

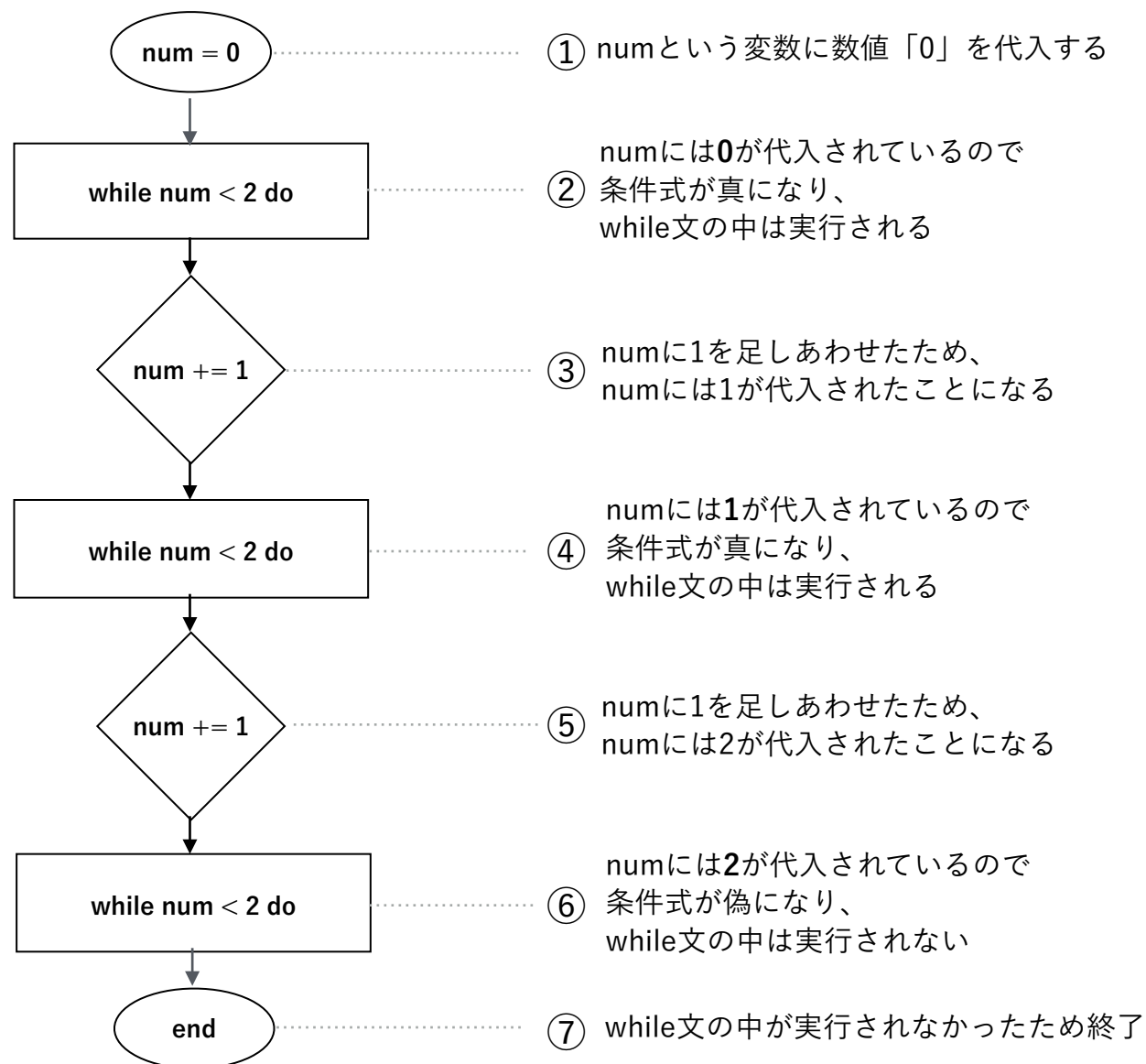
# Ruby要点チェック

## While文・・・条件式がtrueの間、処理を繰り返す

【例】 numが2以下である限り、 numに1を加算する処理を繰り返すwhile文

sample.rb

```
num = 0           ①
while num < 2 do  ② ④ ⑥
  num += 1        ③ ⑤
end               ⑦
```



# Ruby要点チェック

## 関数・・・複数の処理をまとめて、名前をつけることが出来る

【例】 二行の自己紹介の文字列を出力する関数my\_introduction

```
sample.rb
def my_introduction # 関数の定義 ②
  puts "私の名前はTECH 太郎です" ③
  puts "宜しくお願いします!" ④
end
my_introduction # 実行 ①
```

- ① def ~ endと記述されている部分は関数定義のため実行されない。my\_introductionが呼び出されたとき初めて実行される
- ② 関数を実行する。
- ③ 「私の名前はTECH 太郎です」が出力される
- ④ 「宜しくお願いします!」が出力される

## 引数・・・関数の外にある値を関数に渡すことが出来る

【例】 任意の文字列を渡すと文字列末尾に"ジュース"つける関数mixer

```
sample.rb
def mixer(chosen_fruit) ③
  "#{ chosen_fruit }ジュース" ④
end
fruit = "バナナ" ①
mixer(fruit) => "バナナジュース" ② ⑤
```

- ① def ~ endと記述されている関数は呼び出されるまで実行されないため、現時点では fruit = "バナナ" のみが実行される
- ② 関数を実行する。変数fruitに入っている"バナナ"が引数の値として渡される
- ③ chosen\_fruitが②で渡された値"バナナ"を受け取る。chosen\_fruitは、関数mixer内でのみ利用できる。①で定義したfruitとは無関係。
- ④ 関数の中で最後の実行結果"バナナジュース"が返り値となる。  
※returnをつけると明示的に返り値を指定できる。
- ⑤ mixer(fruit)が"バナナジュース"に置き換わる