

「マイクラフトでプログラミング」

PC の場合 プログラミング画面の出し方

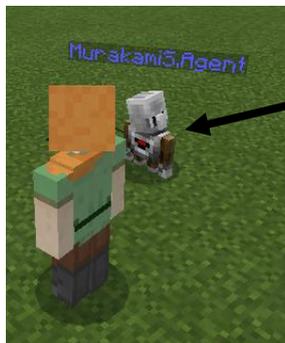
C を押すとプログラミング画面が出てくる

T を押すとプログラミングプログラミング入力画面が出てくる

E を押すとクリエイティブモードの場合、いろいろなブロックを選ぶことができる。

かいたプログラムを実際に作業してくれるのはエージェント

(1) エージェントを自分の位置に戻る(エージェントを呼び戻そう)



エージェントって、作業するロボット



(2) エージェントを左をむかせる



コマンド名は何でもよい
覚えやすいもの
ただし半角英数

エージェントにブロックを置いてもらおう(建築物を作る基本)

ーブロックを並べる必要最低限のコマンドプログラミングー

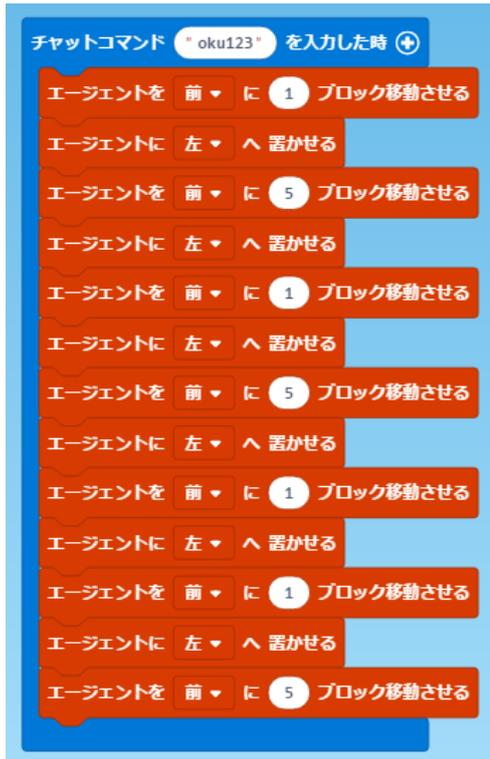
(1) 3つのブロックを左に並べる 左の置くー1歩前ー左の置くー1歩前ー左の置くー1歩前



(2) 3つのブロックを並べ5歩いて3並べるを3回繰り返す



(3) ブロックを1個 5歩 2個 5歩 3個 ならべる



(4) 1辺が3の正方形

使うコマンド

「エージェントを 前 に 1 ブロック移動させる」

「エージェントに 下 にへ置かせる」

「エージェントの向きを 左 に変える」

繰り返しコマンドを使わない場合長いプログラムになる



(5)1 辺が 5 の正方形

1 辺が 5 が正方形

プログラムがとても長い！

くりかえしコマンドを

どの部分を何回

くり返しているか

考えよう



繰り返しコマンド

次に繰り返しコマンドを利用するとプログラムが簡単になるということを理解する

(1) 繰り返しコマンドを使って、ブロックを20個連続で並べてみよう。



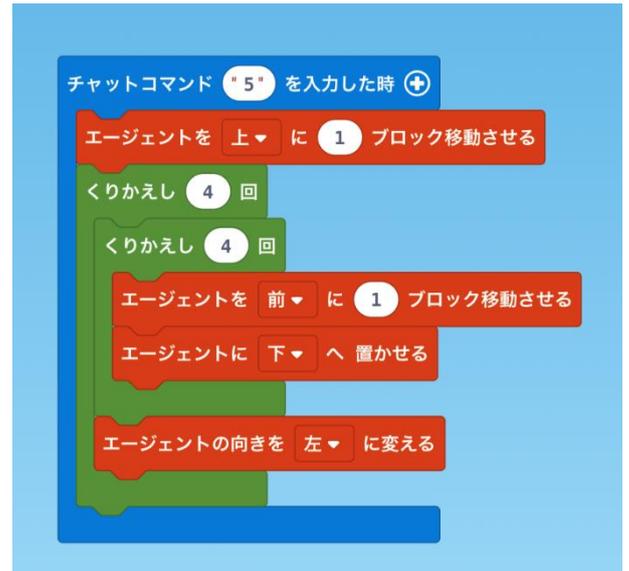
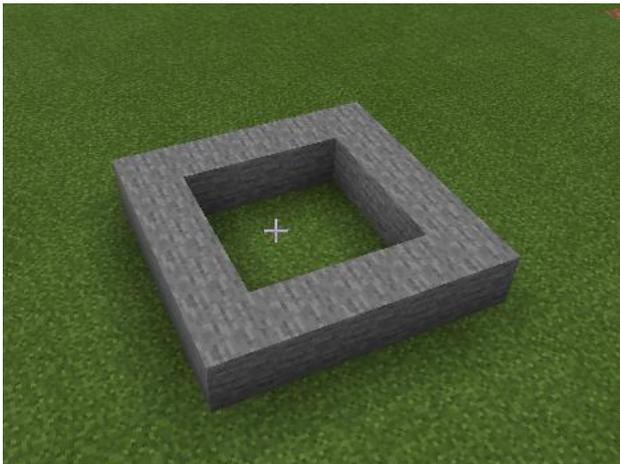
(2) 3つのブロックを並べ5歩いて3並べるを5回繰り返す



何を何回繰り返しているか を考えてくりかえしコマンドのなかに入れよう
くりかえしコマンドの中にくりかえしコマンドを入れることができます

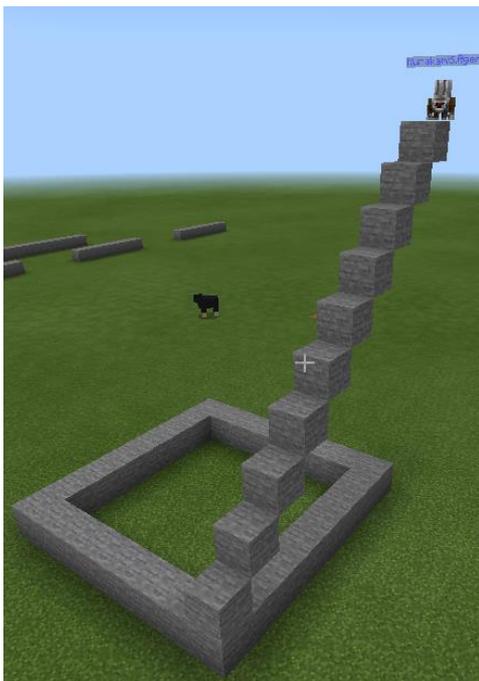


(3) 1 辺が 5 の正方形



発展練習問題

(4) 10 段階をつくる



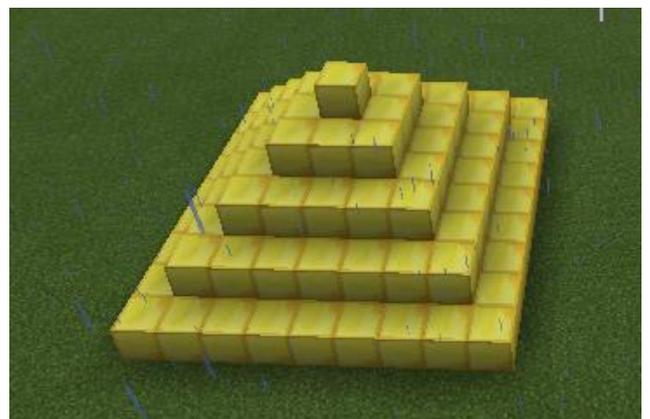
(5)3 段ピラミッド

エージェントの向きを変えるというコマンドが必要になる
何を何回繰り返すのか 規則性を考えよう
またピラミッドの対称性を利用しよう



(6)5 段ピラミッド

何を何回繰り返すのか 規則性を考えよう



またピラミッドの対称性を利用しよう

- (1) 図のような高さ5段のピラミッドを作るには、1番下の段は何個のブロックが必要だろうか
- (2) 1段上がるごとに何個ずつブロックが減っていくだろうか
- (3) 規則性を見つけよう
高さn段のピラミッドをつるには、1番下の段は何個のブロックが必要だろうか
- (4) 対称性注目して、何を何回繰り返すプログラムを作ればよいであろうか。

ヒント・・・ ブロックは下に置く という設定の方がよい。

変数を利用しよう

変数を使えるようになれば、自分が作りたいプログラムは、たいていできるようになる

① 1種類の変数を用意して、変数に数字を代入

n=10を代入すると10回繰り返すといった繰り返しの指定をできるようにする

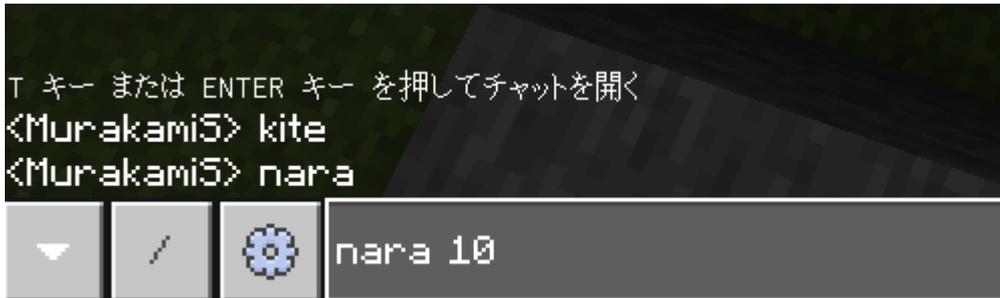
(1) ブロックをn個並べよう

n = 10を指定すると、ブロックをn個並べてくれる



使い方

『チャットコマンド (スペース) 半角数字』で入力



(2) 3つのブロックを並べ5歩いて3並べるをn回繰り返す



変数をここに入れる



1辺がnの正方形

n = 7 の場合



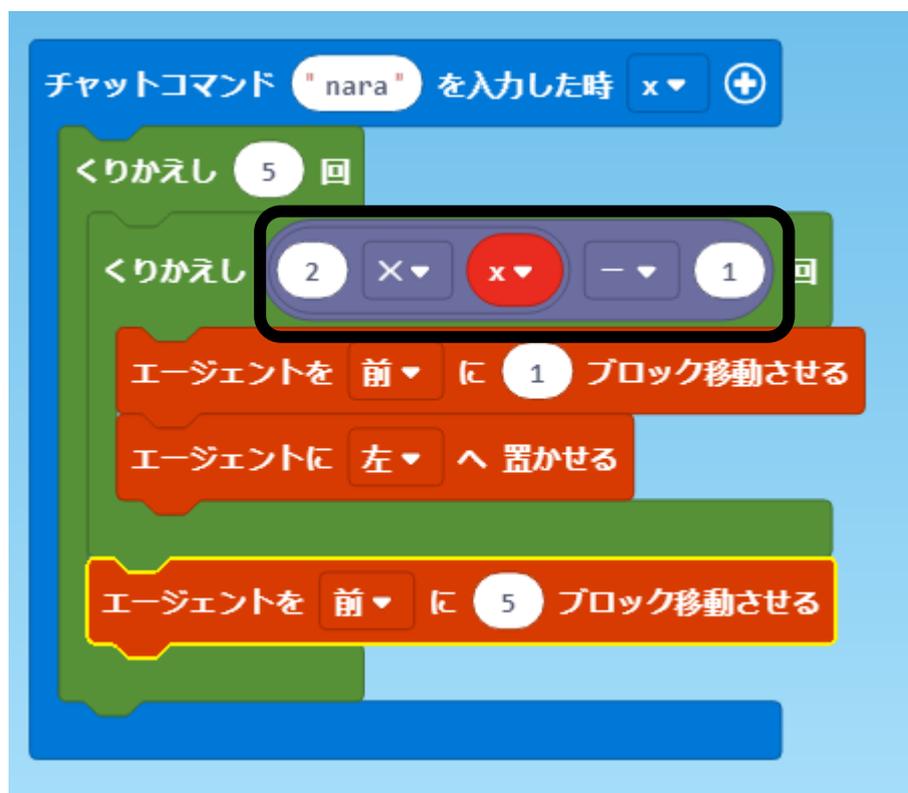


(4) 入力した数字を奇数回繰り返す - 数式 $2n-1$ の作り方 -

$2n-1$ 個のブロックを並べる

ブロックを 7個 -5歩 - 7個 -5歩 - 7個 -5歩 - 7個 -5歩 - 7個 -5歩ならべる

$n=4$ を代入して、ブロックを並べるようにする



② 2種類の変数を用意して、変数に数字を代入

ポイント

2種類の変数を利用するときには、
入力用 と プログラムの中で動く用
の2種類を用意する

プログラムの中で動く用 は必ず最初にいくらから出発するのかがかかないといけない
(初期値をいくらにするかがかかないといけない)

(1) たとえば3と入力するとブロックを左1個 5歩 左2個 5歩 左3個 ならべるプログラムを作る



たとえば10と入力すると
ブロックを1個 5歩 2個 5歩 3個・・・5歩 10個 ならべる



```
when chat command "henrepoku" is entered
  set variable okuhen to 0
  if okuhen <= kaisu then loop
    move agent left
    move block forward by 1
    move agent forward by 5
    increase okuhen by 1
```

入力用の変数をここに入れ

増やしていくもう1つの変数、いくらから出発するか定義する

何回繰り返すか、いつまで繰り返すか数式で定義する

(2) 下の図のような正方形の1辺を増やしながら正方形を並べる

■1辺1の正方形 5歩 1辺2の正方形 5歩 1辺3の正方形 5歩 1辺4の正方形
(繰り返しと変数を用いる)

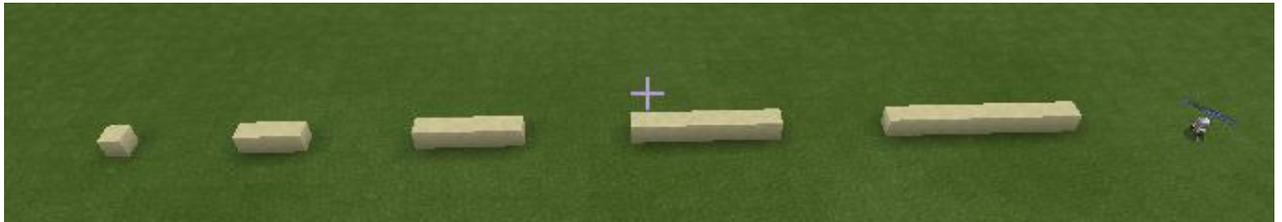


```
チャットコマンド "seihou" を入力した時 kaisu ▼ (+)
変数 okuhen ▼ を 0 にする
もし okuhen ▼ ≤ kaisu ▼ ならくりかえし
  くりかえし okuhen ▼ 回
    くりかえし okuhen ▼ 回
      エージェントに 左 ▼ へ 置かせる
      エージェントを 上 ▼ に 1 ブロック移動させる
      エージェントを 下 ▼ に okuhen ▼ ブロック移動させる
      エージェントを 前 ▼ に 1 ブロック移動させる
      エージェントを 前 ▼ に 5 ブロック移動させる
      変数 okuhen ▼ を 1 だけ増やす
```

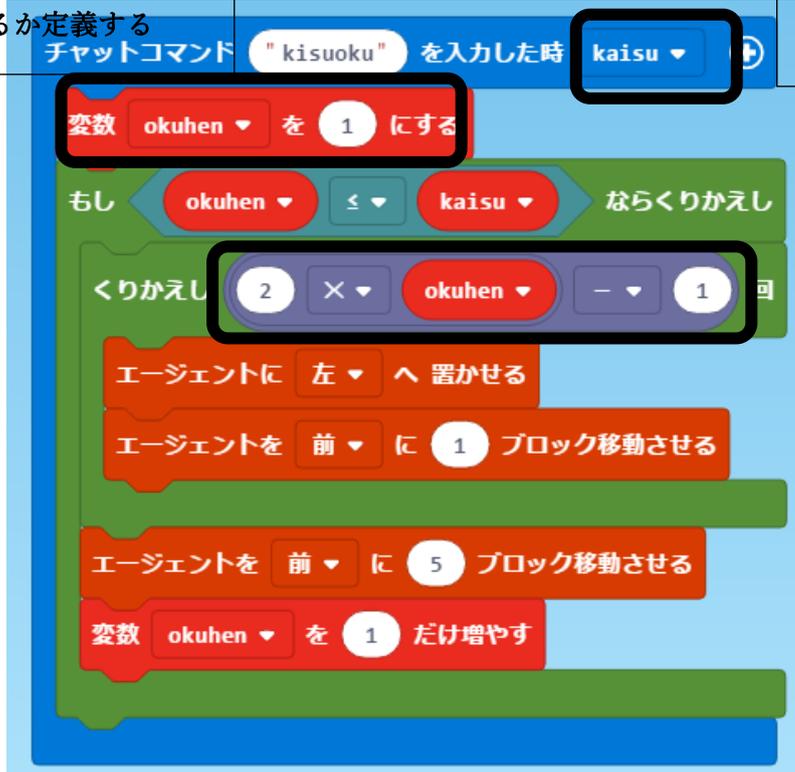
(3) 入力した数字を奇数回繰り返すー数式 $2n-1$ の作り方ー

$2n-1$ 個のブロックを並べる

ブロックを 1個 -5歩- 3個 -5歩- 5個 -5歩- 7個 -5歩- 9個ならべる
奇数個、数字を増やしながら並べる
 $n=5$ を代入して、ブロックを並べるようにする



増やしていくもう1つの変数、いくらかから出発するか定義する



入力用の変数をここに入れ

奇数を表す数式
中学数学で学んだ内容を利用している

(4) 高さ n 段のピラミッドを作ってみよう
高さを 10 に設定したときのピラミッド



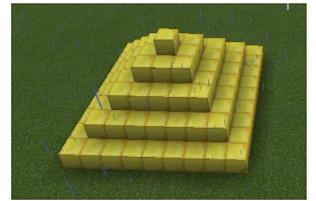
何を何回繰り返すのか 規則性を考えまたピラミッドの対称性を利用しよう

(1) 図のような高さ5段のピラミッドを作るには、1番下の段は何個のブロックが必要だろうか
9個

(2) 1段上がるごとに何個ずつブロックが減っていくだろうか . . . 2個

(3) 規則性を見つけよう

高さn段のピラミッドをつるには、1番下の段は何個のブロックが必要だろうか . . . $2n - 1$ 個



対称性注目して、何を何回繰り返すプログラムを作ればよいであろうか。

```
チャットコマンド "pyra" を入力した時 takasa
変数 n を takasa にする
エージェントを上 に 1 ブロック移動させる
くりかえし takasa - 1 回
くりかえし 4 回
  くりかえし 2 x n - 1 回
    エージェントを前 に 1 ブロック移動させる
    エージェントに下 へ 置かせる
  エージェントの向きを左 に変える
  エージェントを上 に 1 ブロック移動させる
  エージェントを前 に 1 ブロック移動させる
  エージェントを左 に 1 ブロック移動させる
  変数 n を -1 だけ増やす
  エージェントに下 へ 置かせる
```