

Unit 3: 条件, if, while

Skill Builder 2: フラグを立てる

このレッスンでは、入力された数と数の合計を計算し、数の平均を決定することによって、カウンタとアキュムレータ(演算装置による演算結果を累積する、すなわち総和を得るというような計算に使うレジスタや変数のこと)について学習します。

目標

- カウンタステートメント
- アキュムレータステートメント
- フラグ付きのwhileループ
- 平均

一連の数値を数えて合計するプログラムを作成します。いくつ数字があるかは事前に分かりません。このレッスンでは、カウンタステートメント、アキュムレータステートメント、ループを終了するためのフラグ値、入力された数値の平均の計算を紹介します。

1. 新規のPythonファイルを開始し、 **count_ttl_avg.py**という名前を付けます。

2つの変数を作成して、両者をゼロに設定します。

```
count = 0
total = 0
```

変数**count**は入力された数値の数を追跡し、
変数**total**は入力された数値の現在の合計を保持します。
3番目の変数**num**は、一度に1つ入力された各数値を格納します。
0に設定しないで**input**ステートメントを使って、ユーザーから最初の数値を取得します。

```
num=float(input("Enter a number: "))
```

2. ここで、特定の数が入力されたときに終了する**while**ループを開始します。オプションは、0, -999, -1です。この一意の数は、処理しようとしている数のセットの一部にはできないことに注意してください。数の入力が終了したことを示すフラグ(flag)値として -999を採用します。

menu > Built-ins > Control. (メニュー>組み込み>コントロール)から**while**構造を選択します。

"等しくない"記号**!=**は**ctrl+=**にあります。**num**が-999に等しくない限りループを継続させたいので、これを使います。
行末にコロンの(:)を残すことを忘れないでください。
つぎに、**while**ブロックに入ります...

3. **while**ループブロックは、次のようになります。

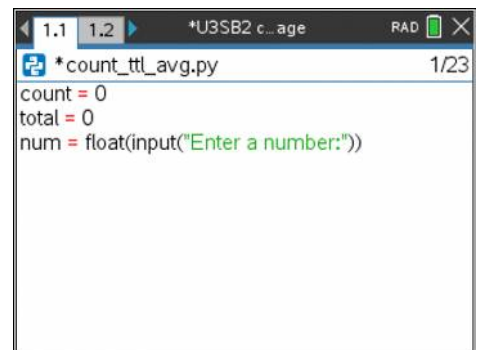
- a) 入力された数字の数を数えます。
- b) それらを合計します。
- c) つぎに、別の番号を尋ねます。

カウントはステートメントで行われます。

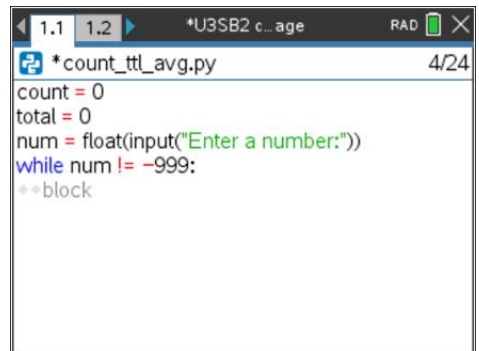
```
count = count + 1
```

これは、処理されるたびに変数**count**に1を加えます。

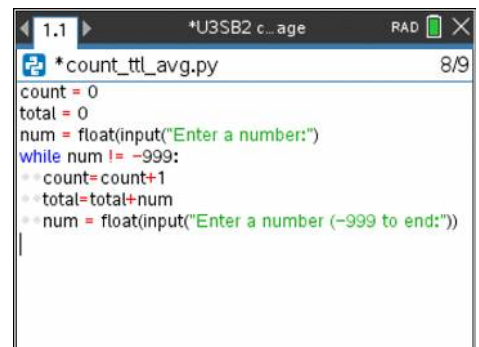
それらを合計することは、次のステートメントで行われます。



```
*count_ttl_avg.py 1/23
count = 0
total = 0
num = float(input("Enter a number:"))
```



```
*count_ttl_avg.py 4/24
count = 0
total = 0
num = float(input("Enter a number:"))
while num != -999:
  +=block
```



```
*count_ttl_avg.py 8/9
count = 0
total = 0
num = float(input("Enter a number:"))
while num != -999:
  +=count=count+1
  +=total=total+num
  +=num = float(input("Enter a number (-999 to end:)))
```



total = total + num

別の数の要求は、最初のようなinputステートメントです。

num=float(input("Enter a number (-999 to end): "))

ただし、ループ終了の方法を示すメッセージを追加します。

- ループが終了しました。1行か2行スキップして、インデントスペースを消去(バックスペース)します。

入力された数进行处理するときです。平均を計算できるように、**count**(カウント)と**total**(合計)を追跡しました。

次のステップを見る前に、自分で試してみてください。

```

1.1 1.2 *U3SB2 c...age RAD 6/18
count_ttl_avg.py
count = 0
total = 0
num = float(input("Enter a number:"))
while num != -999:
    count = count + 1
    total = total + num
    num = float(input("Enter a number (-999 to end):"))

```

- ステートメント

avg = total / count

は、平均を計算し、結果を変数avgに格納します。

これについて考えてみてください。前回のレッスンのように//ではなく、/をここで使うのはなぜですか。

最後に、カウント、合計、平均をユーザーに報告するprintステートメントを記述します。それぞれ別のprintステートメントを使うことをお勧めします。次のステップでコードを示します。

```

1.1 *U3SB2 c...age RAD 10/11
count_ttl_avg.py
count = 0
total = 0
num = float(input("Enter a number:"))
while num != -999:
    count = count + 1
    total = total + num
    num = float(input("Enter a number (-999 to en
avg = total / count

```

- 3つのprintステートメントを使って、結果を報告します。

print(" Count = ", count)
print(" Total = ",total)
print(" Average = ",avg)

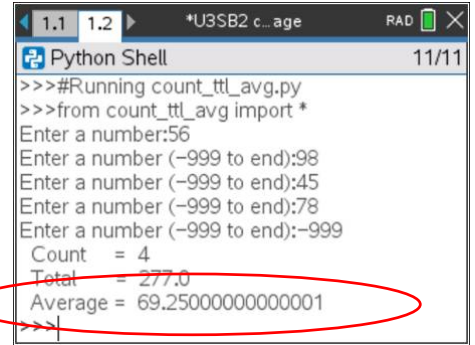
```

1.1 1.2 *U3SB2 RAD 11/11
Python Shell
>>>#Running count_ttl_avg.py
>>>from count_ttl_avg import *
Enter a number:65
Enter a number -999 to end:83
Enter a number -999 to end:7
Enter a number -999 to end:2
Enter a number -999 to end:-999
Count= 4
Total = 157.0
Average= 39.25
>>>|

```

特記事項 :

平均(Average)は、ゼロが後ろにたくさんあり、その後1が1つあることに気がきます。Pythonの方法は、2進浮動小数点数を格納・計算し、それらを10進近似値に変換します。一部の計算で小数が迷う可能性があります。この異常は、事実上コンピューティングプラットフォーム上のPythonすべてで普通であり、TI-Nspireとは何の関係もありません。



```
Python Shell 11/11
>>>#Running count_ttl_avg.py
>>>from count_ttl_avg import *
Enter a number:56
Enter a number (-999 to end):98
Enter a number (-999 to end):45
Enter a number (-999 to end):78
Enter a number (-999 to end):-999
Count = 4
Total = 277.0
Average = 69.25000000000001
>>>
```