

自動販売機をプログラミングする -フローチャートやいろいろな設計-



自動販売機って、どこにでもあって便利だね

自動販売機も中に入っているコンピュータで動作しています。今回の授業では、どのようなプログラムが動いているか考えてみましょう。

今回は、Scratchで自動販売機のプログラムを作って、プログラムはどのように設計するか見ていきましょう

Ver 1.0 2017/02/15
© Go Ota, 2014

プログラムなどは、「高校「情報科」の教材・指導案作ってみました。」
<http://www.beyondbb.jp/> Zipの教材内に入っています。

No.1 一番簡単な自動販売機

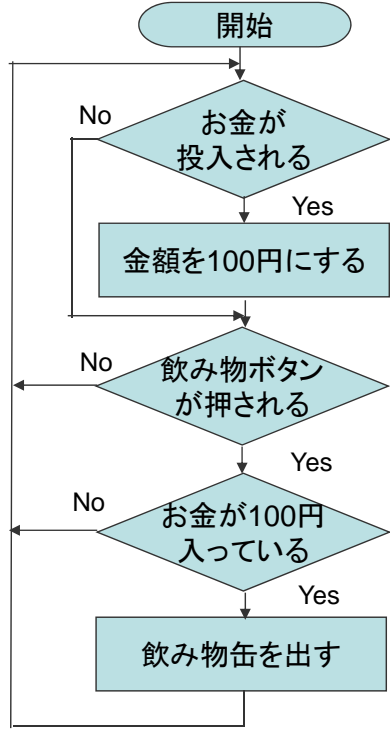


簡単な仕組みの自動販売機

- ・100円玉を1枚だけ入れられる
- ・商品は1つだけ
- ・商品切れランプは無し
- ・お金返却ボタンは無し
- ・つり銭切れランプは無し
- ・お金を入れて一定時間たったら自動的にお金返却は無し

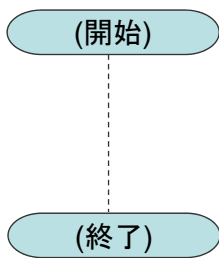


まず、始めの一番簡単な自動販売機について考えてみましょう。お金を入れることと製品のボタンを押すことしかできません。
このプログラムの動作をフローチャートで表すと右の図のようになります。



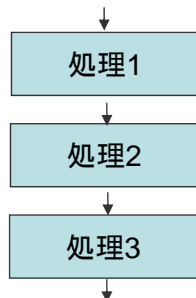
フローチャートの書き方

プログラムの
開始と終わり

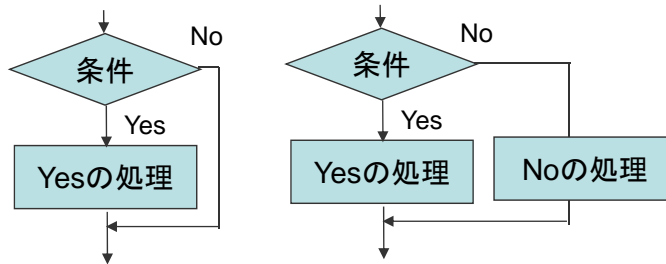


処理の流れの表記(アルゴリズム)

逐次型(直線型)



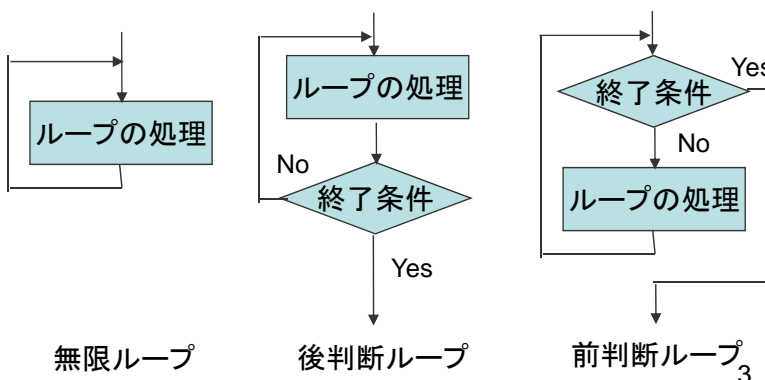
分岐型



プログラムや人間の判断
などのアルゴリズムは基本
的に、逐次型、分岐型、
ループ型の組み合わせで
表現できますね。



ループ型(繰返し型)

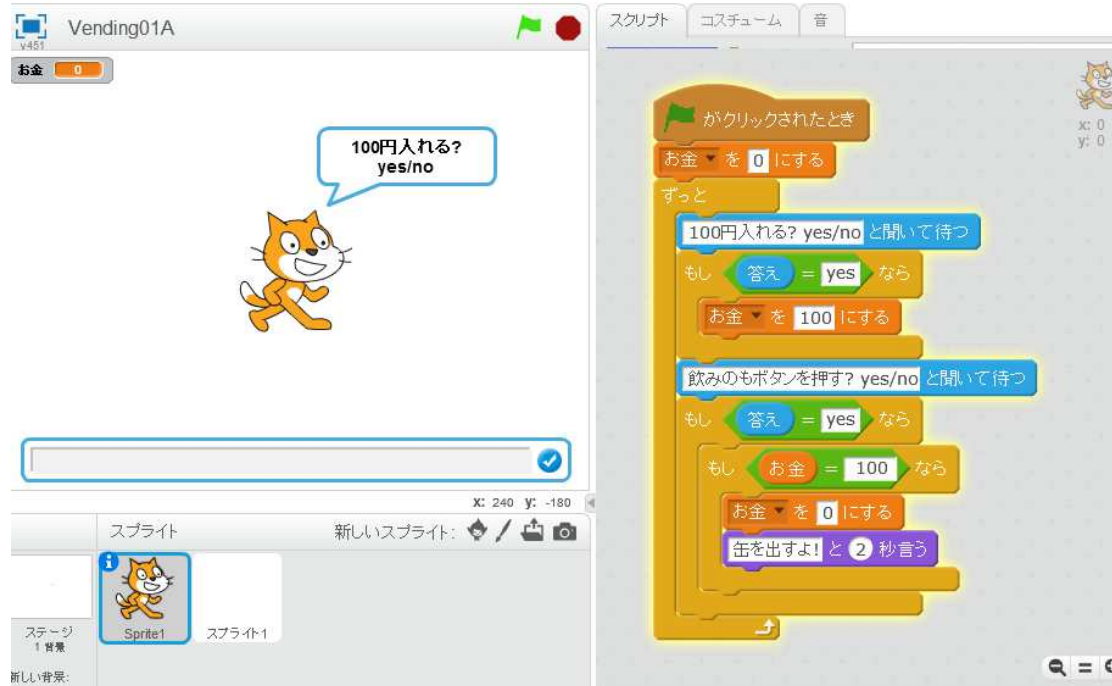


演習: 一番簡単な自動販売機を作ってみよう。

では、一番簡単な自動販売機をScratch
で作っていきましょう。
こんな簡単なしくみでも、ちゃんとした見栄
えを考えると大変かもしれませんね。



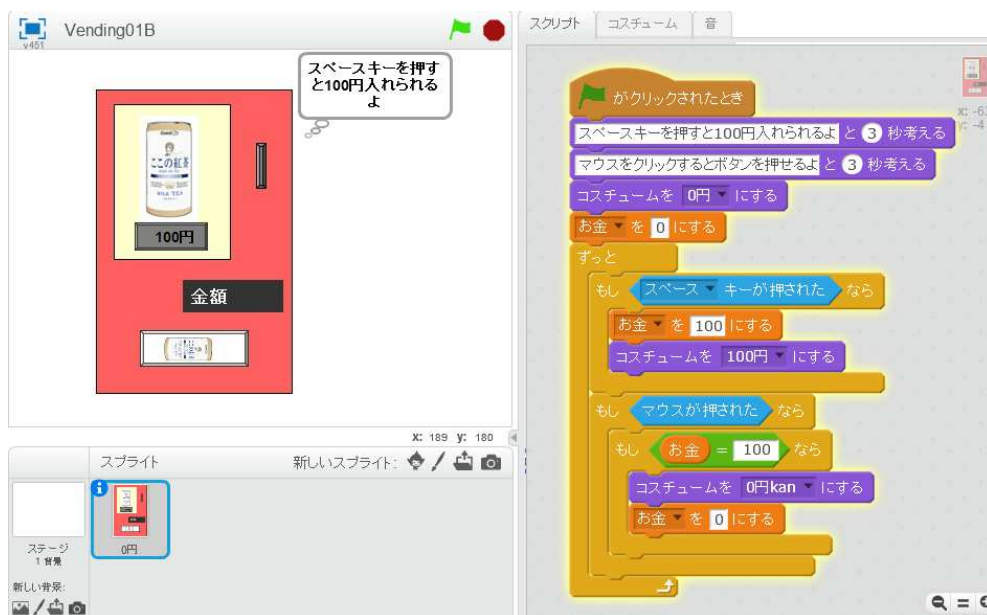
プログラムサンプル1A。[Vending01A]



自動販売機の動きだけをテキストでやってみました。

5

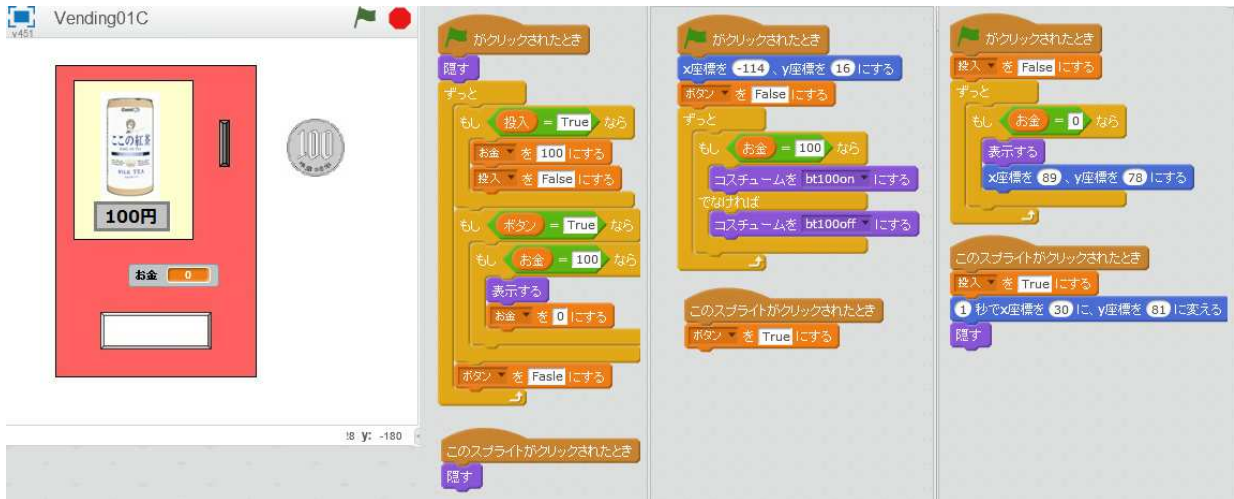
プログラムサンプル1B。[Vending01B]



自動販売機のグラフィックを少しだけ追加してみました。

6

プログラムサンプル1C。[Vending01C]



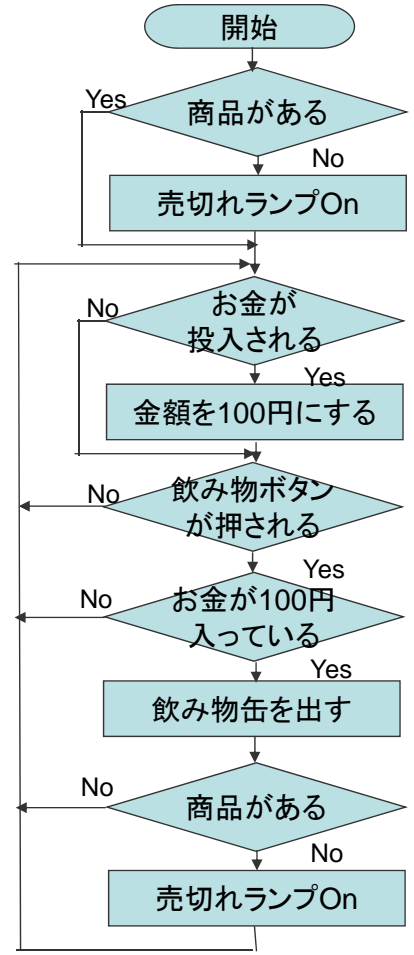
お金を入れたり、ボタンを押したりできるようにすると、少しプログラム自体も複雑になってきます。

ちょっとだけ複雑な自動販売機



- 簡単な仕組みの自動販売機**
- ・100円玉を1枚だけ入れられる
 - ・商品は1つだけ
 - ・商品切れランプは有り
 - ・お金返却ボタンは無し
 - ・つり銭切れランプは無し
 - ・お金を入れて一定時間たったら自動的にお金返却無し

もう少し複雑な自動販売機を考えてみましょう。売切れランプを追加します。フローチャートも少し複雑になりますね。ポイントは、開始直後と「飲み物缶を出す」の後の二か所に、商品の確認をして売切れランプの処理が入ることにあります。初めに品物が無い場合もありますから



演習: もうちょっとだけ複雑な自動販売機



- 簡単な仕組みの自動販売機**
- ・100円玉を1枚だけ入れられる
 - ・商品は1つだけ
 - ・商品切れランプは有り
 - ・お金返却ボタンは有り
 - ・つり銭切れランプは無し
 - ・お金を入れて一定時間たったら自動的にお金返却無し

もう少し複雑な自動販売機を考えてみましょう。お金返却ボタンを追加します。
この自動販売機のプログラムを作ってみてください。
(早く終わった人は、複数の缶がある場合も作ってみてください。)

今までやったフローチャートにとらわれずに、いろいろな方法でプログラムができますので知恵を絞ってみてください。



9

作ったプログラムを見よう A~Eまであるよ

どんなプログラムができましたか?

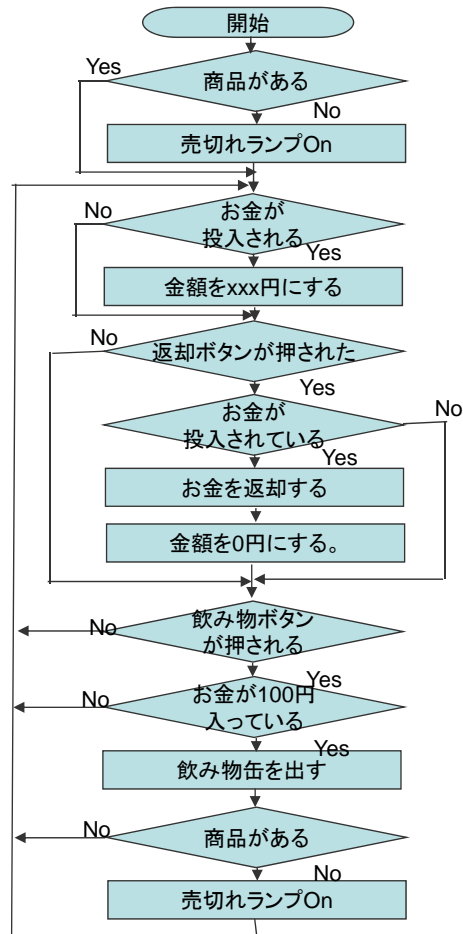
自動販売機といっても、いろいろな方法でプログラミングできます。そのいくつかを、その設計図といっしょに見ていきましょう。



10

A:単純なフローチャートをもとにしたプログラム

返却ボタンの追加では、お金が現在投入されたとか、金額表示を0円にするなどの処理も必要ですね。
このぐらいになるとフローチャートもごちゃごちゃして判りにくくなりますね。



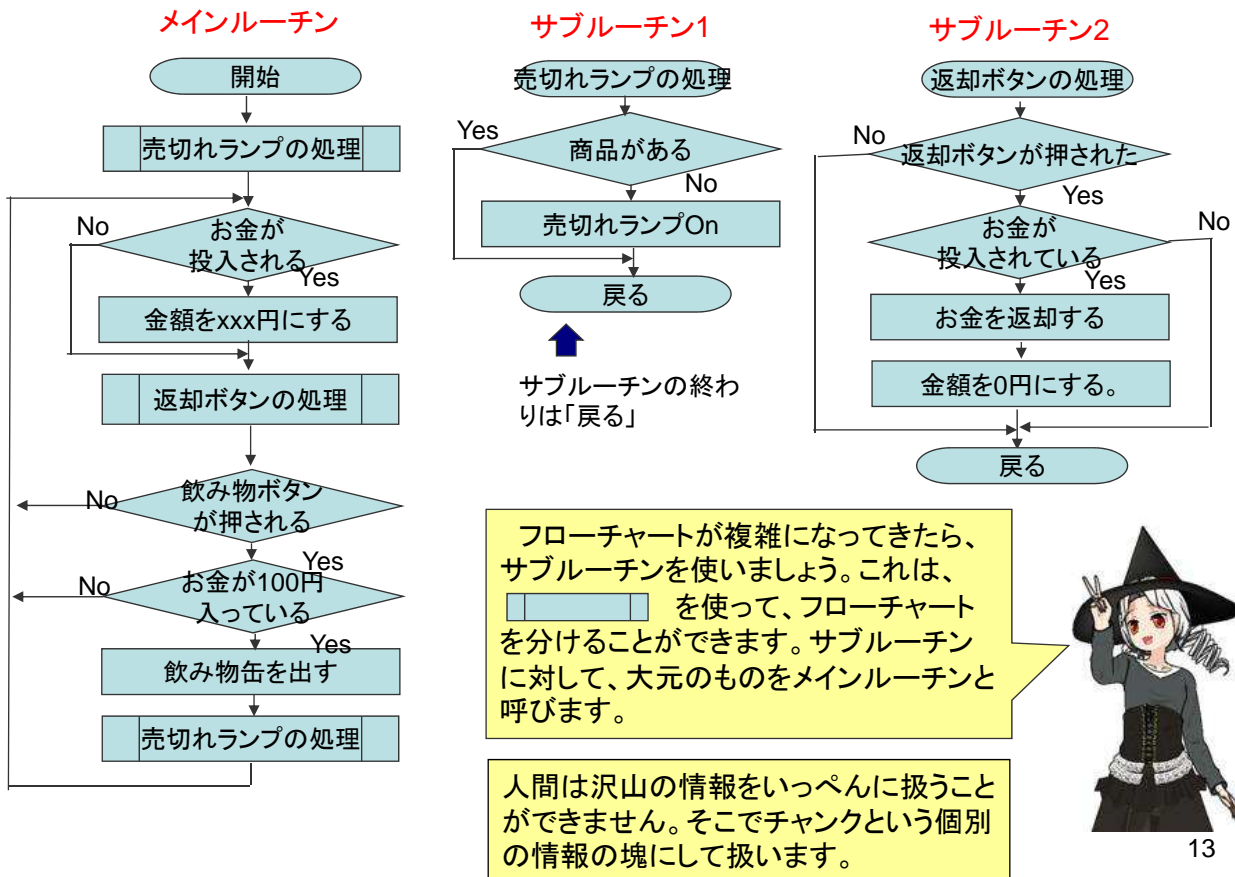
11

プログラムサンプル2A。[Vending02A]

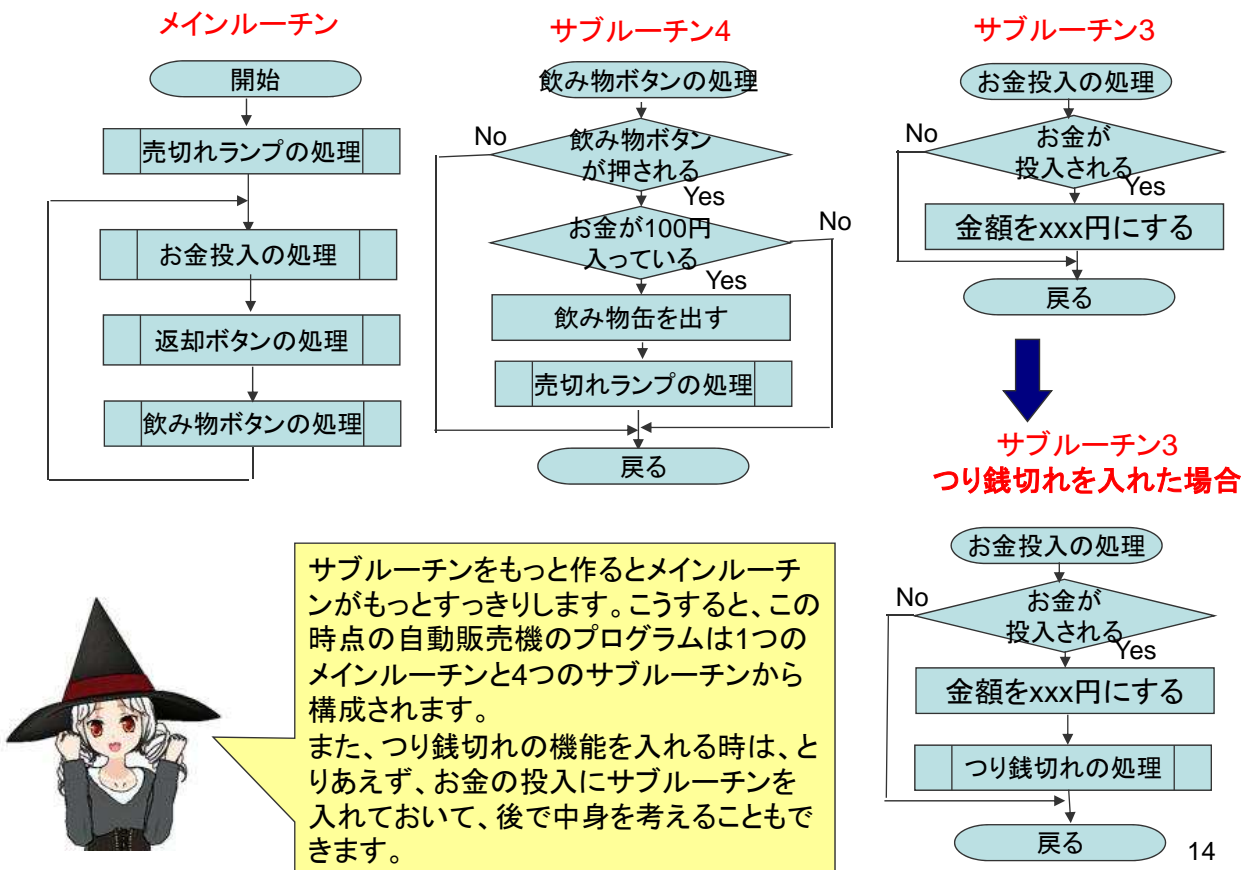
フローチャートをもとにしたプログラムです。これもかなりぐちゃぐちゃしてきましたね。

12

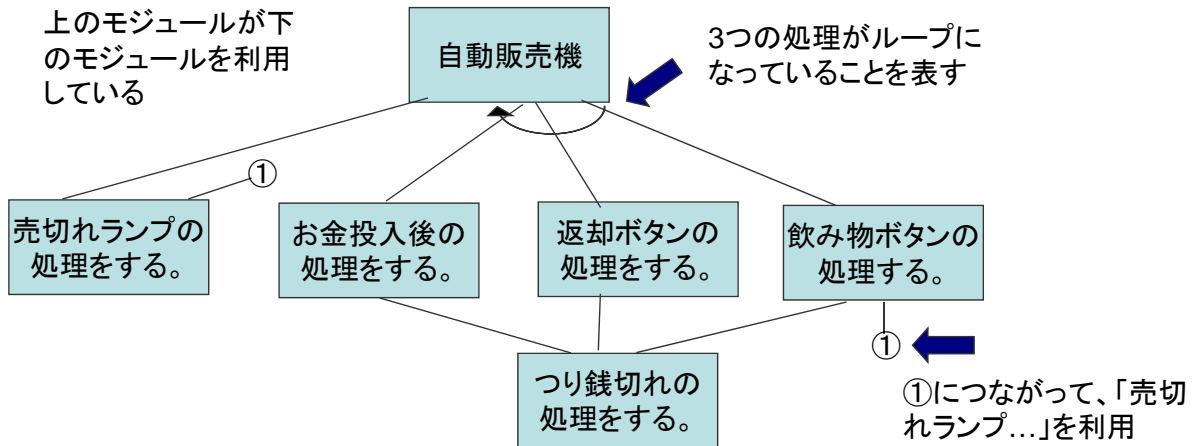
B/C サブルーチンでフローチャートをすっきりさせたプログラム



B/C サブルーチンでもっとフローチャートをすっきり



B/C 構造化設計でプログラミング考えてみる



フローチャートはコンピュータのプログラムを設計する一つの手法ですが、あまり大きなプログラムには向いていません。現在では多くの設計手法がありますが、構造化設計は古いものですが、手軽に利用できます、また現在の設計手法の考え方のベースになっています。フローチャートは処理の流れを中心に考えますが、構造化設計はモジュール(サブルーチン)の構造や関係を中心に表現します。この構造を考えると上図のようにシンプルなものになります。

参考図書:ソフトウェアの複合/構造化設計,G.J. マイヤーズ(1978)

15

プログラムサンプル2B。[Vending02B]

モジュール(サブルーチン)使ったプログラムです。随分すっきりしたように見えます。

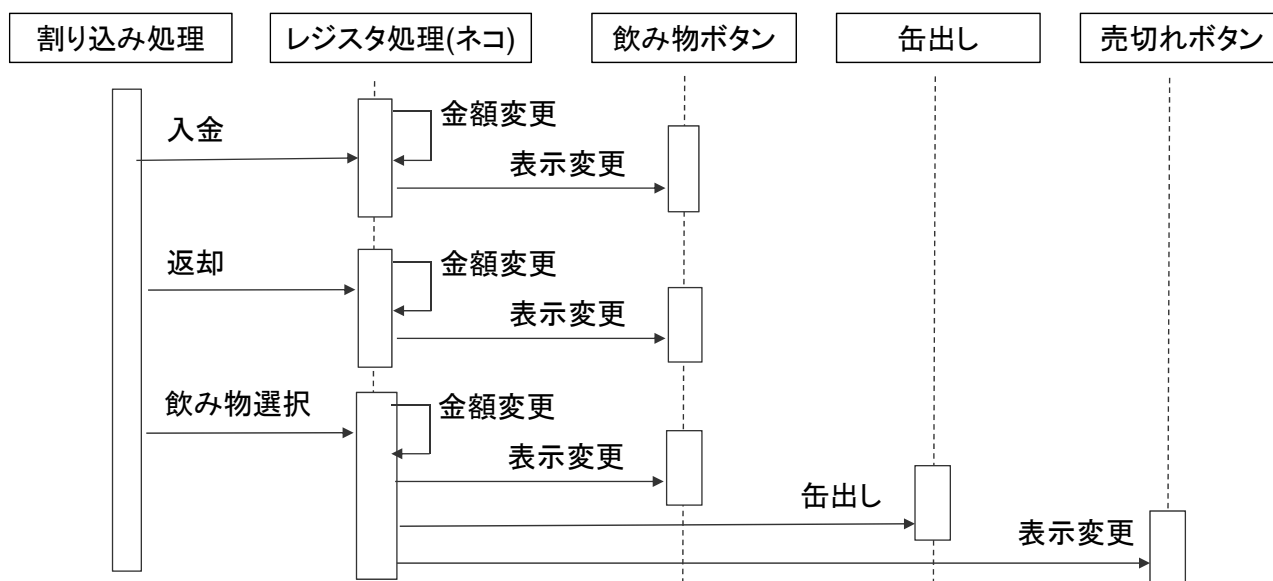
16

プログラムサンプル2C。[Vending02C]

同じくモジュール使ったプログラムです。モジュールはScratchの各スプライトに入れて、メッセージで呼び出しています。

17

D 割り込み処理でプログラムを考えてみる



割り込み処理は多くのコンピュータを内蔵した製品で使用されています。ここではUMLのステート図を使って、動作を表現してみました。

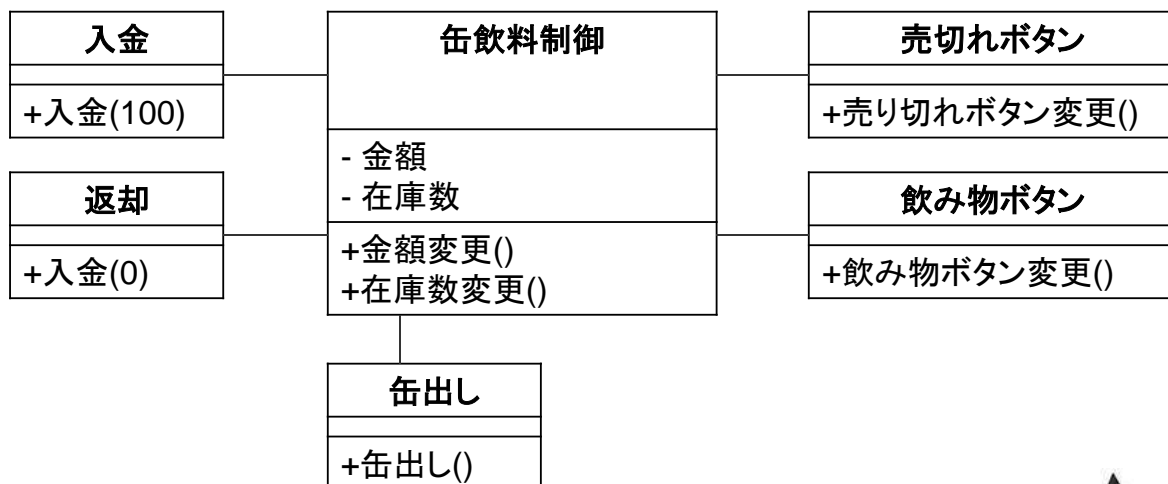
18

プログラムサンプル2D。[Vending02D]

各スプライトをクリックすることで割り込みをメッセージで通知するプログラムです。各イベントの発生に対してのプログラムがすっきりします。

19

E オブジェクト指向っぽくプログラミングしてみる



現在多くのプログラムがオブジェクト指向をもとに開発されています。オブジェクトは個々のプログラムがある役割を持って、それが協働して動くことにより大きな仕事をするものです。ここでは、Scratchのスクリプトごとに機能をもたせるようにプログラミングしてみました。また、その動作をUMLのオブジェクト図類似したもので表現してみました。



20

プログラムサンプル2E。[Vending02E]

各スプライトのスク립トを参照してください。

各スプライトごとの役割がはっきりしています。

21



22