP "ゲーム", FALSE
24
25
26
27
28
29
30
31
32
33 STX, STY, BTN
34
35
36
37
38
39
40
41
42
43
44
45
46
47
```

②

```
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
```

③

```
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
```

名前を変えてSAVEしておこう

RUN

MON_S5、MON_S6、MON_S7

プレイヤーの攻撃を組み込む



①~④でプレイヤーの攻撃（下画面で撃って上画面に炎）

①

```
122 :マSPRITEプログラム
123 :
124 :
125 :ATTACK
126 :P=CALLIDX
127 :
128 :OF SP OUT X,Y
129 :
130 :
131 :
132 :
133 :
134 :
135 :
136 :
137 :
138 :
139 :
140 :
141 :
142 :
143 :
144 :
145 :
146 :
147 :
148 :
149 :
150 :
151 :
152 :
153 :
154 :
155 :
156 :
157 :
158 :
159 :マSPRITEプログラム
```

②

```
150 :マSPRITEプログラム
151 :
152 :
153 :HOT
154 :P=CALLIDX
155 :
156 :OF SP OUT X,Y
157 :
158 :IF X>-16 && X<320+16 && Y>-16 THEN
159 :PCHK(SP) AND #CHKXY THEN RETURN
160 :
161 :
162 :
163 :
164 :
165 :
166 :
167 :
168 :
169 :
170 :
171 :
172 :
173 :
174 :
175 :
176 :
177 :
178 :
179 :
180 :
181 :
182 :
183 :
184 :
185 :
186 :
```

③

```
186 :マSPRITEプログラム
187 :
188 :
189 :
190 :
191 :
192 :
193 :
194 :
195 :
196 :
197 :
198 :
199 :
200 :
201 :
202 :
203 :
204 :
205 :
206 :
207 :
208 :
209 :
210 :
211 :
212 :
213 :
214 :
```

④

```
215 :
216 :
217 :
218 :
219 :
220 :
221 :
222 :
223 :
224 :
225 :
226 :
227 :
228 :
229 :
230 :
231 :
232 :
233 :
234 :
235 :
236 :
237 :
238 :
239 :
240 :
241 :
242 :
243 :
```

RUN
名前を変えてSAVEしておこう



MON_S8、MON_S9、MON_S10、MON_S11

敵の動きを組み込む

①敵用スプライトの処理、②~③モンスター生成

①

```
245 | モンスターSPRITEプログラム
246 | -----
247 | DEF FUNCMON
248 | VAR SP=CALLIDX
249 |
250 | IF SPCHK(SP) AND #CHKXY THEN RETURN
251 |
252 | VAR I=SPVAR(SP,0)
253 | IF I THEN
254 |   RND(0,0,0,0)
255 |   RND(0,0,0,0)
256 |   RND(0,0,0,0)
257 |   RND(0,0,0,0)
258 |   RND(0,0,0,0)
259 |   RND(0,0,0,0)
260 |   RND(0,0,0,0)
261 |   RND(0,0,0,0)
262 |   RND(0,0,0,0)
263 |   RND(0,0,0,0)
264 |   RND(0,0,0,0)
265 |   RND(0,0,0,0)
266 |   RND(0,0,0,0)
267 |   RND(0,0,0,0)
268 |   RND(0,0,0,0)
269 |   RND(0,0,0,0)
270 |   RND(0,0,0,0)
271 |   RND(0,0,0,0)
272 |   RND(0,0,0,0)
273 |   RND(0,0,0,0)
```

②

```
275 | モンスターはせいしゅうきのかんり
276 |
277 | HOKO
278 | BORN
279 | IIT
280 | WT
281 | IIT<=0 THEN
282 |   RND(WTMAX)+WTSUB
283 |   I=0 TO WTGRP
284 |   IIT
285 |   ZZZZ
286 |   COMACT
287 |   COMACT
288 |   COMACT
289 |   COMACT
290 |   COMACT
291 |   COMACT
292 |   COMACT
293 |   COMACT
294 |   COMACT
295 |   COMACT
296 |   COMACT
297 |   COMACT
298 |   COMACT
299 |   COMACT
300 |   COMACT
301 |   COMACT
302 |   COMACT
303 |   COMACT
```

③

```
289 | モンスターはせいしゅうきのかんり
290 |
291 | HOKO
292 | BORN
293 | IIT
294 | WT
295 | IIT<=0 THEN
296 |   RND(WTMAX)+WTSUB
297 |   I=0 TO WTGRP
298 |   IIT
299 |   ZZZZ
300 |   COMACT
301 |   COMACT
302 |   COMACT
303 |   COMACT
304 |   COMACT
305 |   COMACT
306 |   COMACT
307 |   COMACT
308 |   COMACT
309 |   COMACT
310 |   COMACT
311 |   COMACT
312 |   COMACT
313 |   COMACT
314 |   COMACT
315 |   COMACT
316 |   COMACT
317 |   COMACT
```

名前を変えてSAVEしておこう

RUN

①時間で新しい敵出現、②種類を決める、③モンスター0番

①

```
215     モンスターはせいしゅうきのかんり
216
217 DEF BORN
218   DEF I
219   CURT
220   WT
221   IF WT<=0 THEN
222     WT=RND(WTMAX)+WTSUB
223     FOR I=0 TO WTGRP
224       SETMON
225     NEXT I
226   ENDIF
227 END
228
229     モンスター1ピキラまれる(#100-199)
```

②

```
289     モンスター1ピキラまれる(#100-199)
290
291 DEF SETMON
292
293   VAR T=RND(4)
294   DEF DT=8+T
295   P
296   SET P=100,199,DF OUT SP
297   IF P<1 THEN RETURN
298   VAR P
299   IF P<10 THEN MON0
300   IF P<20 THEN MON1
301   IF P<30 THEN MON2
302   IF P<40 THEN MON3
303   IF P<50 THEN MON4
304   IF P<60 THEN MON5
305   IF P<70 THEN MON6
306   IF P<80 THEN MON7
307   IF P<90 THEN MON8
308   IF P<100 THEN MON9
309   ENDIF
310   CALL SP,"FUNCMON"
311
312
313
```

③

```
311
312
313 DEF MON0 SP
314
315   VAR T=RND(4)
316   DEF DT=8+T
317   P
318   SET P=100,199,DF OUT SP
319   IF P<1 THEN RETURN
320   VAR P
321   IF P<10 THEN MON0
322   IF P<20 THEN MON1
323   IF P<30 THEN MON2
324   IF P<40 THEN MON3
325   IF P<50 THEN MON4
326   IF P<60 THEN MON5
327   IF P<70 THEN MON6
328   IF P<80 THEN MON7
329   IF P<90 THEN MON8
330   IF P<100 THEN MON9
331   ENDIF
332   CALL SP,"FUNCMON"
333
334
335 DEF MON1 SP
336
337   VAR T=RND(4)
338   DEF DT=8+T
339   P
340   SET P=100,199,DF OUT SP
341   IF P<1 THEN RETURN
342   VAR P
343   IF P<10 THEN MON0
344   IF P<20 THEN MON1
345   IF P<30 THEN MON2
346   IF P<40 THEN MON3
347   IF P<50 THEN MON4
348   IF P<60 THEN MON5
349   IF P<70 THEN MON6
350   IF P<80 THEN MON7
351   IF P<90 THEN MON8
352   IF P<100 THEN MON9
353   ENDIF
354   CALL SP,"FUNCMON"
355
356
357 DEF MON2 SP
358
359   VAR T=RND(4)
360   DEF DT=8+T
361   P
362   SET P=100,199,DF OUT SP
363   IF P<1 THEN RETURN
364   VAR P
365   IF P<10 THEN MON0
366   IF P<20 THEN MON1
367   IF P<30 THEN MON2
368   IF P<40 THEN MON3
369   IF P<50 THEN MON4
370   IF P<60 THEN MON5
371   IF P<70 THEN MON6
372   IF P<80 THEN MON7
373   IF P<90 THEN MON8
374   IF P<100 THEN MON9
375   ENDIF
376   CALL SP,"FUNCMON"
377
378
379 DEF MON3 SP
380
381   VAR T=RND(4)
382   DEF DT=8+T
383   P
384   SET P=100,199,DF OUT SP
385   IF P<1 THEN RETURN
386   VAR P
387   IF P<10 THEN MON0
388   IF P<20 THEN MON1
389   IF P<30 THEN MON2
390   IF P<40 THEN MON3
391   IF P<50 THEN MON4
392   IF P<60 THEN MON5
393   IF P<70 THEN MON6
394   IF P<80 THEN MON7
395   IF P<90 THEN MON8
396   IF P<100 THEN MON9
397   ENDIF
398   CALL SP,"FUNCMON"
399
400
401 DEF MON4 SP
402
403   VAR T=RND(4)
404   DEF DT=8+T
405   P
406   SET P=100,199,DF OUT SP
407   IF P<1 THEN RETURN
408   VAR P
409   IF P<10 THEN MON0
410   IF P<20 THEN MON1
411   IF P<30 THEN MON2
412   IF P<40 THEN MON3
413   IF P<50 THEN MON4
414   IF P<60 THEN MON5
415   IF P<70 THEN MON6
416   IF P<80 THEN MON7
417   IF P<90 THEN MON8
418   IF P<100 THEN MON9
419   ENDIF
420   CALL SP,"FUNCMON"
421
422
423 DEF MON5 SP
424
425   VAR T=RND(4)
426   DEF DT=8+T
427   P
428   SET P=100,199,DF OUT SP
429   IF P<1 THEN RETURN
430   VAR P
431   IF P<10 THEN MON0
432   IF P<20 THEN MON1
433   IF P<30 THEN MON2
434   IF P<40 THEN MON3
435   IF P<50 THEN MON4
436   IF P<60 THEN MON5
437   IF P<70 THEN MON6
438   IF P<80 THEN MON7
439   IF P<90 THEN MON8
440   IF P<100 THEN MON9
441   ENDIF
442   CALL SP,"FUNCMON"
443
444
445 DEF MON6 SP
446
447   VAR T=RND(4)
448   DEF DT=8+T
449   P
450   SET P=100,199,DF OUT SP
451   IF P<1 THEN RETURN
452   VAR P
453   IF P<10 THEN MON0
454   IF P<20 THEN MON1
455   IF P<30 THEN MON2
456   IF P<40 THEN MON3
457   IF P<50 THEN MON4
458   IF P<60 THEN MON5
459   IF P<70 THEN MON6
460   IF P<80 THEN MON7
461   IF P<90 THEN MON8
462   IF P<100 THEN MON9
463   ENDIF
464   CALL SP,"FUNCMON"
465
466
467 DEF MON7 SP
468
469   VAR T=RND(4)
470   DEF DT=8+T
471   P
472   SET P=100,199,DF OUT SP
473   IF P<1 THEN RETURN
474   VAR P
475   IF P<10 THEN MON0
476   IF P<20 THEN MON1
477   IF P<30 THEN MON2
478   IF P<40 THEN MON3
479   IF P<50 THEN MON4
480   IF P<60 THEN MON5
481   IF P<70 THEN MON6
482   IF P<80 THEN MON7
483   IF P<90 THEN MON8
484   IF P<100 THEN MON9
485   ENDIF
486   CALL SP,"FUNCMON"
487
488
489 DEF MON8 SP
490
491   VAR T=RND(4)
492   DEF DT=8+T
493   P
494   SET P=100,199,DF OUT SP
495   IF P<1 THEN RETURN
496   VAR P
497   IF P<10 THEN MON0
498   IF P<20 THEN MON1
499   IF P<30 THEN MON2
500   IF P<40 THEN MON3
501   IF P<50 THEN MON4
502   IF P<60 THEN MON5
503   IF P<70 THEN MON6
504   IF P<80 THEN MON7
505   IF P<90 THEN MON8
506   IF P<100 THEN MON9
507   ENDIF
508   CALL SP,"FUNCMON"
509
510
511 DEF MON9 SP
512
513   VAR T=RND(4)
514   DEF DT=8+T
515   P
516   SET P=100,199,DF OUT SP
517   IF P<1 THEN RETURN
518   VAR P
519   IF P<10 THEN MON0
520   IF P<20 THEN MON1
521   IF P<30 THEN MON2
522   IF P<40 THEN MON3
523   IF P<50 THEN MON4
524   IF P<60 THEN MON5
525   IF P<70 THEN MON6
526   IF P<80 THEN MON7
527   IF P<90 THEN MON8
528   IF P<100 THEN MON9
529   ENDIF
530   CALL SP,"FUNCMON"
531
532
533
```

RUN

名前を変えてSAVEしておこう

①モンスター1番、②モンスター2番、③モンスター3番

①

```
333 DEF MON1 SP
334 VAR T=-60*2
335 PPOFS PP,"XY",T,RND(400),DY,500
336 PANIM P,"Z",T,0,1
337 SPANIM SP,"R",-16,6,-32,-6,-16,0,0
338 T=T/2
339 VAR C=RGB(255,15,15)
340 PPOFS PP,"C",T,C,T,#WHITE,1
341 PANIM SP,"S",T,C,T,#WHITE,1
342 END
343
344
345
346
347
348
349
350
351
352
```

RUN

③

```
371 DEF MON3 SP
372 VAR T=-60*3
373 PPOFS PP,"XY",T,RND(400),DY,500
374 PANIM P,"Z",T,0,1
375 SPANIM SP,"R",-16,6,-32,-6,-16,0,0
376 T=T/2
377 VAR C=RGB(255,15,15)
378 PPOFS PP,"C",T,C,T,#WHITE,1
379 PANIM SP,"S",T,C,T,#WHITE,1
380 END
381
382
383
384
385
386
387
388
389
390
391
392
393
```

RUN

②

```
350 DEF MON2 SP
351 VAR T=-60*3
352 PPOFS PP,"XY",T,RND(400),DY,500
353 PANIM P,"Z",T,0,1
354 SPANIM SP,"R",-16,6,-32,-6,-16,0,0
355 T=T/2
356 VAR C=RGB(255,15,15)
357 PPOFS PP,"C",T,C,T,#WHITE,1
358 PANIM SP,"S",T,C,T,#WHITE,1
359 END
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
```

RUN

名前を変えてSAVEしておこう

オリジナリティは絵で強調するのが一番早い

画像の修正

お絵かきツールで絵を修正！

- スマイルボタンを押すと表示されるツールの右上にある「おえかき」のボタンをタッチ
- 標準で搭載されているお絵かきツールが起動します。
- 今回のゲームの中で使われている画像は、以下の4枚
 - SP（スプライト用の画像
 - GRP（ゲーム中の背景画像
 - TIT（タイトル画面用の画像
 - GOV（ゲームオーバー用の画像
- 上記画像をお絵かきツールのLOADボタンから読み込んで編集します
- 良い感じに変更できたらSAVEボタンで同じ名前で保存しましょう
- 次にゲームを実行すると自分で描いた絵がゲーム内に登場します

プログラマーの仕事、とプログラマ以外の仕事

ゲームを作る仕事について

ニンテンドー3DSにはどんな部品がありますか？

- ・ニンテンドー3DSに搭載されている部品を調べてみましょう

入力関係

表示関係

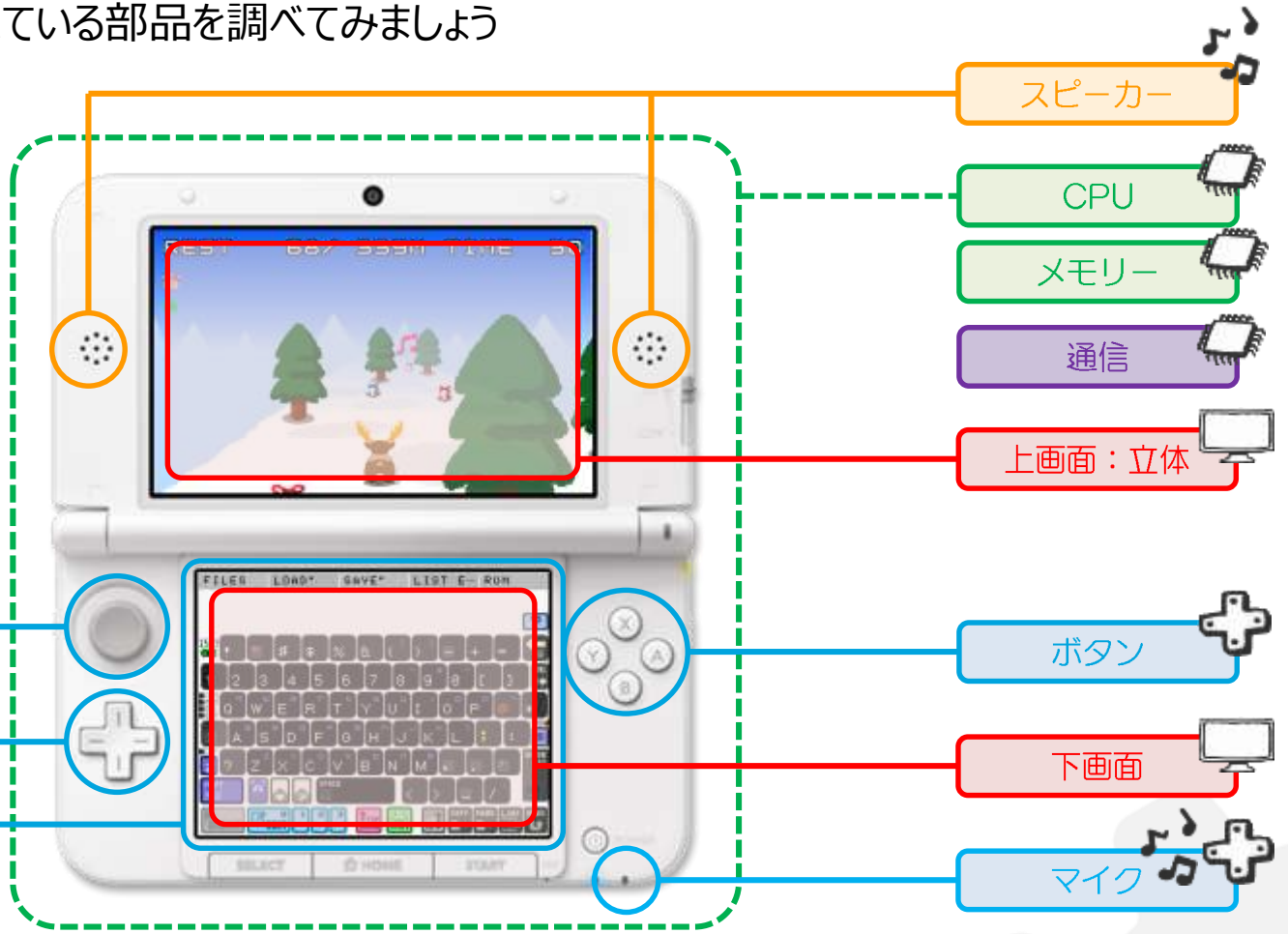
+ 傾きセンサー

+ 加速度センサー

+ スティック

+ 十字キー

+ タッチパネル



「表示機能」「入力機能」「音を鳴らす機能」「通信機能」「CPU」5つの要素が組み込まれている
※PlayStation、Xbox、WiiU、パソコン、スマホなどはどうだろう？

機能ごとのゲームの仕事（職種）

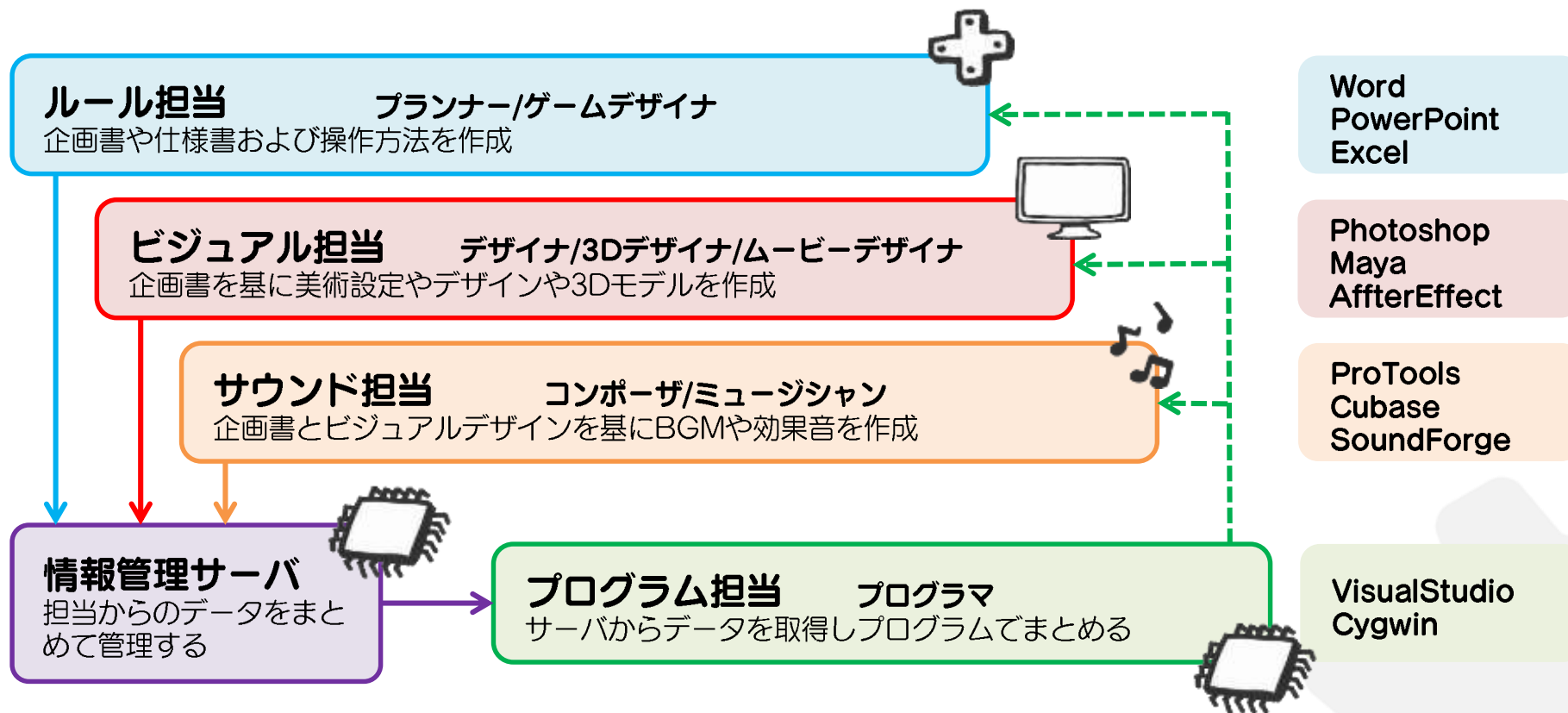
- それぞれ専門の仕事を担当する人達が集まって1つのゲームを作っています

| 部品 | 仕事の名前 | 仕事の内容 | 説明 |
|---|-----------------|---------------------|---------------------------------------|
| 表示機能  | グラフィックデザイナー | 絵を描く仕事 | ゲームの世界設定に合わせた原画を元にドット画像や3Dモデルを作る |
| 入力機能  | プランナまたはゲームデザイナー | ルールを考える仕事 | 操作方法やルールを考えて企画書や仕様書を作成する |
| 音を鳴らす機能  | サウンドコンポーザ | 作曲したり効果音を作る仕事 | ゲームの世界設定に合わせた原画やグラフィック画像を元に効果音やBGMを作る |
| 通信機能  | サーバープログラマ | ネット用のプログラムを作る仕事 | インターネット経由でサーバーとやり取りをするプログラムを作る |
| CPUやメモリー  | プログラマ | 表示や操作や制御のプログラムを作る仕事 | 用意された仕様書を元にゲーム全体のプログラムを作る |

ソフトウェア開発に必要な要素

• ゲーム（ソフトウェア）を作る流れ

- あらゆる作業を一人でこなせないため現在は完全分業化が進んでいる
- 作業ごとに必要なソフトウェアも異なり一人ですべてを覚えられない
- バラバラに作りこんだデータをまとめて管理する方法が重要になっている



最後のごあいさつ（プログラマーになるためには）

- 今日は長い時間がんばってくれてありがとうございます
- プログラムを入力するのは面倒な作業だという事は身にしみて感じられたと思います
- ただ、他の人のプログラムを見ながら入力することと、自分で考えて入力するのは異なります
- 自分が考えて入力する時は想像しながら作っているので時間を忘れて入力できます
- 今回の入力作業が大変だったとしてもプログラムに興味があれば続けてみてください
- 次は自分で考えて一から入力してみると今日とは違った感じが見つかるかも・・・
- プログラマーと言ってもゲーム関係なのかネットワーク屋さんなのかで仕事が変わります
- ゲームのプログラマは絵や音のことも気にしなければならないので他の仕事の知識も必要です
- プチコンだったら一人で絵や音もチャレンジできるので全部一人で作る体験も可能です
- ゲームだけじゃなく色々な知識や技術を身に着ければ他にはないアイデアが生まれます
- 一つのこと集中しすぎないで広くたくさんの経験を積んでほしいと願っています

何か質問があればお応えします。

あとで質問したくなったら、takaki.kobayashi@smileboom.com メールもOKです。