

第06回の演習課題

演習課題 ex06-3.c

2つの正の実数 a , b の値を入力すると $\log_a b$ の値を小数点以下8桁まで出力するプログラム **ex06-3.c** を作成し、kiso2 コマンドを用いて提出しなさい。ただし、 a または b の値が 0 以下の場合は何も出力せずに再び値の入力を行い、 a の値として 1 が入力されるとプログラムは終了するものとする。

実行例：

```
t190900@so1cd0542-160:~/kiso2-2019/ex06$ ./ex06-3
a b? 0 0
a b? 2 3
1.58496250
a b? 3 2
0.63092975
a b? 0.2 4.6
-0.94819209
a b? 0.01 0.001
1.50000000
a b? 2 999
9.96434087
a b? 2.3 0.01
-5.52901878
a b? 1 0
```

演習課題 ex06-4.c

3つの整数 a , b , c の値を入力すると、それらの値を比較し

- すべて互いに異なる値
- いずれかの2つが同じ値で、他の1つが異なる値
- すべて同じ値

のいずれであるかを出力するプログラム **ex06-4.c** を作成し、kiso2 コマンドを用いて提出しなさい。ただし、3つの値の入力と、比較結果の出力は繰り返し行い、すべての値に 0 が入力されるとプログラムは終了するものとする。

実行例：

```
t190900@so1cd0542-160:~/kiso2-2019/ex06$ ./ex06-4
a b c? 1 2 3
すべて異なる
a b c? 3 2 1
すべて異なる
a b c? 1 3 1
2つが同値で1つ異なる
a b c? 1 1 3
2つが同値で1つ異なる
a b c? 3 1 1
```

2つが同値で1つ異なる

a b c? 1 1 1

すべて同値

a b c? 9 9 9

すべて同値

a b c? 0 0 0

演習課題 ex06-5.c

キーボードから自然数 N と実数 x の値を入力すると e^x のマクローリン展開 (0点周りのテイラー展開) の N 項まで $\sum_{k=0}^N \frac{x^k}{k!}$ の値を順に小数点以下10桁まで出力するプログラム **ex06-5.c** を作成し、kiso2コマンドを用いて提出しなさい。ただし、 N に負の値が入力されるとプログラムは何も出力せずに終了するものとする。

実行例：

```
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? 1 -1
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? 1 0
0: 1.00000000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? 1 1
0: 1.00000000000
1: 2.00000000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? 1 9
0: 1.00000000000
1: 2.00000000000
2: 2.50000000000
3: 2.66666666667
4: 2.70833333333
5: 2.71666666667
6: 2.71805555556
7: 2.7182539683
8: 2.7182787698
9: 2.7182815256
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? 1.2 10
0: 1.00000000000
1: 2.20000000000
2: 2.92000000000
3: 3.20800000000
4: 3.29440000000
5: 3.31513600000
6: 3.31928320000
7: 3.3199941486
8: 3.3201007909
9: 3.3201150098
10: 3.3201167161
```

```
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-5
x N? -1 10
0: 1.00000000000
1: 0.00000000000
2: 0.50000000000
3: 0.33333333333
4: 0.37500000000
5: 0.36666666667
6: 0.36805555556
7: 0.3678571429
8: 0.3678819444
9: 0.3678791887
10: 0.3678794643
```

$0! = 0\$$ としたいところですが、数学では一般的に $0! = 1\$$ と定義されます。プログラムでは $k! = 0\$$ で場合分けするのが簡単です。

演習課題 ex06-6.c

関西電力の月ごとの電気料金契約「従量電灯A」は、次のような料金単価体系（電気料金は定期的に変動するため単価が異なっているかも）になっている。

1. 1契約あたりの基本料金は 334.82円（税込、以下同様）であり、**15kWh** までの電力量ごとの料金はこの基本料金に含まれる(定額である)。
2. **15kWh** を超え、**120kWh** までの電力量ごとの料金は、1kWh あたり 19.95 円である。
3. **120kWh** を超え、**300kWh** までの電力量ごとの料金は、1kWh あたり 25.33 円である。
4. **300kWh** を超えた分の電力量ごとの料金は、1kWh あたり 28.76 円である。

キーボードから一ヶ月に利用した電力量を kWh を単位とする整数値で入力すると、「従量電灯A」で契約した場合の月額電気料金を小数点以下2桁まで（四捨五入して）出力するプログラム **ex06-6.c** を作成し、`kiso2` コマンドを用いて提出しなさい。ただし、電力量の入力と電気料金の出力は繰り返し行えるものとし、電気量に 0 以下の値が入力されるとプログラムが終了するものとする。

実行例：

```
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-6
kWh? 1
334.82 Yen
kWh? 15
334.82 Yen
kWh? 16
354.77 Yen
kWh? 120
2429.57 Yen
kWh? 121
2454.90 Yen
kWh? 300
6988.97 Yen
kWh? 301
7017.73 Yen
kWh? 999
27092.21 Yen
```

kWh? 0

プログラムのソースコード中に 19.95 などの定数を直接書き込むと、コードが読みにくくなり（数字の意味が不明）、修正もし難く、バグが入り込みやすくなります。#define によるマクロ定数定義を積極的に使いましょう。

演習課題 ex06-7.c

半径 r の n 次元ユークリッド球面の体積 $V_n(r)$ は、ユークリッド空間の次元 n が自然数であるとき、 k を自然数とした偶数次元の場合 $V_{2k}(r)$ と奇数次元の場合 $V_{2k+1}(r)$ に分けて $V_{2k}(r) = \frac{\pi^k}{k!} r^{2k}$, $V_{2k+1}(r) = \frac{2(k!)(4\pi)^k}{(2k+1)!} r^{2k+1}$ で表すことができる。

次元数 n を整数値として、半径 r を実数値としてそれぞれ入力すると、 n 次元ユークリッド球面の体積 $V_n(r)$ を計算して出力するプログラム **ex06-7.c** を作成し、kiso2コマンドを用いて提出しなさい。ただし、 n, r に 負の値が入力されるとプログラムは何も出力せずに終了するものとする。

実行例：

```
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 0 0
半径 0.000000 の 0 次元ユークリッド球面の体積は 1.000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 0 1
半径 1.000000 の 0 次元ユークリッド球面の体積は 1.000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 1 1
半径 1.000000 の 1 次元ユークリッド球面の体積は 2.000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 1 2
半径 2.000000 の 1 次元ユークリッド球面の体積は 4.000000
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 2 1
半径 1.000000 の 2 次元ユークリッド球面の体積は 3.141593
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 2 2
半径 2.000000 の 2 次元ユークリッド球面の体積は 12.566371
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 3 1
半径 1.000000 の 3 次元ユークリッド球面の体積は 4.188790
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 3 2
半径 2.000000 の 3 次元ユークリッド球面の体積は 33.510322
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 4 1
半径 1.000000 の 4 次元ユークリッド球面の体積は 4.934802
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 4 2
半径 2.000000 の 4 次元ユークリッド球面の体積は 78.956835
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 9 1
半径 1.000000 の 9 次元ユークリッド球面の体積は 3.298509
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
```

```
n r? 9 2
半径 2.000000 の 9 次元ユークリッド球面の体積は 1688.836558
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-7
n r? 32 2.2
半径 2.200000 の 32 次元ユークリッド球面の体積は 390215.175952
```

べき乗の計算は `math.h` の関数 `pow()` を利用できます。

演習課題 ex06-8.c

キーボードから整数値をデータとして入力する度に、それまでに入力されたデータ数、総和、平均値、最大値、最小値を出力するプログラム **ex06-8.c** を作成し、`kiso2` コマンドを用いて提出しなさい。ただし、全ての項目について、それぞれの値を10文字幅で表示し、また、平均値は小数点以下3桁までを四捨五入して表示する。

実行例：

```
t190900@s01cd0542-160:~/kiso2-2019/ex06$ ./ex06-8
? 9
データ数: 1, 総和: 9, 平均値: 9.000, 最大値: 9, 最小値: 9
? 10
データ数: 2, 総和: 19, 平均値: 9.500, 最大値: 10, 最小値: 9
? 999
データ数: 3, 総和: 1018, 平均値: 339.333, 最大値: 999, 最小値: 9
? -10000
データ数: 4, 総和: -8982, 平均値: -2245.500, 最大値: 999, 最小値: -10000
? 1
データ数: 5, 総和: -8981, 平均値: -1796.200, 最大値: 999, 最小値: -10000
? 2
データ数: 6, 総和: -8979, 平均値: -1496.500, 最大値: 999, 最小値: -10000
? 3
データ数: 7, 総和: -8976, 平均値: -1282.286, 最大値: 999, 最小値: -10000
? 4
データ数: 8, 総和: -8972, 平均値: -1121.500, 最大値: 999, 最小値: -10000
? 5
データ数: 9, 総和: -8967, 平均値: -996.333, 最大値: 999, 最小値: -10000
? -10001
データ数: 10, 総和: -18968, 平均値: -1896.800, 最大値: 999, 最小値: -10001
? 9999
データ数: 11, 総和: -8969, 平均値: -815.364, 最大値: 9999, 最小値: -10001
? 111
データ数: 12, 総和: -8858, 平均値: -738.167, 最大値: 9999, 最小値: -10001
?
```

端末上で実行中のプログラムを強制終了する場合は、`Ctrl-C` (コントロールキ) + `C` を押します。

From:
<https://www-slab.math.ryukoku.ac.jp/> - **www-slab.math**

Permanent link:
<https://www-slab.math.ryukoku.ac.jp/lecture/kiso2/2018/ex06>

Last update: **2019/09/23 13:24**

