大人のScratchプログラム機能部品カード　No.01

四則演算と変数への代入

足し算、引き算、掛け算、割り算の基本的な使い方と、数値や計算結果の変数への代入方法です。

部品 01

具体例

変数Iの内容に1を足した値

変数Aと変数Iと10を足した値

変数Xの内容から2を引いた値

変数 A, 変数B, 変数Cの内容を足す

変数Xと変数Iをかけた値

変数Aを10で割った値

変数Aの内容を0にする。

変数Xの内容を変数Iに1を足したものにする。

裏面

大人のScratchプログラム機能部品カード　No.01

四則演算と変数への代入

図式例

A = 0

X = I + 1

大人のScratchプログラム機能部品カード　No.02

キーボードから数値や文字の入力

キーボードから数値や文字を入力して変数に入れます。

部品 02

具体例

キーボードから入力した数を変数Xに入れる。

キーボードから入力した数を変数Aと変数Bに入れる。

裏面

大人のScratchプログラム機能部品カード　No.02

キーボードから数値や文字の入力

図式例

X = (入力)

A = (入力)

補足

入力する命令も、変数自体も特に文字用、数値用の区別はなく。どちらも扱えます。文字か数値かはプログラムを作る人が判断して利用します。

大人のScratchプログラム機能部品カード　No.03

条件を判断して、命令を実行

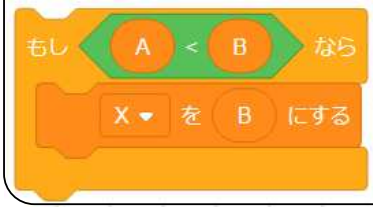
条件を判断して、条件が成り立つ時に命令を実行します。

具体例

部品 03

変数Aの内容が70より大きければ、合格(Goukaku)と表示する。

変数Aの内容が、変数Bの内容より小さければ、変数Xに変数Bの内容を入れる。

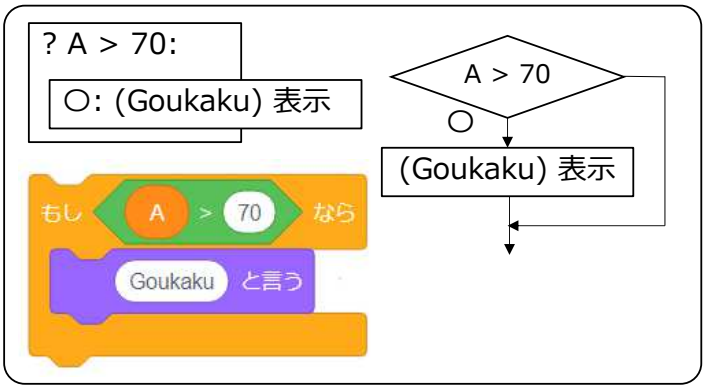


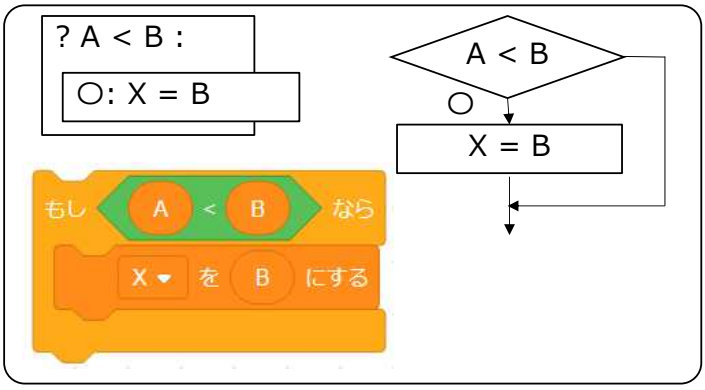
裏面

大人のScratchプログラム機能部品カード　No.03

条件を判断して、命令を実行

図式例





大人のScratchプログラム機能部品カード　No.04

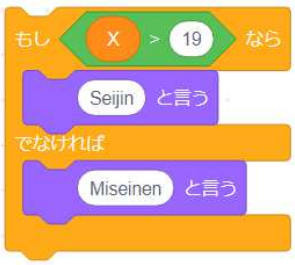
条件を判断して、○と×で違う命令を実行

条件を判断して、条件が成り立つ時と成り立たない時に別々の命令を実行します。

具体例

部品 04

変数Xの内容が19より大きいか判断して。大きければ成人(Seijin)と表示し、そうでなければ、未成年(Miseinen)と表示する。



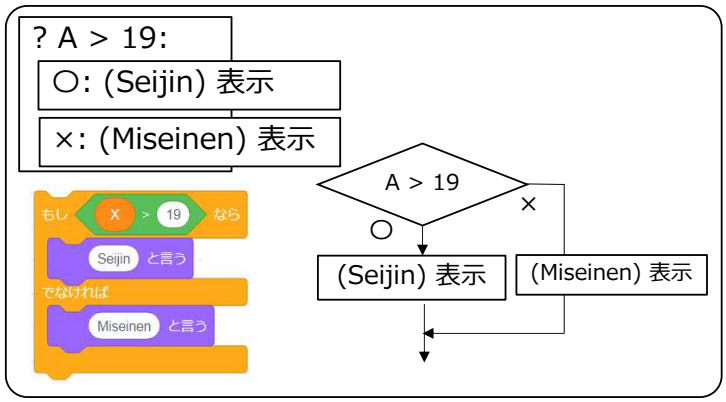
変数Aの内容が変数Bの内容より大きいか判断して。大きければ変数Xに変数Aの内容を入れ、そうでなければ、変数Xに変数Bの内容を入れる

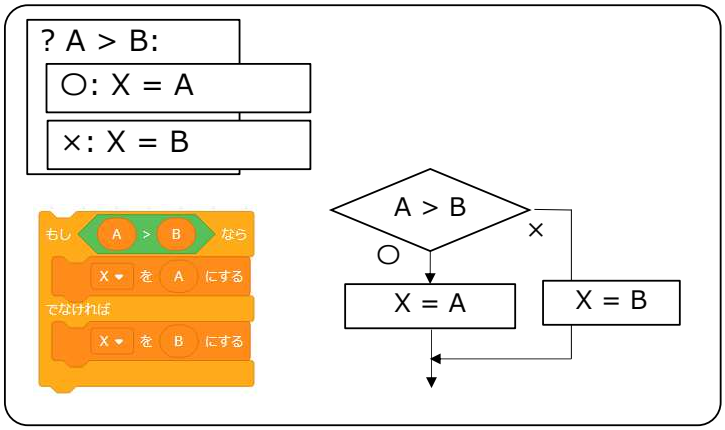
裏面

大人のScratchプログラム機能部品カード　No.04

条件を判断して、○と×で違う命令を実行

図式例





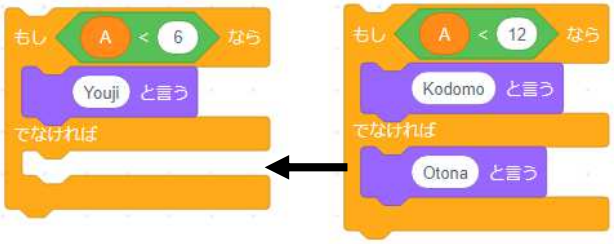
大人のScratchプログラム機能部品カード　No.05

複数の条件を判断して違う命令を実行

部品 05

複数の条件を判断して、違う命令を実行します。
No.03やNo.04で、実行する命令として、No.03やNo.04自体を使用します。

具体例



条件を判断する二つの部品を組み合わせます。

まず、変数Aの内容が6未満であるか判断して、未満であれば、幼児(Youji)と表示します。

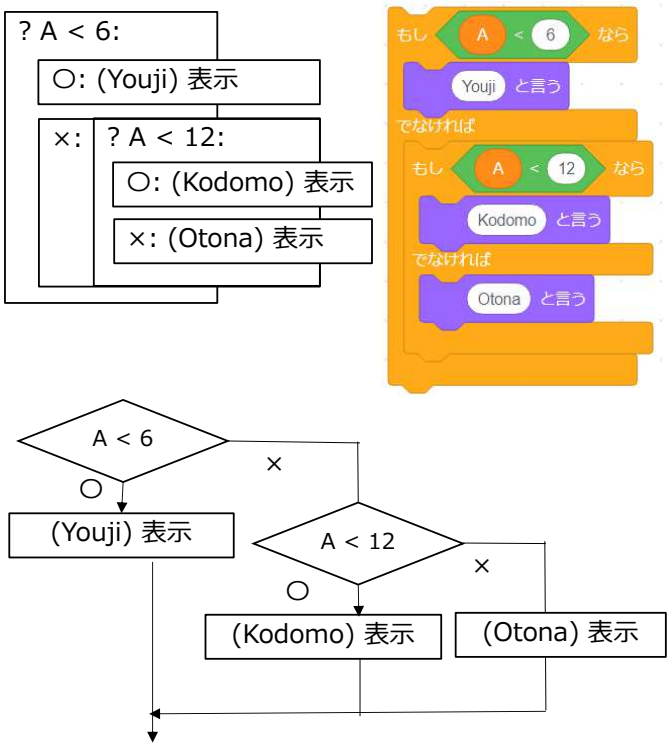
もし、そうでなければ、今度は、変数Aの内容が12未満であるか判断して、未満であれば、子供(Kodomo)と表示します。そうでなければ、大人(Otona)と表示します。

裏面

大人のScratchプログラム機能部品カード　No.05

複数の条件を判断して違う命令を実行

図式例



大人のScratchプログラム機能部品カード　No.06

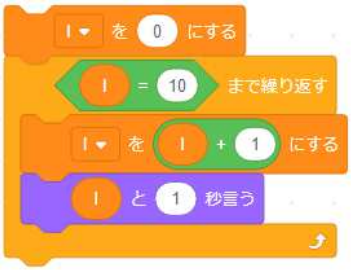
数をカウントアップしながら繰り返す

部品 06

ある変数の内容をカウントアップしながら、指定した数になるまで、繰り返して処理します。

具体例

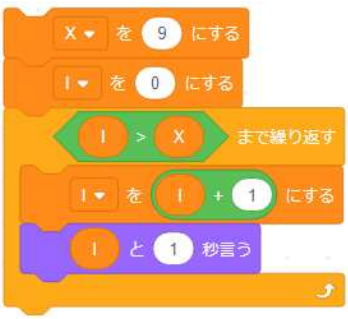
変数Iの内容を1, 2, 3 …と10までカウントアップして、処理を繰り返しています。



<- 変数Iの内容を0に設定

<- 変数Iの内容を判断して、繰り返しの終了を判断
<- 変数Iの内容を1づつカウントアップ

変数Iの内容を1, 2, 3 …とカウントアップして、変数Xの内容+1まで処理を繰り返しています。(上と同じ処理になっています)



<- 変数Xに終了の判断の数を設定
<- 変数Iの内容を0に設定

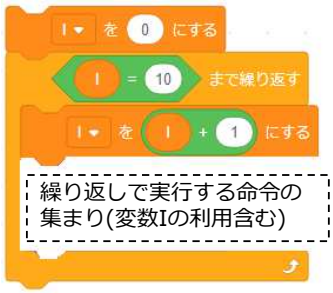
<- 変数Iの内容を判断して、繰り返しの終了を判断
<- 変数Iの内容をづつカウントアップ

裏面

大人のScratchプログラム機能部品カード　No.06

数をカウントアップしながら繰り返す

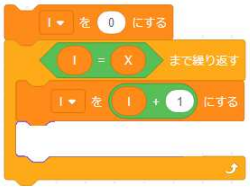
図式例



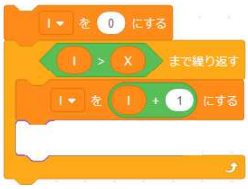
I = 1,2,…
I = 10まで

繰り返しで実行する命令の集まり(変数Iの利用含む)

補足



変数Xの内容の数まで繰り返します。
例えば、変数Xに10が入っていた場合Iが1, 2, 3, …9, 10まで変化します。



変数Xの内容の数+1まで繰り返します。
例えば、変数Xに10が入っていた場合Iが1, 2, 3, …10, 11まで変化します。

大人のScratchプログラム機能部品カード　No.07

リストの処理(取り出し、入れ替え、追加)

リストの中の個々の要素の取り出し、入れ替え、追加、クリアをします。

部品 07

具体例

リストDの3番目の内容です。

リストDの(Iの変数の内容)番目の内容です。
例えばIに4が入っていたら4番目の内容です。

リストDの3番目の内容を2にします。

リストDの(Iの変数の内容)番目の内容を変数Xの内容にします。
例えばIに4、Xに100か入っていたら4番目の内容を100にです。

裏面

大人のScratchプログラム機能部品カード　No.07

リストの処理(取り出し、入れ替え、追加)

図式例

A = D[3]

? D[I] < 10 :
〇: (処理)

D[3] = 2

D[I] = X

大人のScratchプログラム機能部品カード　No.08

二つの変数の内容を入れ替える

二つの変数の内容を入れ替えます。入れ替える時、片一方の変数の内容を一時的に他の変数に入れるのがポイントです。

部品 08

具体例

変数Aと変数Bの内容を入れ替えます。(変数AにBの内容を入れる前に、変数Aの内容をTに入れて退避させています)

リストDの1番目の箱の内容とI番目の内容を入れ替えています。例えば、変数Iに5が履いていたら、リストDの1番目と5番目の内容を入れ替えています。

裏面

大人のScratchプログラム機能部品カード　No.08

二つの変数の内容を入れ替える

図式例

A <-> B　又は

T = B
A = B
B = T

D[1] <-> D[I]　又は

T = D[1]
D[1] = D[I]
D[I] = T

大人のScratchプログラム機能部品カード　No.9

繰り返しとリスト利用の組み合わせ

ある変数の内容をカウントアップしながら、指定した数になるまで、繰り返して処理します。その処理の中で変数の値を番号としてリストを利用します。

部品 09

具体例

変数Iの内容を1づつカウントアップして、1, 2, 3, 4, 5まで繰り返します。その繰り返しの中の処理で、変数Iの内容を番号としてリストDの箱の中を順番に表示しています。

裏面

大人のScratchプログラム機能部品カード　No.9

繰り返しとリスト利用の組み合わせ

図式例

I = 1,2,…
I = 10まで

繰り返しで実行する命令の集まり(変数Iの利用含む)

D[I] の利用

補足

多くの場合、繰り返しとリストを組み合わせて利用する場合は繰り返しのカウントアップを、リストの番号指定に使用します。

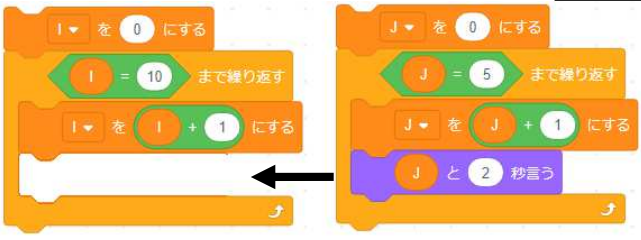
大人のScratchプログラム機能部品カード　No.10

二重の繰り返し

ある変数の内容をカウントアップする繰り返しを二つ組み合わせて使用します。

部品 10

具体例



変数Iでカウントアップ　　変数Jでカウントアップ

繰り返しの二つの部品を組み合わせます。

変数Iの内容が1, 2, 3…10とカウントアップして下記の変数Jを使った繰り返しを10回繰り返します。

変数Jの内容が1, 2, 3, 4, 5とカウントアップして表示します。

裏面

大人のScratchプログラム機能部品カード　No.10

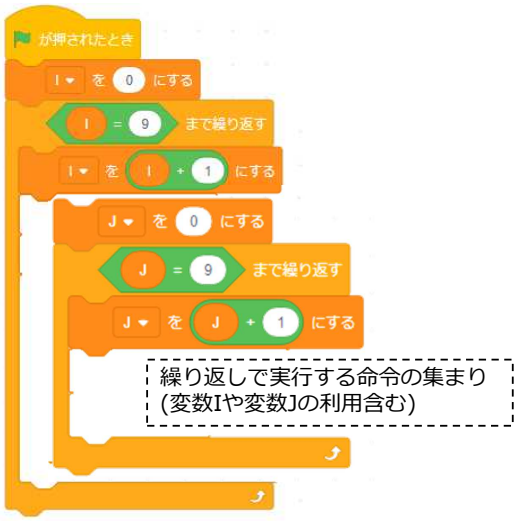
二重の繰り返し

図式例

I = 1,2,…　I = 9まで

J = 1,2,…
J = 9 まで

繰り返しで実行する命令の集まり
(変数Iや変数Jの利用含む)



大人のScratchプログラム機能部品カード　No.11

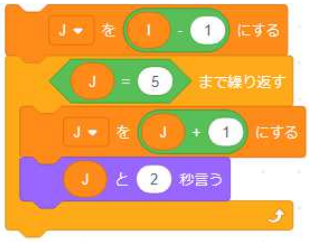
二重の繰り返し(内の繰り返しの開始を変更)

ある変数の内容をカウントアップする繰り返しを二つ組み合わせて使用します。但し、内側のカウントアップする数の初めを外側の繰り返しの変数の内容にしています。

部品 11

具体例

変数Iでカウントアップ



変数Jでカウントアップ
初めにI-1を設定(処理内はIから)

繰り返しの二つの部品を組み合わせます。

変数Iの内容が1, 2, 3, 4, 5とカウントアップして下記の変数Jを使った繰り返しを5回繰り返します。

変数Jの内容が、個々の繰り返しが始まった時のIの内容から始めるので
1, 2, 3, 4, 5
2, 3. 4. 4
3, 4, 5
4, 5
5とカウントアップして表示します。

裏面

大人のScratchプログラム機能部品カード　No.11

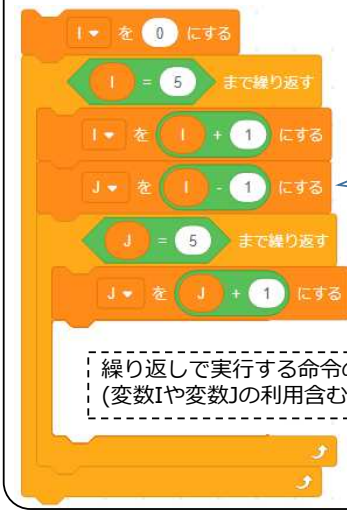
二重の繰り返し(中の繰り返しの開始を変更)

図式例

I = 1,2,…　I = 5まで

J = I, I +1, I+2 …
J = 5 まで

繰り返しで実行する命令の集まり
(変数Iや変数Jの利用含む)



I - 1にするのがポイント

繰り返しで実行する命令の集まり
(変数Iや変数Jの利用含む)