

02. 変数、条件と繰り返し、サブルーチン

02.0 前回の課題の発表会

02.1 ボタンを押した回数を記録する：「変数」と「goto を使った繰り返し」

```
.  
  
; 変数とボタン  
; copyleft 2015-05-21 sugiura@nagoya-u.jp  
; 2019-05-31  
; 2020-05-29 pushTheButton2  
  
kai = 0 ; 変数の初期化  
  
*hajime  
  cls  
  
  pos 300, 200  
  mes kai ; 変数の中身を表示  
  
  button " 押してね ", *tasu  
  button " おしまい ", *owari  
  
  stop  
  
*tasu  
  kai = kai + 1 ; 変数の中身の数を増やす  
  goto *hajime  
  
*owari  
  cls  
  
  ; 改良点  
  ; if kai < 1 { ; if 構文を分けて書く  
  ;   mes " 一度は押してね！ "  
  ;   wait 200  
  ;   goto *hajime  
  ; }  
  
  pos 300, 200  
  mes " あなたは " + kai + " 回押しましたね。お疲れ様。 " ; 文字列と変数の連結  
  
  wait 200  
  
end
```

02.2 「変数」という入れ物

- ・「変数」に数字を入れる。

変数への代入 =

- ・押すたびに、数字を増やす。（「回数の記録」）

n = n + 1

- ・記録された結果（変数の中身）を表示する。

mes 変数

- ・ 変数の中身と文字列をつなげて表示する。(「+」)

mes 変数 + " 文字列 "

02.3 条件で場合分け (流れの制御)

■ pushTheButton2.hsp の改良

(コメント部分の解説)

- ・ 万が一、押さなかった場合の対応 (条件で場合分け)

```
if 条件 {  
    命令  
}
```

- ・ 一行で書く時の書き方

```
if 条件 : 命令
```

02.4 問題文を提示し、反応を記録する

・

```
; 2 問クイズ  
; copyleft 2016-05-08 sugiura@nagoya-u.jp  
; 2019-05-31  
; 2020-05-29 2monQuiz2  
  
*mon1  
cls  
pos 150, 150  
font " M S 明朝 ", 18, 1  
mes " 正しい場合は、マウスの左ボタン "  
mes " 間違いの場合、マウスの右ボタン "  
mes ""  
key = 0 ; どのキーを押したかを記録  
  
mes " 第 1 問 "  
mes ""  
mes " アメリカの首都はニューヨークである。 "  
mes ""  
  
repeat  
    stick key ; 押されたキーの番号取得し key に入れる  
    if key==256 : gosub *matigai : break ; 256 はマウス左ボタン  
    if key==512 : gosub *seikai : break ; 512 はマウス右ボタン  
    await 1 ; repeat のループに一瞬の隙を作る  
loop  
  
*mon2  
cls  
pos 150, 150  
font " M S 明朝 ", 18, 1  
mes " 正しい場合は、マウスの左ボタン "  
mes " 間違いの場合、マウスの右ボタン "  
mes ""  
key = 0
```

```

mes " 第 2 問 "
mes ""
mes " カリフォルニアの州都はサンフランシスコである。 "
mes ""

repeat
  stick key
  if key==256 : gosub *matigai : break      ; key&256 でもよい
  if key==512 : gosub *seikai : break
  await 1
loop

;-----
*owari

cls
pos 150, 150

mes " あっていた数は、 " +maru+ " 個です。 "
mes ""
mes " まちがった数は、 " +batu+ " 個です。 "

wait 300

end

;-----sub-----
; サブルーチンは、必ず return で、元に戻る

*seikai
  maru = maru + 1
return

*matigai
  batu = batu + 1
return

```

■ スクリプトの解説：プログラムの流れ

- ・ 問題文を呈示する。
- ・ 「はい」「いいえ」で回答してもらう。

■ 「stick」を使った入力のチェック：どのボタンを押したか

```
stick
```

■ 「repeat」を使った繰り返し

```
repeat 繰り返す回数
  命令
loop
```

- ・ 繰り返し回数を書かない = 永遠に続ける
 - ・ ひたすら「内容」を実行し続ける状態（指定回数なし = 無制限）
 - ・ を書いておくことになっている。書かないと「暴走」し続けてしまう。

```
repeat
  繰り返す内容
  await 1
loop
```

- ・ 条件が満たされるまで待つ：条件が合ったら「break」で抜け出す

```
repeat
```

```
繰り返す内容
if 条件 : break
await 1
loop
```

■ 回答の記録：場合に分けて、別の変数に記録

- ・ 合っていたら「正答数」に記録する。
- ・ 間違っていれば「誤答数」に記録する。
- ・ いくつ合っていたかを最後に表示する。

02.5 サブルーチン

- ・ 同じ「処理」は、一カ所に「部品」としてまとめ、再利用
 - ・ 普通の「段落」との違いは、ところ
- ・ スクリプトの一番下にまとめて書いておく

課題： 10問クイズを作る

■ できるだけサブルーチンにまとめてみよう。

スクリプトエディターの使い方のコツ

- 1) コメントを使って、試行錯誤
- 2) ヘルプでキーワード検索
- 3) 検索と置換

=> TextForHSP03