

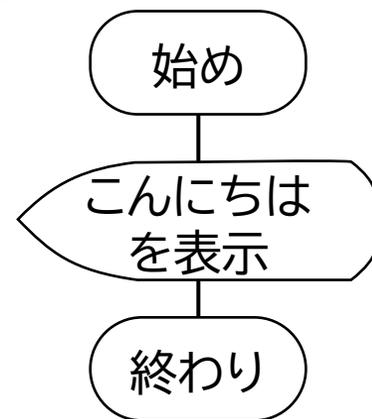
# プログラミング練習問題

# 学習の仕方

- プログラミング環境を用意
  - PC室ならIDLEを起動
  - Surface等なら
    - Googleコラボトリー(おすすめ)  
<https://colab.research.google.com>
- できるまで先に進まない
  - 正しく直せば正しく動く
  - エラーメッセージをしっかりと読もう(検索してもよい)
  - エラーを修正するのがプログラミングの重要ポイント

# 01 画面に表示する1

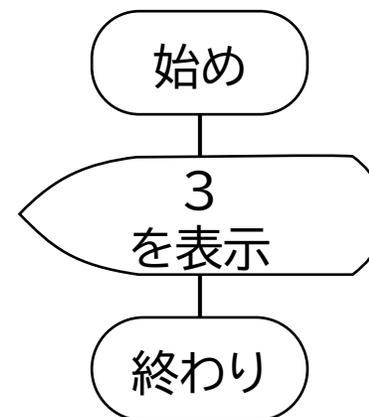
- 以下の条件を満たすプログラムを作成せよ
  - こんにちは と画面に表示



実行結果  
こんにちは

## 02 画面に表示する2

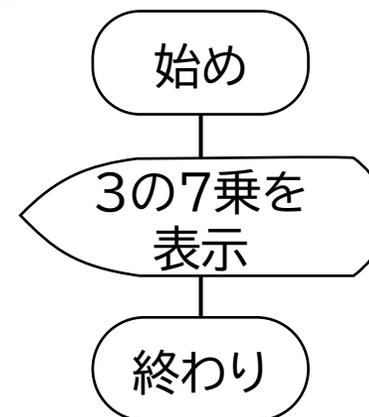
- 以下の条件を満たすプログラムを作成せよ
  - 3 という数値を画面に表示



実行結果  
3

# 03 画面に表示する3

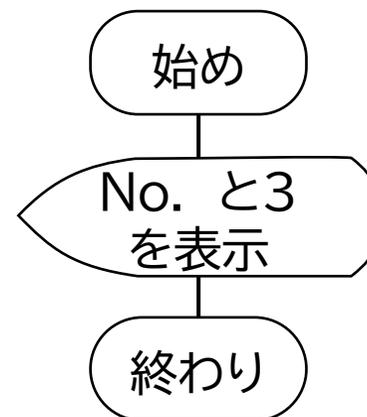
- 以下の条件を満たすプログラムを作成せよ
  - 3の7乗の計算結果を画面に表示



実行結果  
2187

# 04 画面に表示する4

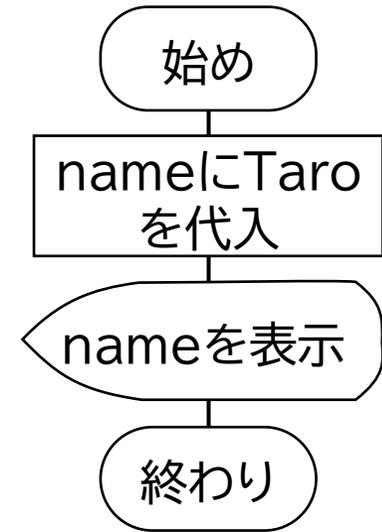
- 以下の条件を満たすプログラムを作成せよ
  - No. 3 と画面に表示



実行結果  
No. 3

# 05 変数を使う1

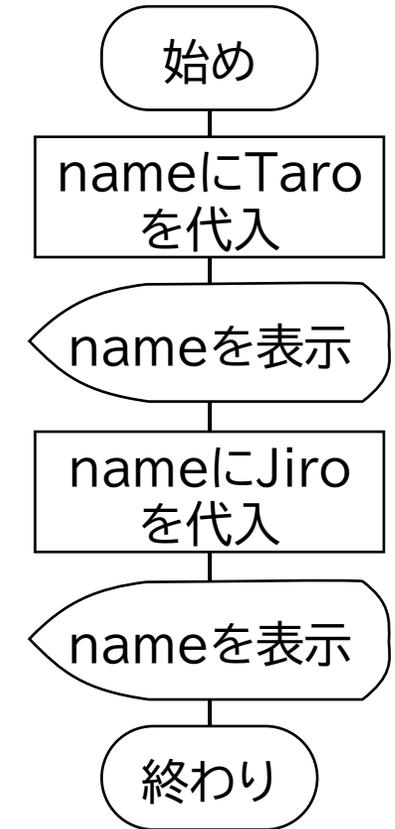
- 以下の条件を満たすプログラムを作成せよ
  - 変数 name に Taro を代入
  - 変数 name の データを画面に表示



実行結果  
Taro

# 06 変数を使う2

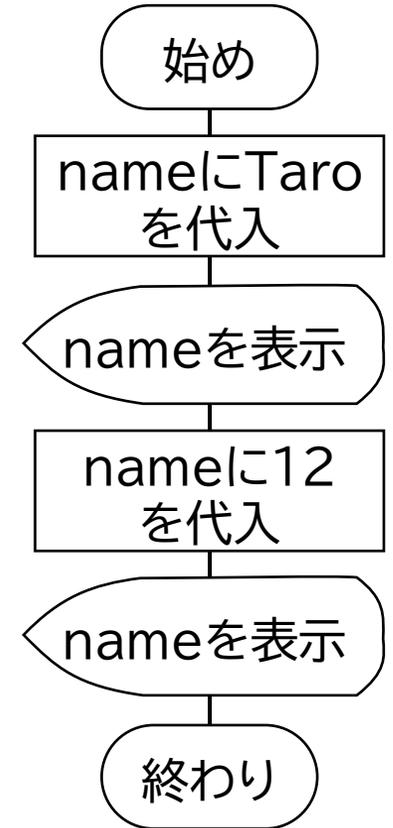
- 以下の条件を満たすプログラムを作成せよ
  - 変数 name に Taro を代入
  - 変数 name の データを画面に表示
  - 変数 name に Jiro を代入
  - 変数 name の データを画面に表示



実行結果  
Taro  
Jiro

# 07 変数を使う

- 以下の条件を満たすプログラムを作成せよ
  - 変数 name に Taro を代入
  - 変数 name の データを画面に表示
  - 変数 name に数値 12 を代入
  - 変数 name の データを画面に表示

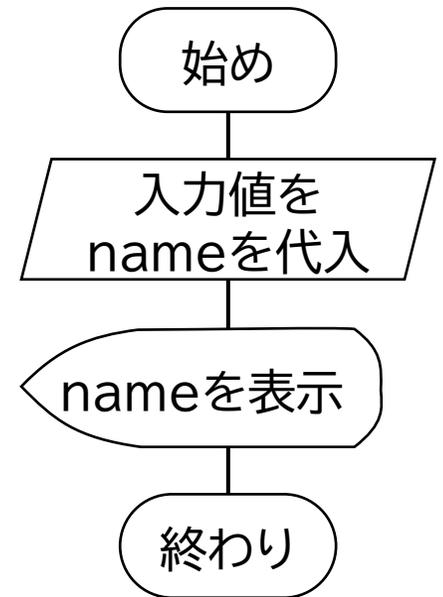


実行結果

Taro  
12

# 08 文字を入力する

- 以下の条件を満たすプログラムを作成せよ
  - name?と画面に表示し、  
入力された値を変数nameに代入
  - 変数nameのデータを画面に表示

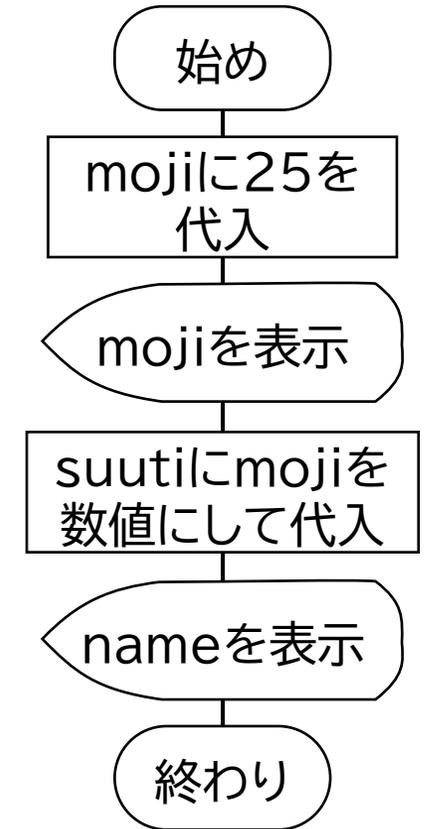


- 自分の名前を入力して表示されるか確認する

実行結果  
自分の名前

# 09 文字列の数を数値にする1

- 以下の条件を満たすプログラムを作成せよ
  - 変数mojiに 25 という文字を代入
  - mojiのデータを画面に表示
  - 変数mojiのデータを数値に変換し変数suutiに代入
  - suutiのデータを画面に表示

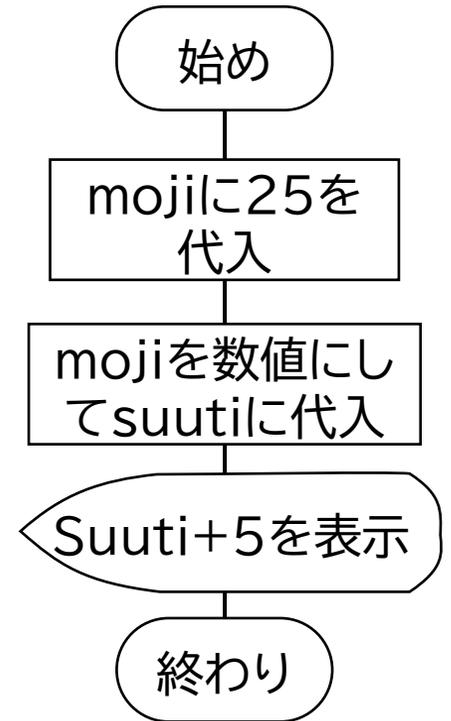


実行結果

25  
25

# 10文字列の数を数値にする2

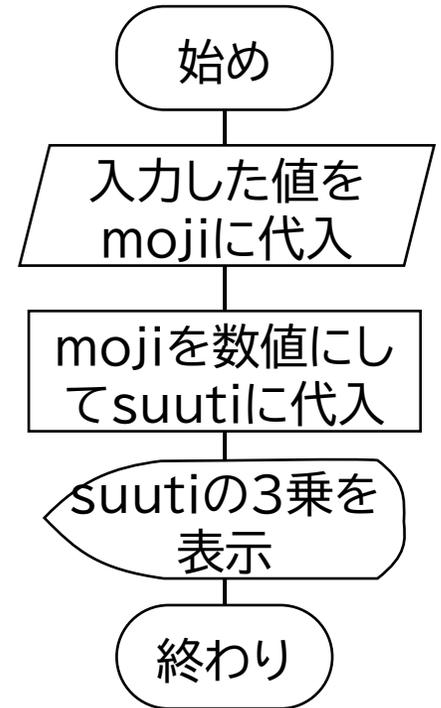
- 以下の条件を満たすプログラムを作成せよ
  - 変数mojiに 25 という文字を代入
  - 変数mojiのデータを数値に変換し  
変数suutiに代入
  - 変数suutiに5を加えたデータを画面に表示



実行結果  
30

# 11 文字列の数を数値にする3

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値を変数mojiに代入
  - 変数mojiのデータを数値に変換し変数suutiに代入
  - suutiの3乗を画面に表示



- 入力する数値を変えて数回実行する

実行結果

Number? 5

125

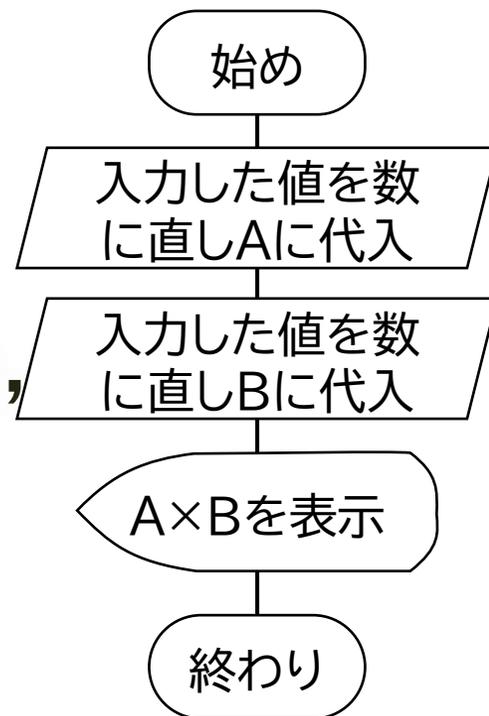
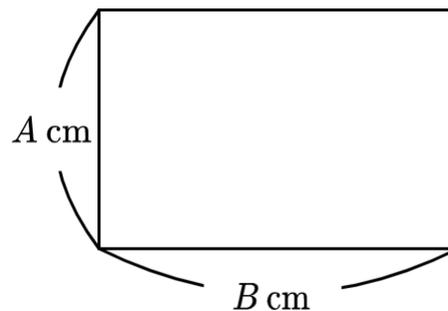
Number? 11

1331

入力して  
[Enter]

# 12 文字列の数を数値にする

- 以下の条件を満たすプログラムを作成せよ
  - 整数  $A$ ,  $B$  が与えられる. 縦の辺の長さが  $A$  cm, 横の辺の長さが  $B$  cmである下図のような長方形の面積は何 $\text{cm}^2$  か求めよ.
  - 問題文を書き直すと
    - 入力された数値を変数 $A$ に代入
    - 入力された数値を変数 $B$ に代入
    - $A$ と $B$ の積を画面に表示



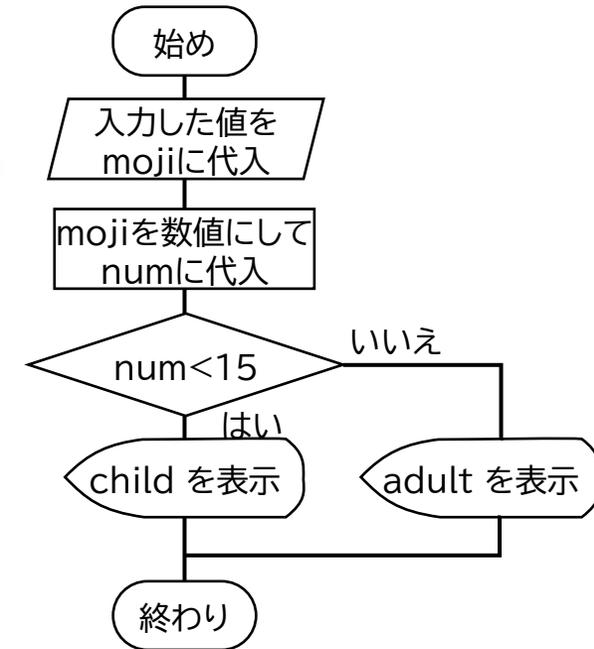
- 入力する数値を変えて数回実行する

入力して  
[Enter]

実行結果  
2  
3  
6

# 13 条件によって違う処理をする1

- 以下の条件を満たすプログラムを作成せよ
  - age?と画面に表示し、入力された値を変数mojiに代入
  - 変数mojiのデータを数値に変換し変数numに代入
  - もし 変数numが15未満なら画面に child と表示
  - そうでなければ画面に adult と表示
- 入力する数値を変えて数回実行する



実行結果  
age? 15  
adult

入力して  
[Enter]

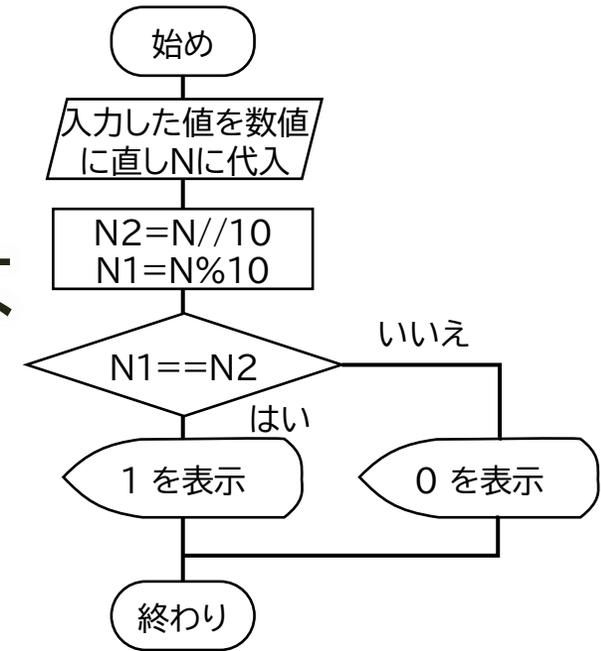
# 14 条件によって違う処理をする1

• 以下の条件を満たすプログラムを作成せよ

- 十進法で 2 桁の整数Nが与えられる。  
N の十の位の数字と一の位の数字が  
同じである場合は 1 を, そうでない場合は  
0 を出力せよ。

- 問題文を書き直すと(ゾロ目なら1)

- 入力された2桁の数を数値に変換して変数Nに代入
- Nの10の位を変数N2に代入する
- Nの1の位を変数N1に代入する
- もし、N1とN2が等しければ、画面に 1 と表示
- そうでなければ、画面に 0 と表示



実行結果

15  
0

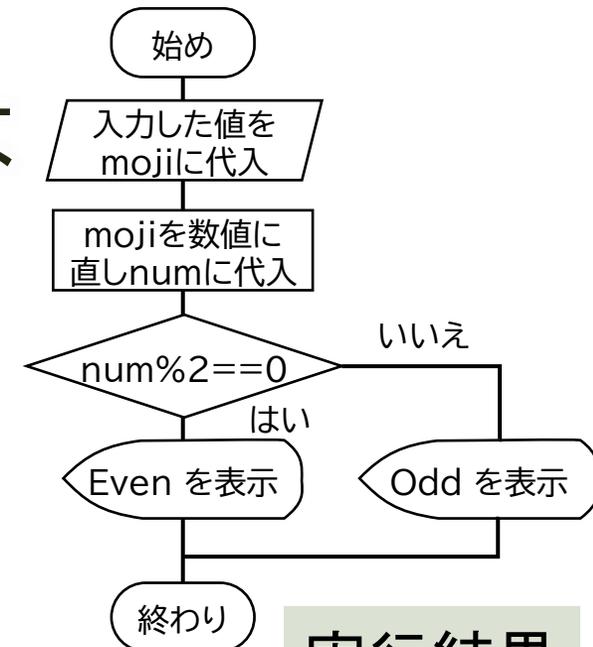
入力して  
[Enter]

実行結果

33  
1

# 15 if文:条件によって違う処理をする2

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値を変数mojiに代入
  - 変数mojiのデータを数値に変換し変数numに代入
  - もしnumが偶数ならEvenと表示そうでなければOddと表示



実行結果

16  
Even

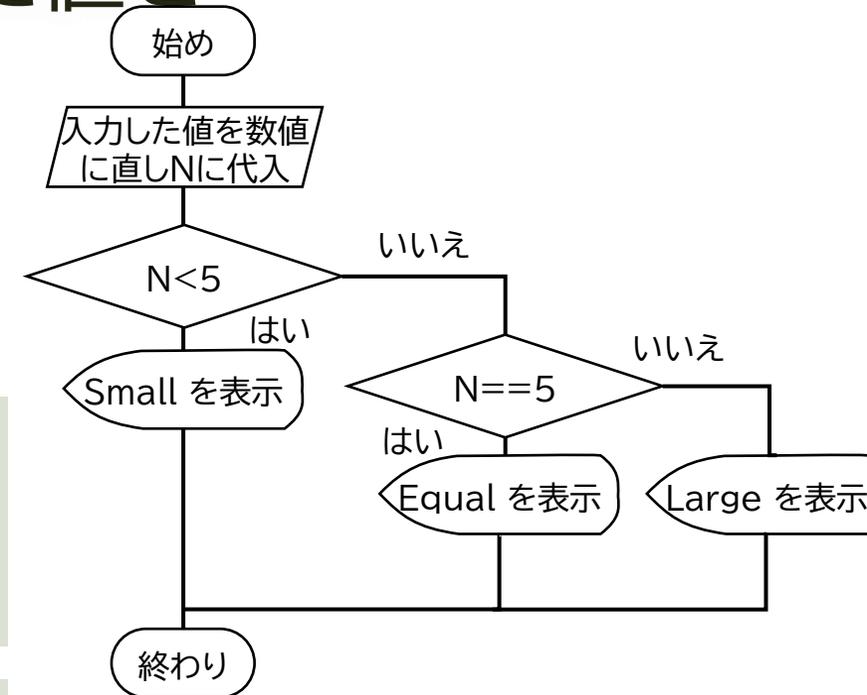
入力して  
[Enter]

実行結果

33  
Odd

# 16 if文:条件によって違う処理をする2

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値を数値に直し変数Nに代入
  - Nが5より小さければSmall  
5と等しければEqual  
そうでなければLarge と表示



実行結果

6  
Large

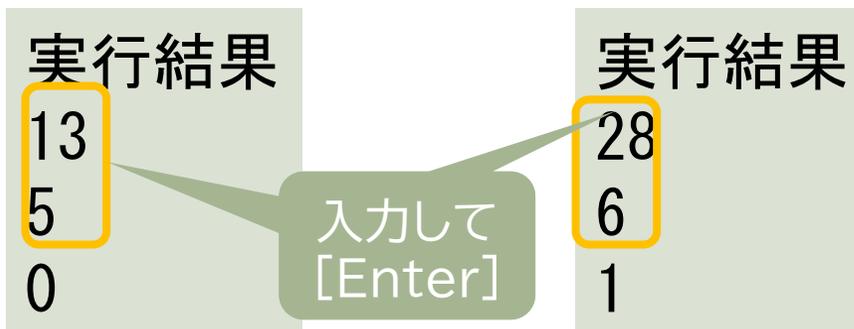
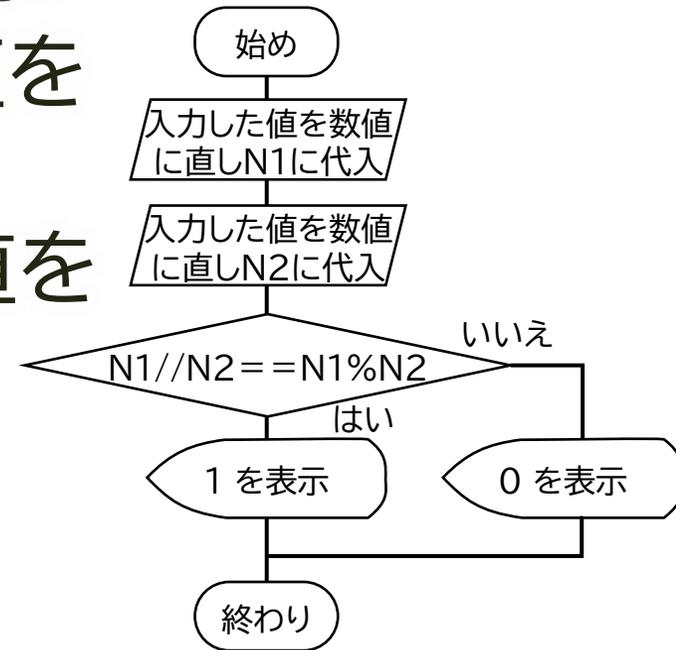
入力して [Enter]

実行結果

3  
Small

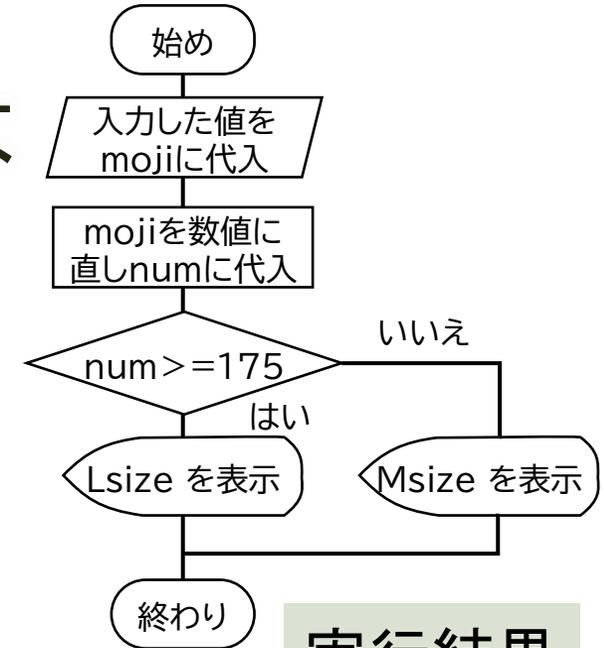
# 17 if文:条件によって違う処理をする2

- 以下の条件を満たすプログラムを作成せよ
  - Number1?と画面に表示し、入力された値を数値に直し変数N1に代入
  - Number2?と画面に表示し、入力された値を数値に直し変数N2に代入
  - N1をN2で割った商と余りが等しければ1そうでなければ0を表示



# 18 if文:条件によって違う処理をする

- 以下の条件を満たすプログラムを作成せよ
  - 数値を入力させ、  
175以上ならLsize、それ未満ならMsize  
と表示



実行結果

160

Msize

入力して  
[Enter]

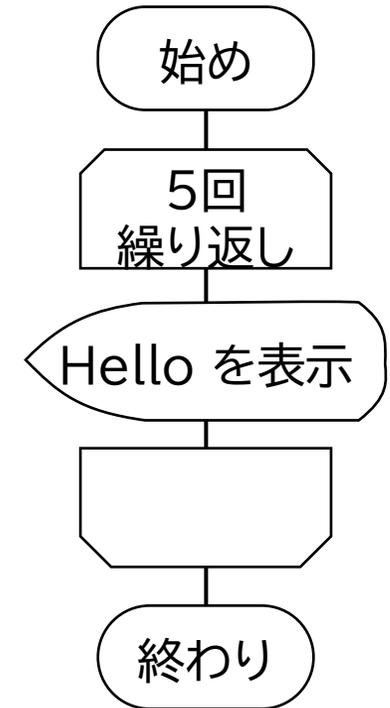
実行結果

175

Lsize

# 19 for文:繰り返して実行する1

- 以下の条件を満たすプログラムを作成せよ
  - Helloと5回表示
  - プログラムは2行で書く

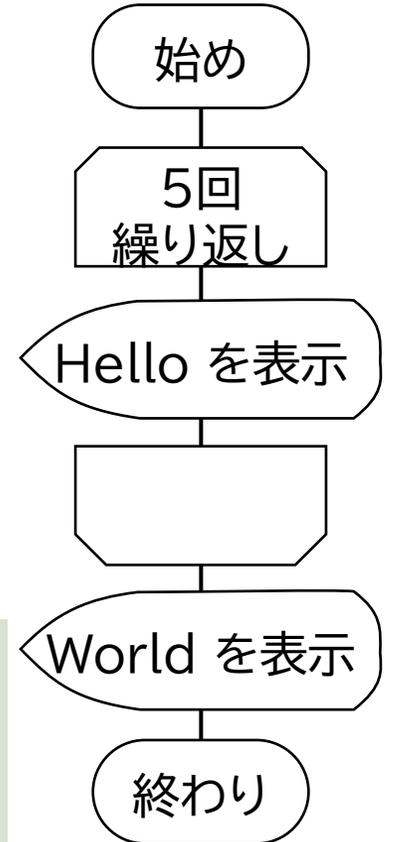


実行結果

```
Hello  
Hello  
Hello  
Hello  
Hello
```

# 20 for文:繰り返して実行する1

- 以下の条件を満たすプログラムを作成せよ
  - Helloと5回表示
  - Worldと1回表示
  - プログラムは3行で書く

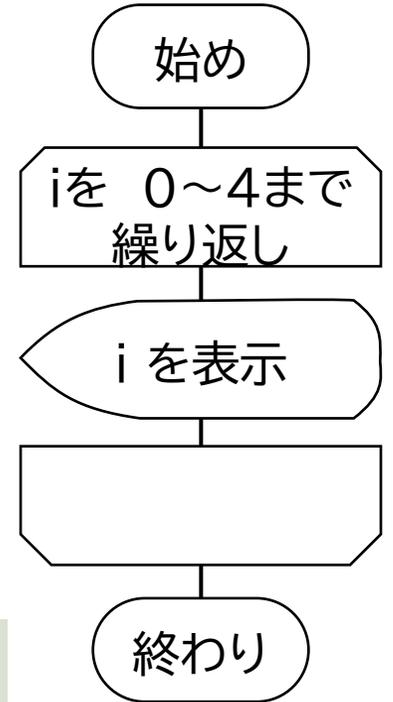


## 実行結果

```
Hello  
Hello  
Hello  
Hello  
Hello  
World
```

# 21 for文:繰り返して実行する2

- 以下の条件を満たすプログラムを作成せよ
  - 0から4まで順に表示
  - プログラムはforを使って2行で書く

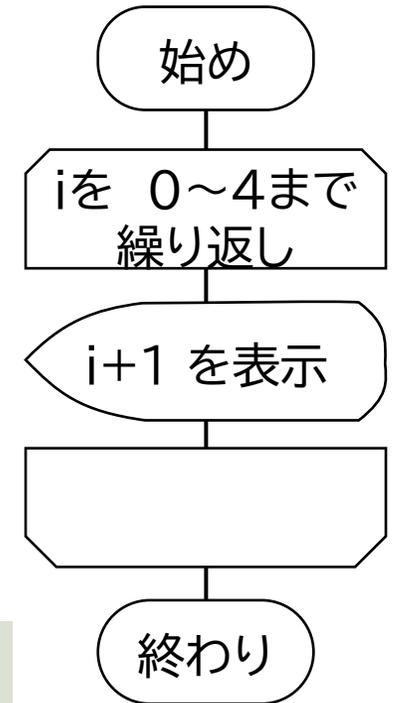


実行結果

0  
1  
2  
3  
4

# 22 for文:繰り返して実行する2

- 以下の条件を満たすプログラムを作成せよ
  - 1から5まで順に表示
  - プログラムはforを使って2行で書く

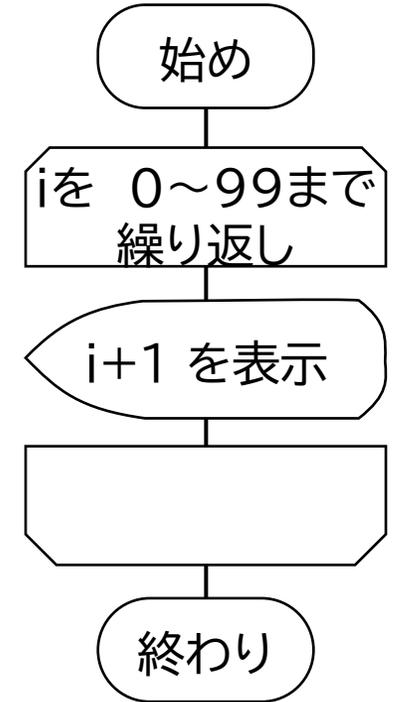


実行結果

1  
2  
3  
4  
5

# 23 for文:繰り返して実行する

- 以下の条件を満たすプログラムを作成せよ
  - 1から100まで順に表示
  - プログラムはforを使って2行で書く



実行結果

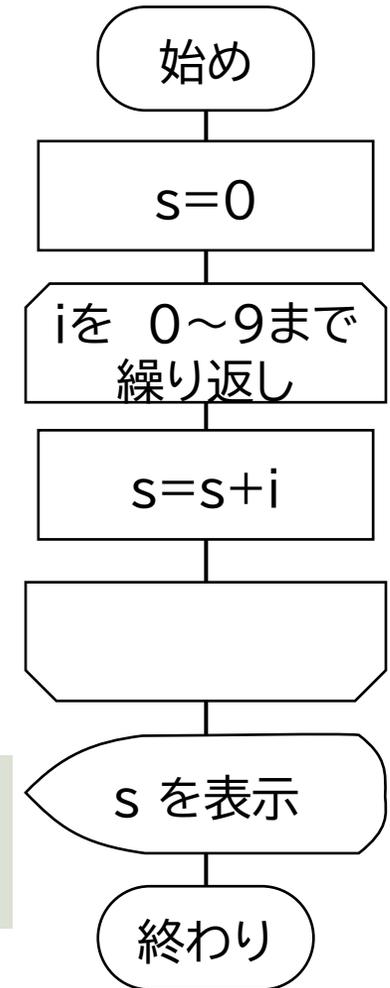
1  
2

98  
99  
100

# 24 for文:繰り返して実行する3

- 以下の条件を満たすプログラムを作成せよ
  - 0から9までの和を表示
  - プログラムはforを使って4行で書く
- 考え方
  - 合計のための変数sを用意
  - forを使って0から9まで順に加算

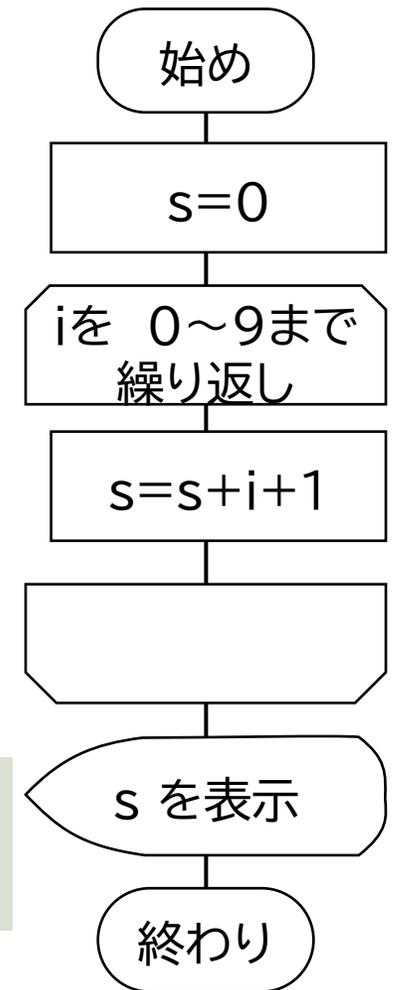
実行結果  
45



# 25 for文:繰り返して実行する3

- 以下の条件を満たすプログラムを作成せよ
  - 1から10までの和を表示
  - プログラムはforを使って4行で書く

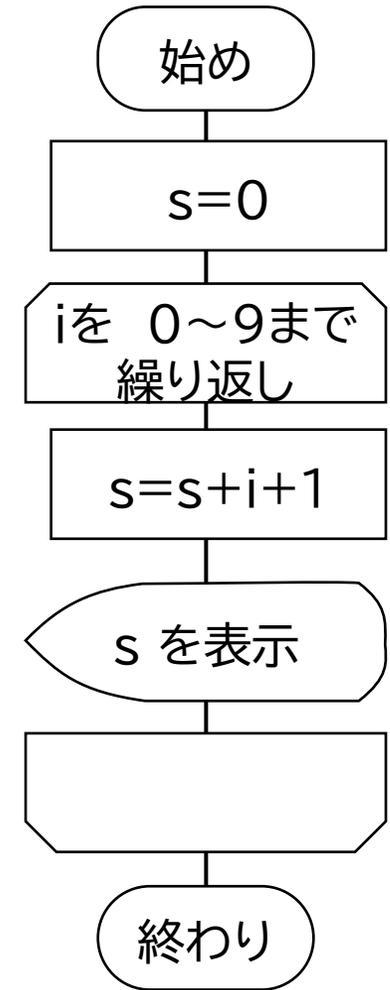
実行結果  
45



# 26 for文:繰り返して実行する3

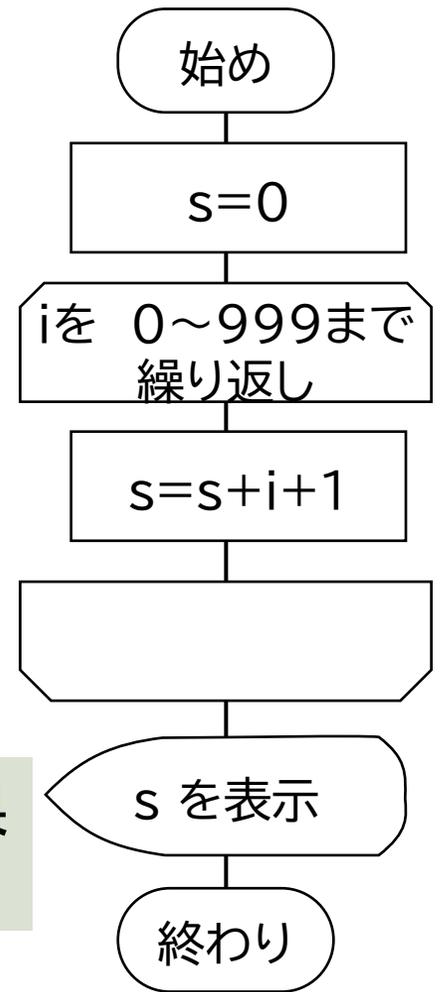
- 以下の条件を満たすプログラムを作成せよ
  - 1から10まで順に加え表示する
  - プログラムはforを使って4行で書く

実行結果  
1  
3  
6  
10  
15  
21  
28  
36  
45  
55



# 27 for文:繰り返して実行する

- 以下の条件を満たすプログラムを作成せよ
  - 1~1000までの数の和を表示



実行結果  
500500

# 28 制御構造の組み合わせ1

- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの偶数を表示

実行結果

2

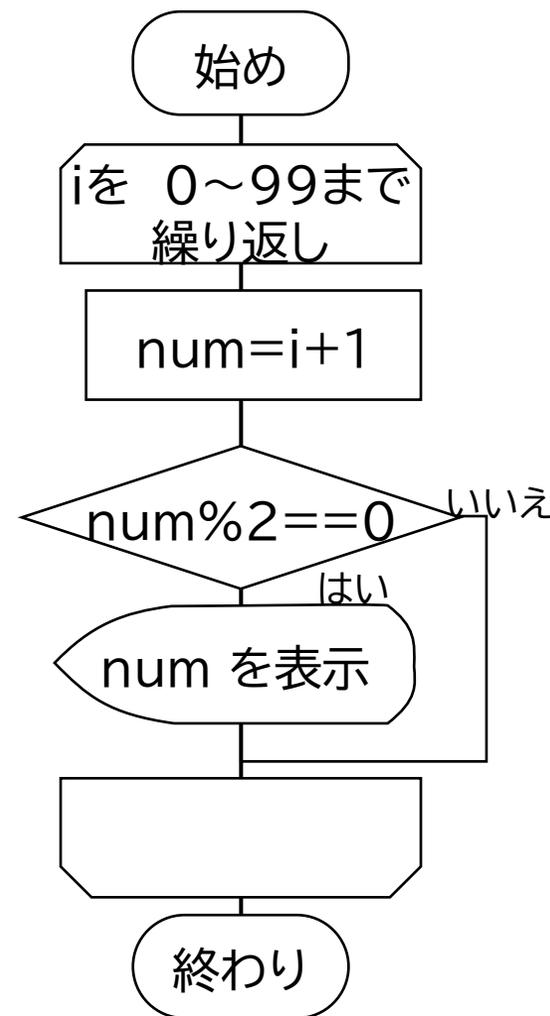
4

6

96

98

100



# 29 制御構造の組み合わせ1

- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの3の倍数を表示

実行結果

3

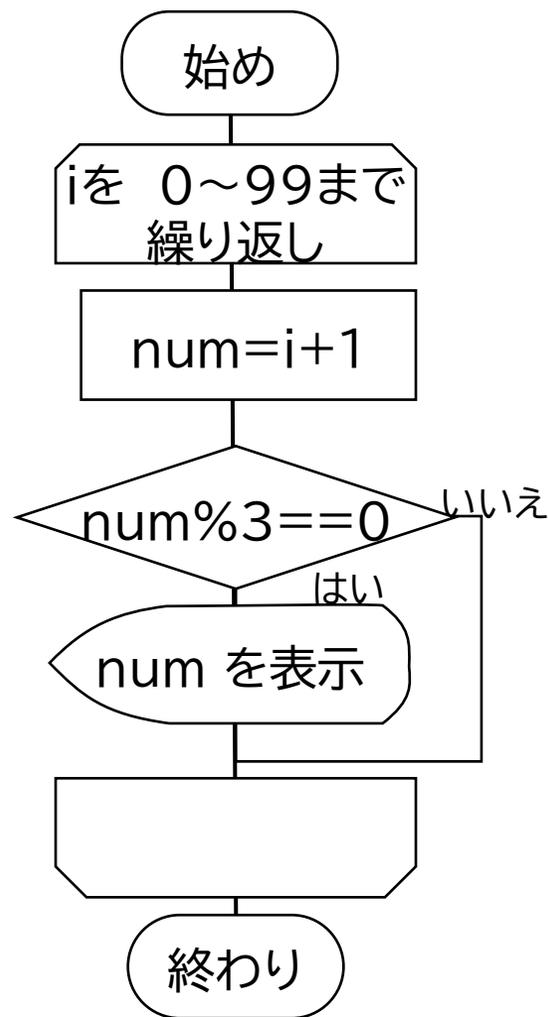
6

9

93

96

99

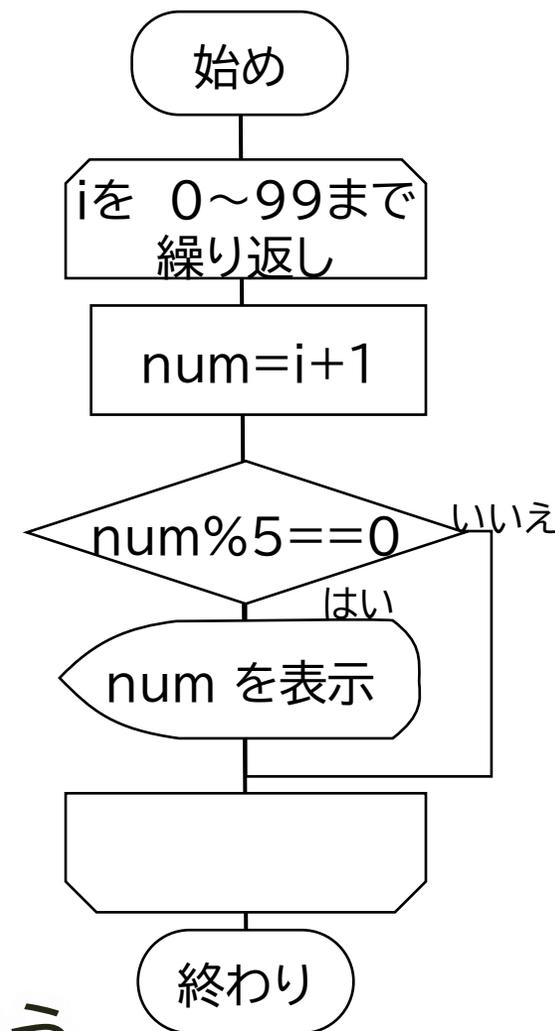


# 30 制御構造の組み合わせ

- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの5の倍数を表示

実行結果

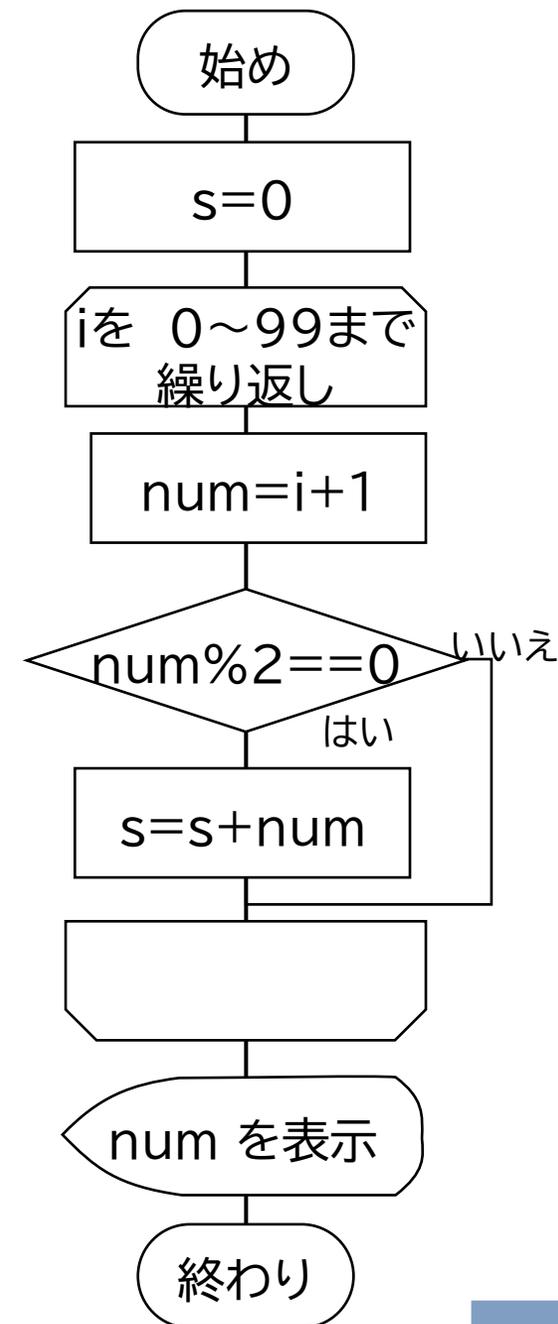
```
5
10
15
...
90
95
100
```



- 右のフローチャートと違う方法でも解いてみよう

# 31 制御構造の組み合わせ2

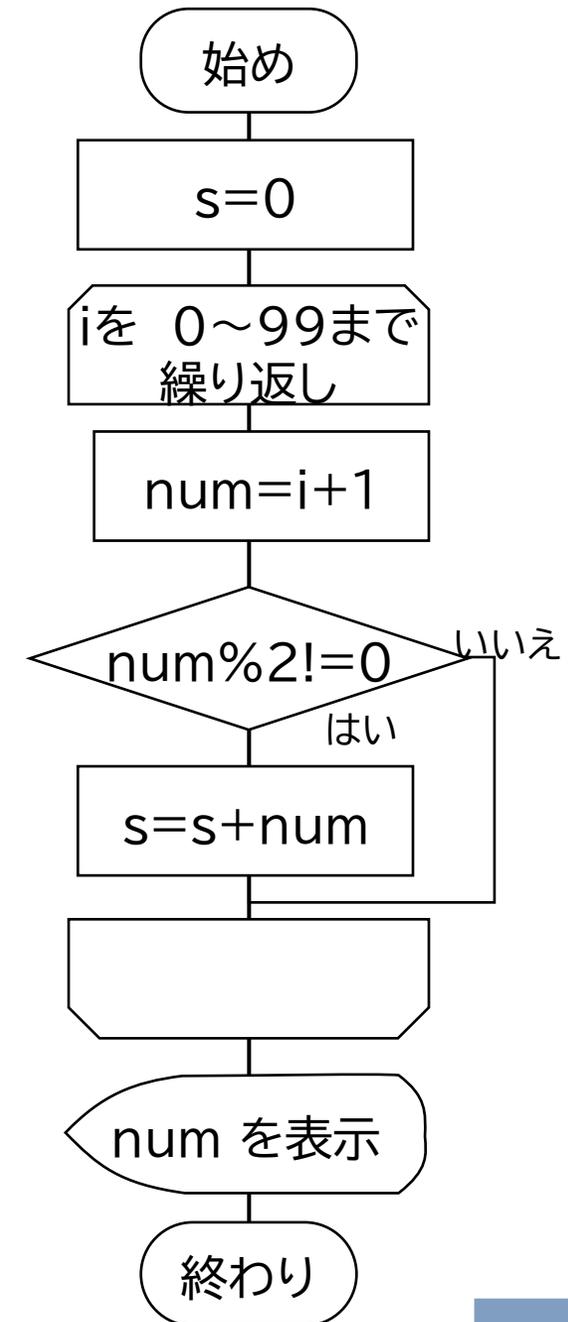
- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの偶数の和を表示



実行結果  
2550

## 32 制御構造の組み合わせ2

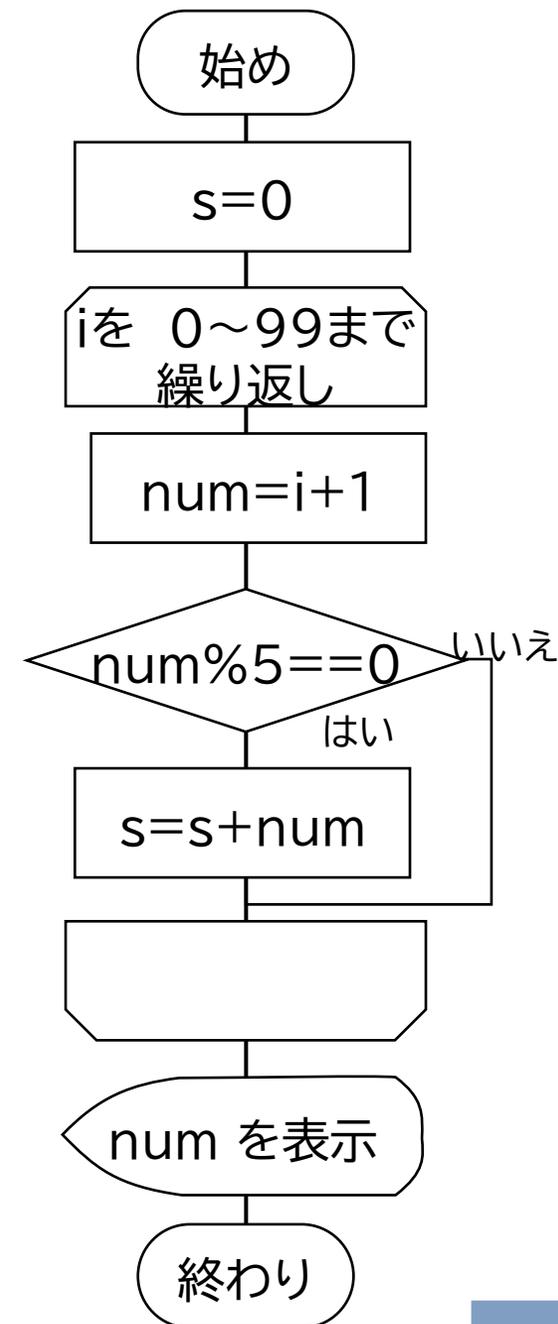
- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの奇数の和を表示



実行結果  
2500

# 33 制御構造の組み合わせ2

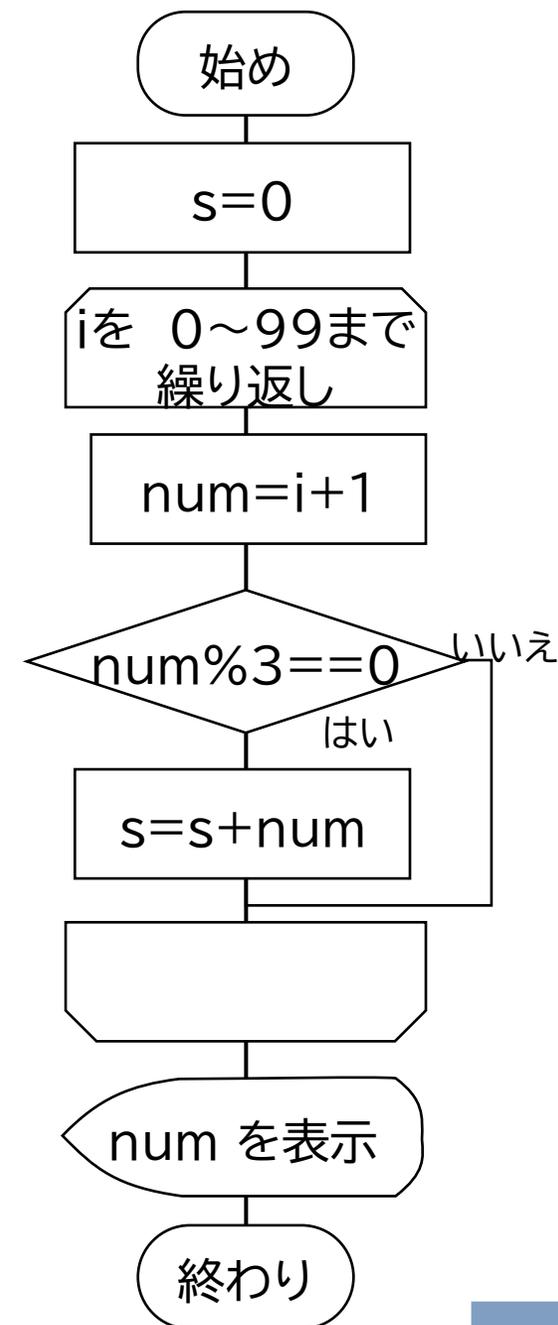
- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの5の倍数の和を表示



実行結果  
1050

# 34 制御構造の組み合わせ

- 以下の条件を満たすプログラムを作成せよ
  - 1~100までの3の倍数の和を表示

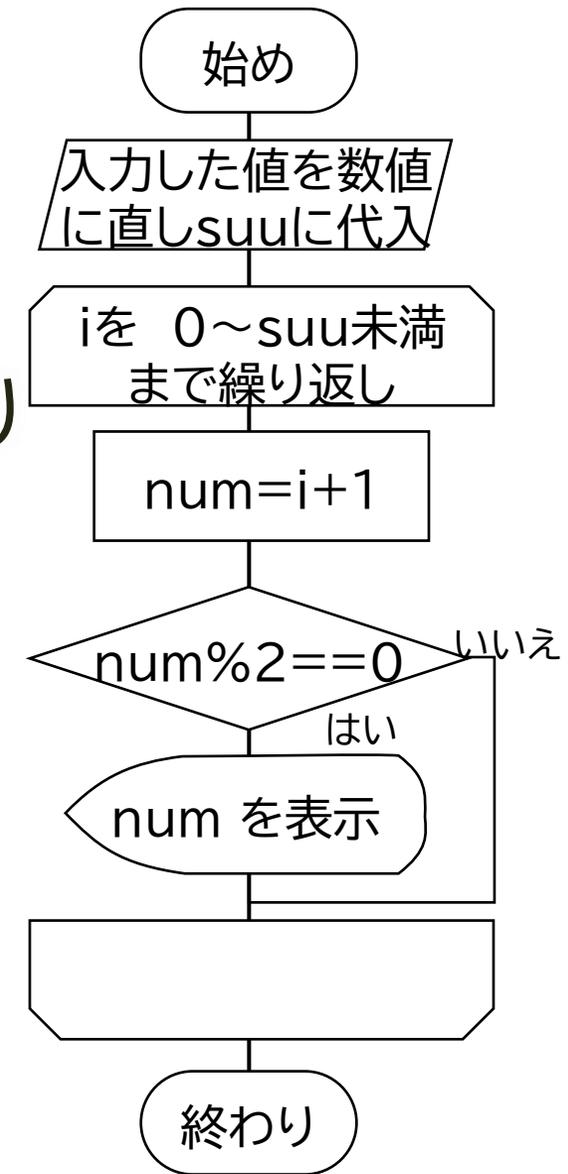


実行結果  
1683

# 35 入力に対応したプログラム

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値より小さい偶数をすべて表示

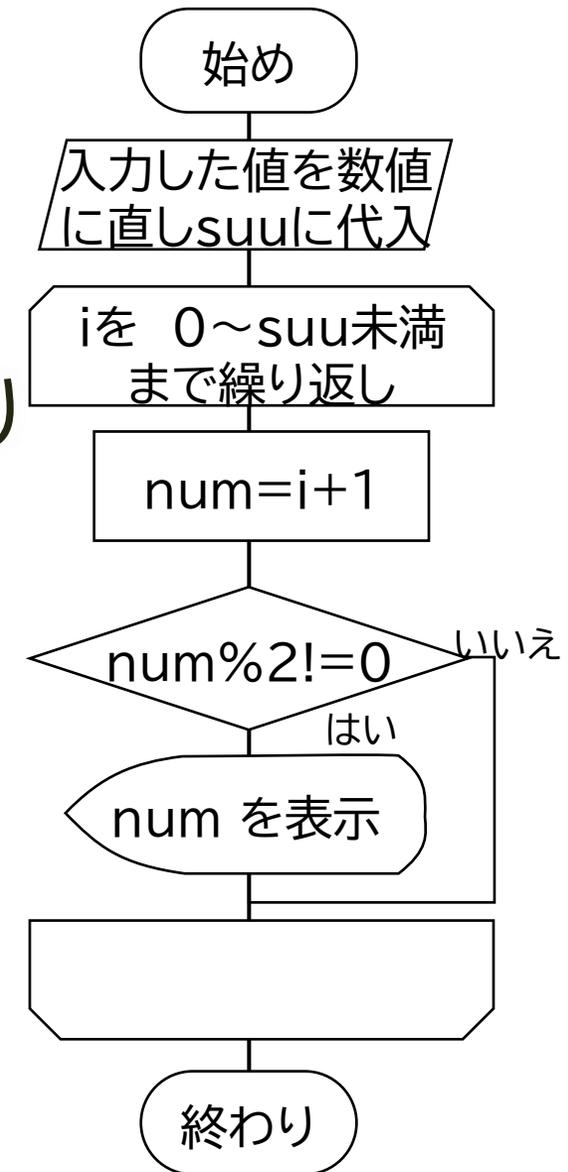
実行結果  
Number?19  
2  
4  
6  
8  
10  
12  
14  
16  
18



# 36 入力に対応したプログラム

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値より小さい奇数をすべて表示

実行結果  
Number?19  
2  
4  
6  
8  
10  
12  
14  
16  
18



# 37 入力に対応したプログラム

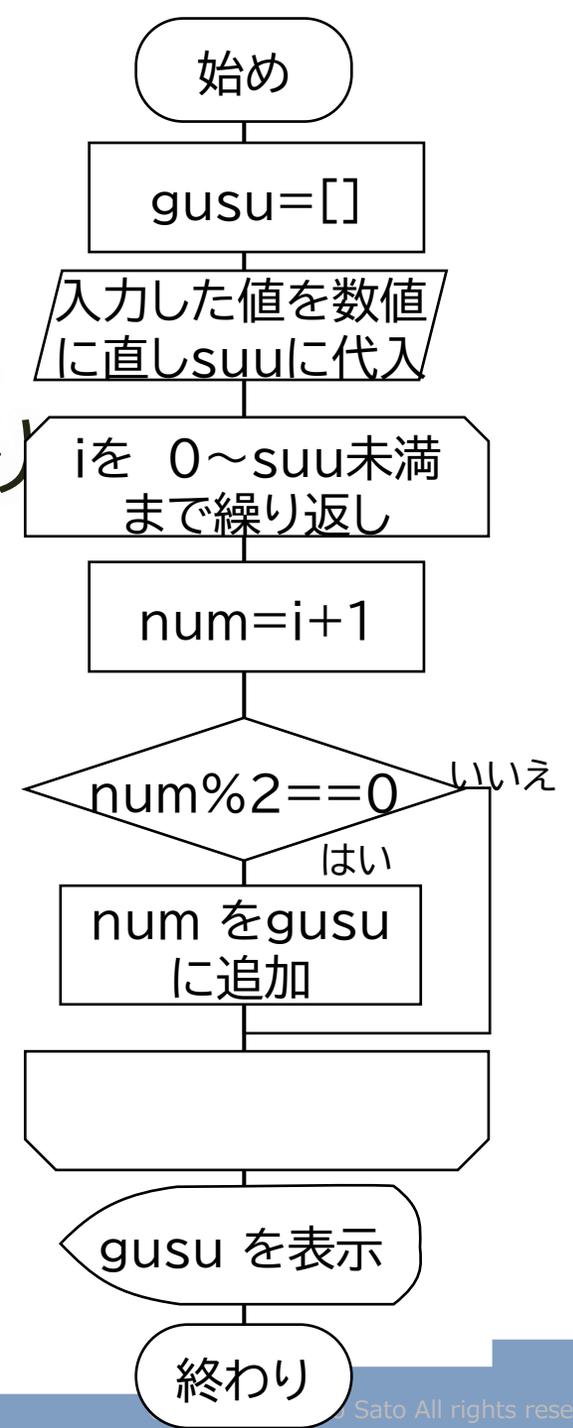
- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、入力された値より小さい偶数をリストにしてまとめて表示

実行結果

Number? 25

[2, 4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24]

入力して  
[Enter]

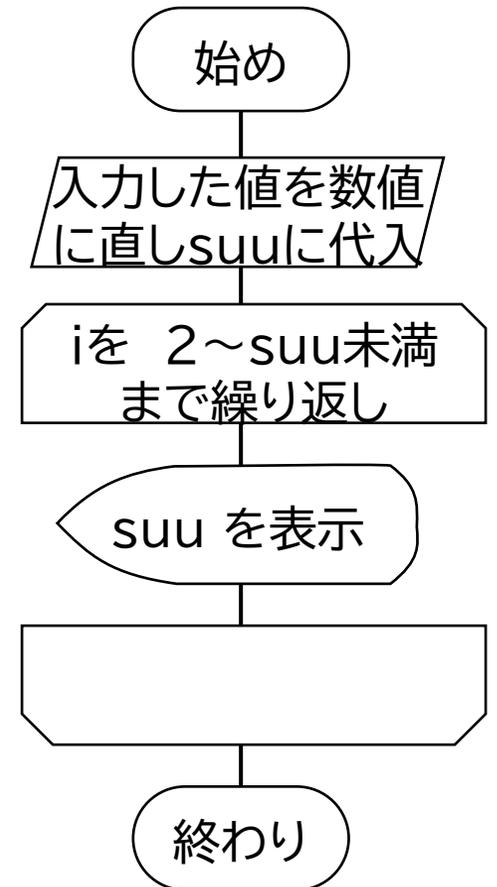


# 38 range関数の使い方

- 以下の条件を満たすプログラムを作成せよ
  - Number?と画面に表示し、  
2から入力された値未満の数を表示

```
実行結果
Number ?8
2
3
4
5
6
7
```

入力して  
[Enter]



# 39 制御構造の組み合わせ3

• 以下の条件を満たすプログラムを作成せよ

- Number?と画面に表示し、  
入力された数が素数か判定する
- 擬似言語で書くと

変数 `suu` に入力された数を代入

変数 `sosu` に `True`(真)を代入

`i`を2から`suu`未満まで変えて繰り返し:

`suu`を`i`で割った余りが0なら:

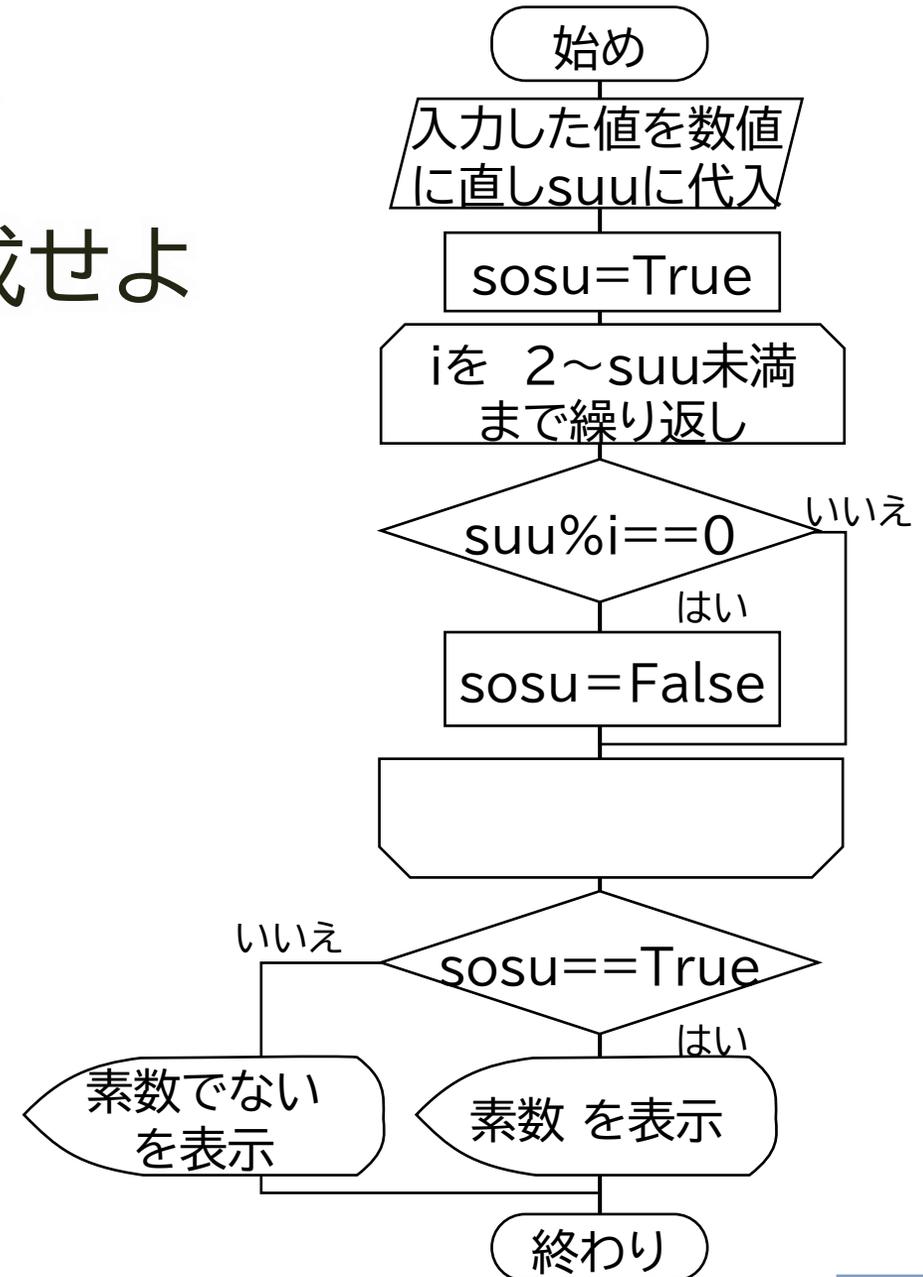
変数 `sosu` に`False`(偽)を代入

`sosu`が真なら:

`suu`は素数と表示

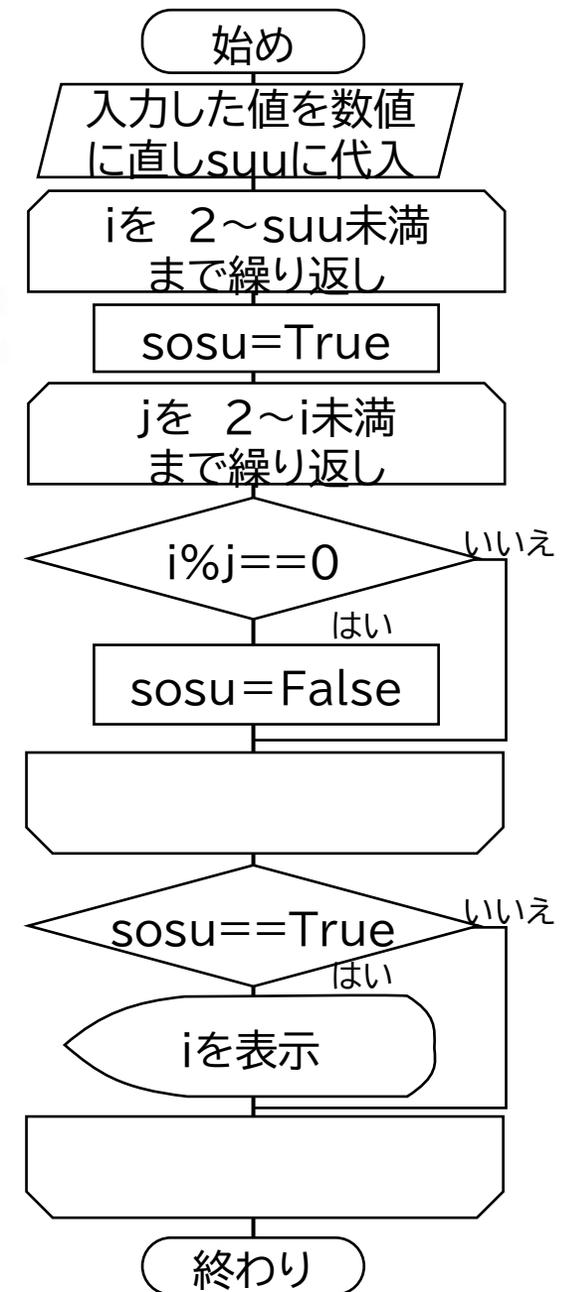
そうでなければ:

`suu`は素数でないと表示



# 40 制御構造の組み合わせ3

- 以下の条件を満たすプログラムを作成せよ
  - 入力された数までの素数を表示する
  - ヒント
    - フローチャートを参考に
    - for文を2回使う



# 41 制御構造の組み合わせ

- 以下の条件を満たすプログラムを作成せよ
  - 入力された数より小さい最大の素数を表示

