2025年度前期
情報処理演習
第7回

2025.6.4

名古屋大学減災連携研究センター
Disaster Mitigation Research Center, NAGOYA UNIVERSITY

平山　修久

数の表現と誤差

- 0.1を10回足しても1.0にならない！
- コンピューターでは，2進数で表現する。

0.1 = (0.CCCCD)16×2-3

```
IMPLICIT NONE
REAL :: N
INTEGER :: I
N=0.0
DO I = 1, 10
        N = N + 0.1
        PRINT *, N
END DO
```

```
MacPro2017:0531 nhirayama$ ./a.out
  0.100000001
  0.200000003
  0.300000012
  0.400000006
  0.500000000
  0.600000024
  0.700000048
  0.800000072
  0.900000095
   1.00000012
MacPro2017:0531 nhirayama$
```

倍精度実数

DOUBLE PRECISION　もしくは　REAL(8)

単精度実数

```
IMPLICIT NONE
REAL :: N
INTEGER :: I
N=0.0
DO I = 1, 10
        N = N + 0.1
        PRINT *, N
END DO
```

倍精度実数

```
IMPLICIT NONE
DOUBLE PRECISION :: N
INTEGER :: I
N=0.0d0
DO I = 1, 10
        N = N + 0.1d0
        PRINT *, N
END DO
```

```
MacPro2017:0531 nhirayama$ gfortran demo5_1.f90
MacPro2017:0531 nhirayama$ ./a.out
   0.100000001
   0.200000003
   0.300000012
   0.400000006
   0.500000000
   0.600000024
   0.700000048
   0.800000072
   0.900000095
    1.00000012
MacPro2017:0531 nhirayama$ gfortran demo5_1.f90
MacPro2017:0531 nhirayama$ ./a.out
   0.10000000000000001
   0.20000000000000001
   0.30000000000000004
   0.40000000000000002
   0.50000000000000000
   0.59999999999999998
   0.69999999999999996
   0.79999999999999993
   0.89999999999999991
   0.99999999999999989
MacPro2017:0531 nhirayama$
```

単精度実数
精度7桁
有効数字6桁

倍精度実数
精度16桁
有効数字15桁

Tips

前回の課題

― 単精度と倍精度との有効数字の違い

$$2.718282x^2 - 684.4566x + 0.3161592 = 0$$

$$x = 4.61914 \times 10^{-4}, 251.797$$

```
MBPro13-2016:0603 nhirayama$ ./a.out
 Enter the coefficients of A, B, C:
2.718282 -684.4566 0.3161592
 The roots are     251.79703371052130
                   4.6191355300849733E-004
```

```
MBPro13-2016:0603 nhirayama$ ./a.out
 Enter the coefficients of A, B, C:
2.718282 -684.4566 0.3161592
 The roots are     251.797028
                   4.60298354E-04
```

$$2.718282x^2 - 1.854089x + 0.3161592 = 0$$

$$x = 0.340569, 0.341512$$

```
MBPro13-2016:0603 nhirayama$ ./a.out
 Enter the coefficients of A, B, C:
2.718282 -1.854089 0.3161592
 The roots are    0.34151206351711483
                  0.34056911864132211
```

```
MBPro13-2016:0603 nhirayama$ ./a.out
 Enter the coefficients of A, B, C:
2.718282 -1.854089 0.3161592
 The roots are    0.341524273
                  0.340556920
```

本日の目標

2025/06/04

- フラグを使いこなす
  - 素数を計算する

MAKE NEW STANDARDS. 東海国立大学機構 | NAGOYA UNIVERSITY

フラグについて

条件（状態）判定

ー onとoffの状態判定に用いる変数

✓ 例えば，「支払い済みフラグ」
✓ 最初は0を入れる。
✓ お金を支払ったら，支払い済みフラグの値を「1」に。
✓ これで，「支払い済みフラグ」の値を見ることで，お金を払ったか否かが判断できる。

```
支払い済みフラグ = 0
IF (お金を払った? == 払った) THEN
  支払い済みフラグ = 1
END IF
IF (支払い済みフラグ == 0) 催促。
```

DOループの入れ子

カウント変数は別に

```
DO M = 1, Last_M
   DO N = 1, Last_N
      Product = M * N
      PRINT *, M, " ", N, " ", Product
   END DO
END DO
```

```
MacPro2017:0531 nhirayama$ gfortran demo5_1.f90
MacPro2017:0531 nhirayama$ ./a.out
           1           1           1
           1           2           2
           1           3           3
           2           1           2
           2           2           4
           2           3           6
           3           1           3
           3           2           6
           3           3           9
           4           1           4
           4           2           8
           4           3          12
MacPro2017:0531 nhirayama$
```

!DO変数の変更は不可

```
DO I = 1, 200
   I = 100
END DO
```


今日の課題

2025.6.4

1. 組み込み関数MODを用いて，正の整数 N までの素数の個数を求めるプログラムを作成する。$N = 1{,}000{,}000$までの素数の個数，ならびに最大の素数を求めよ。

   MOD(m, n)：mをnで割った商の剰余を与える

   提出物：**プログラム　kadai7_1.f90**

   実行結果のスクリーン画像　kadai7_1.png

素数を求める

アルゴリズム

1. 調べる最大の正の整数Pを入力する
2. 素数の数を初期化（2を代入）する。
3. DOループ，n=5, P, 2　←　**nが偶数のときは調べる必要はない。**
   nが素数かどうかについて調べる。→　3からMNまでの整数で割り切れるかどうか順に調べる。MNは$\sqrt{n}$でよい
   1. 素数のフラグ（FP）を0とする。
   2. MNを計算する。MN=INT(SQRT(real(n))
   3. DOループ，m=3，MN
      1. MOD(n，m)==0であれば素数でない。→素数フラグを1にする。
   4. MNまで調べた後，FPが0であれば素数，1であれば素数でない
   5. 素数であれば（FP==0），素数の数を1増やす。