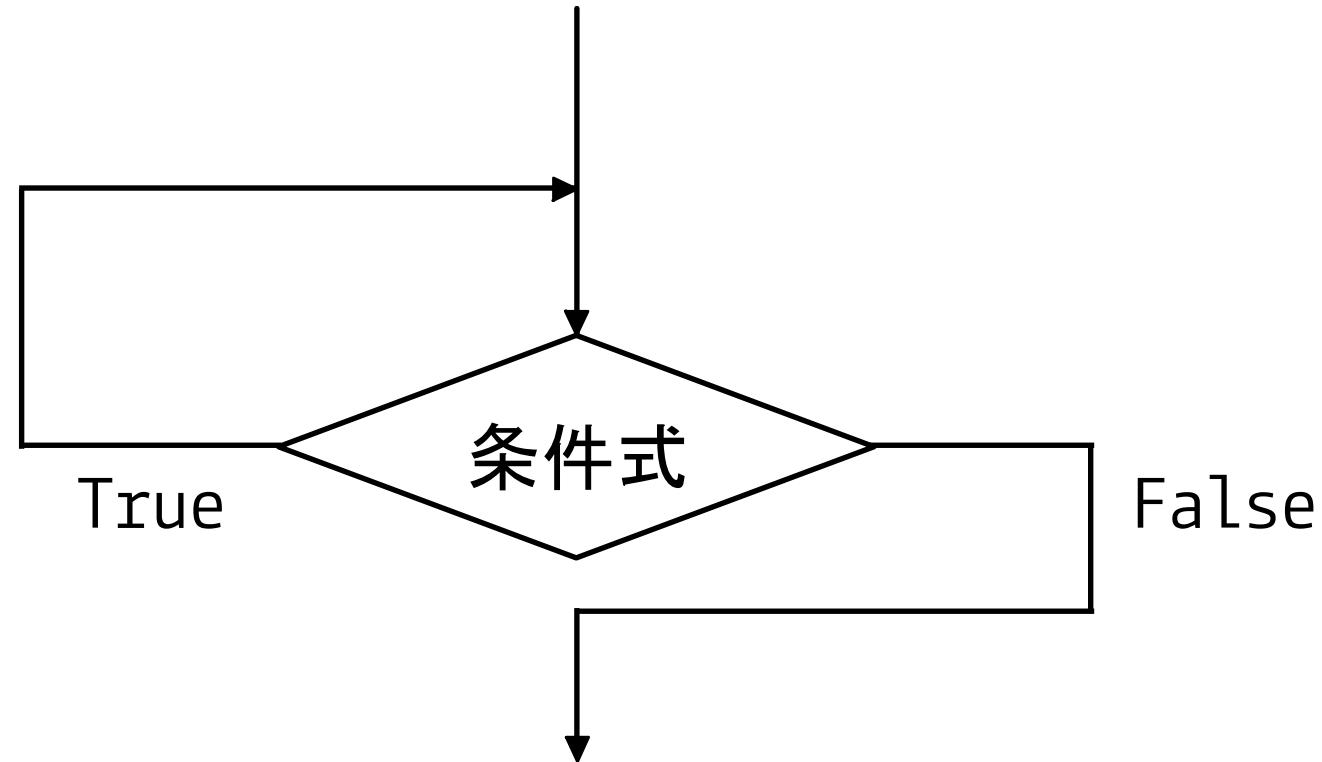


繰り返し処理



繰り返し文

- while文かfor文を使用することで、ブロックにあるプログラムを繰り返し行うことができる



while文

- while文を使用することで、ブロックにあるプログラムを繰り返すことができる
 - while文の条件式がTrueの間、ブロックにあるコードが実行される

```
while(条件式):
    # Trueのときの処理
```

while文の書き方
コロン(:)を忘れずに書く

```
i = 0
while(i < 10):
    if i % 2 == 0:
        print(i)
    i += 1
```

0から9の偶数を出力する
プログラム例

whileによる無限ループとbreak文

- while文の条件式にTrueを与えると無限ループになる
- 無限ループがあるプログラムを実行すると終わらないため、break文を使用することで、繰り返し処理を抜けることができる
- 無限ループに陥るプログラムを実行した場合、「ctrlキー」を押しながら「cキー」を押すことで、プログラムを強制終了できる
 - 「ctrlキー」を押しながら「cキー」を押す操作をctrl-cと表記できる

```
while(True):
    # 無限ループ
    break
```

continue文とbreak文の違い

- 繰り返し処理の中で, continue文が使用できる
 - continue文が実行されると, 即座に繰り返し処理の先頭に戻り, ループ条件式を再評価する

```
i = 0
while(True):
    if i > 2:
        break
    print(i)
    i += 1
```

break文の使用例
(0, 1, 2 と出力)

```
i = 0
while(i < 10):
    if i % 2 == 0:
        i += 1
        continue
    print(i)
    i += 1
```

continue文の使用例
(1, 3, 5, 7, 9 と出力)

for文

- 一定の回数だけコードのブロックを実行したいときには、for文とrange関数を使用する
- range(n)は0, 1, 2, 3, ..., n - 1を表す

```
for 変数 in range関数:
    # 繰り返し処理
```

for文の書き方
コロン(:)を忘れずに書く

```
for i in range(3):
    print(i)
```

0, 1, 2と出力するプログラム例
(i を 0, 1, 2と繰り返す)

range関数

- range関数は、3つまで引数を渡すことができる
- range関数に2つの引数を渡した場合、第1引数が初期値、第2引数がループ終了値を越えない最大値となる
 - `range(m, n)`は`m, m + 1, ..., n - 2, n - 1`を表す

```
for i in range(3, 7):
    print(i)
```

`i` を 3, 4, 5, 6と繰り返す
プログラム例

range関数

- range関数は、3つまで引数を渡すことができる
- for文の際に、range関数に3つの引数を渡した場合、最初の2つの引数は、2つの引数と同様に、第1引数が初期値、第2引数がループ終了値を越えない最大値となる
- 第3引数は、繰り返しを行う際の変数の値を変更する更新量となる

```
for i in range(2, 11, 2):
    print(i)
```

i を 2, 4, 6, 8, 10と繰り返す
プログラム例

range関数

- 第3引数を負の値にすることで、繰り返しを行う際の変数の値を減らすことが可能になる
 - `range(n, 0, -1)`は, `n`, `n - 1`, `...`, `2`, `1`を表す

```
for i in range(5, 0, -1):
    print(i)
```

`i` を 5, 4, 3, 2, 1と繰り返す
プログラム例

ここまでのまとめ演習

- while文とfor文それぞれで, 1からキーボードに入力した正の整数値までの整数の和を計算するプログラムを作成しなさい
 - while文: q3-1.py
 - for文: q3-2.py
- 実行例
 - 10を入力すると, 55が出力する
 - 100を入力すると, 5050が出力する
 - 1000を入力すると, 500500が出力する

解答例

• While文

```
n = input()
i = 0
sum = 0
while(i <= n):
    sum += i
    i += 1

print(sum)
```

• for文

```
n = input()
sum = 0
for i in range(n + 1):
    sum += i
    i += 1

print(sum)
```