

# ゼロからゲームプログラミング2

平山 尚

# 先週のまとめ

1. メモリにセットする方法
2. 計算と式
3. 画面のメモリ
4. 色
5. 繰り返し
6. 条件実行
7. 外(キーボードやマウス)からセットされるメモリ
8. 名前付きメモリ(いわゆる変数)
9. 部分プログラム(いわゆる関数)

# とはいえ

- 8,9に手を染めてるのは少数。
- 今日は「名前付きメモリ」と「部分プログラム」にも親しめるといいですね。
  - なしで大きなものが書けるなら、それはそれで評価するけど。

# よくあるつまづき1

`memory[0] + memory[1] -> memory[2]`

なんで動かないの？→左辺と右辺が逆。

“`memory[式] -> 式`”

の形にしてください。

“`a -> b`”→“a becomes b”

# よくあるつまづき2

- 数には二種類あることがわかってない
  - 出席番号は「名前」。足したり引いたりしないし、大小に意味はない。一種の名前
  - 5個ある、7メートルある、というのは「大きさ」を表す。足したり引いたりするし、大小にも意味があるが、名前としては使えない。
- プログラム中であるメモリがどっちなのかはちゃんと把握しましょう。
  - 「ジャンプ中なら1にしておく」はどっち？

# よくあるつまづき3

- &&と||がない！
  - ifやwhileは「0」と「0以外」の区別しかしてない。
  - 0か1になる物を2個以上使って0や1を作る計算を「作り出す」という問題。
    - 両方1の時だけ1にしたい時は？→掛け算？
    - 両方0の時だけ0にしたい時は？→足し算？
    - 他にもいろいろあるよ
- 「ある数を入れたらある数になるような計算式を求める」という問題は現実には多数ある。

# よくあるつまづき4

$60000 + (y * 100) + x$ の意味は？

0	1	2	3	4	5	6	7	8	9
10	11	12	13	14	15	16	17	18	19
20	21	22	23	24	25	26	27	28	29
30	31	32	33	34	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	64	65	66	67	68	69
70	71	72	73	74	75	76	77	78	79

- 1回10x10だと思って考えてみて？
  - (2,5)は52、(8,2)は28だよね...
  - こうする式は $(y * 10) + x$ じゃない？

# 新機能1

- 音鳴るようにしてみた。新しいzipを使うと、

memory[55006] -> 440 #frequency

memory[55009] -> 10 #decay

memory[55012] -> 5000 #amplitude

で出る。鳴らす気がなければ古いzipでOK。



# 新機能2

- 解像度を変えられるようにした。

memory[55004] -> 200 #width

memory[55005] -> 150 #height

memory[55000] -> 1 #sync

画素とメモリの関係式が変わるから注意。

$(60000 + (y * 200) + x)$

# 点を塗るのが遅い！

```
memory[55001] -> 1 #disable auto-sync  
while 1  
    ... #your code  
    memory[55000] -> 1 #device sync
```

プログラムの最初で55001番に1を入れ、  
あとは定期的に55000番に1を入れる。  
ただし、動作が見えにくくなるので注意。

# 今日やること1

- お絵描きソフト
  - はみ出しても大丈夫にしてください。
  - 点と点とつなげてください。
  - 右クリック2回でその2点で決まる長方形を描いてください。
  - enterキーで画面を真っ黒にしてください。

# 点をつなぐヒント

- 点と点の間が狭くなればいいんじゃない？
  - 適当に何点か打ってみたら？
  - 真面目な解決策でなくても結果が良ければそれでいい。
  - もちろん、真面目に線を引けるならそれが一番いいけど。

# 今日やること2

- ゲームを作ってください。
  - ゲームじゃなくてもいいけど。去年はソフトキーボードがあった。
- ゲームって？
  - インベーダーとか、古い卓球とか、ブロック崩しとか、そういうのは結構簡単。
  - 縦シューティングもやれる。

# 情報源

- 文書フォルダ内(特にSunabaガイド)
- サンプルフォルダ内(ただ書式が日本語だが)
- SunabaGuideSample内にサンプル多数
  - でもlibraryをそのまま使っちゃダメ。