

11042 計算機言語 9 回目

サポートページ:<https://goo.gl/678wGM>

石井 史之

金沢大学 数物科学系

December 11, 2017

計算機言語

- ガイダンス・・・ 10/2
- 復習 1(print 文, 繰返し等)・・・ 10/16
- 復習 2(2章「演算と型」, 繰返し等)・・・ 10/23
- 条件分岐 (3章「プログラムの流れの分岐」)・・・ 10/31(火) は休講 ,11/6
- 繰返し (4章「プログラムの流れの繰返し」)・・・ 11/13
- 配列, 関数, サブルーチン 1 (5章「配列」6章「関数」)・・・ 11/20
- 配列, 関数, サブルーチン 2 (5章「配列」6章「関数」)・・・ 11/27
- 基本型 (7章「基本型」)・・・ 12/4
- ポインタ 1 (10章「ポインタ」)・・・ 12/11
- ポインタ 2 (10章「ポインタ」)・・・ 12/18
- ポインタ 3 (11章「文字列とポインタ」)・・・ 12/25
- 文字列 (9章「文字列の基本」)・・・ 1/15
- 構造体 (C, Fortran) (12章「構造体」)・・・ 1/22
- ファイル入出力 (13章「ファイル入出力」, 自由課題)・・・ 1/29
- まとめ・アンケート, 自由課題・・・ 2/5

動的割付け

型, ALLOCATABLE :: 配列変数名 (:) のリスト

.....

ALLOCATE (配列変数名 (添え字範囲の指定), ...)

動的割付けの解除

DEALLOCATE (配列変数名,)

kuku.f90(静的割付け)

```
PROGRAM array_kuku
  IMPLICIT NONE
  INTEGER :: I, J
  INTEGER :: DIMENSION(9,9) :: KUKU
  DO I=1,9
    DO J=1,9
      KUKU(I,J)=I*J
    END DO
  END DO
  WRITE(*,'(1X,9I5)')KUKU
  STOP
END PROGRAM array_kuku
```

nn.f90(動的割付け)

```
PROGRAM array_nn
  IMPLICIT NONE
  INTEGER :: I, J, NMAX
  INTEGER, ALLOCATABLE :: NN(:, :)
  PRINT *, 'Input maximum integer for multiplication table:'
  READ *, NMAX
  ALLOCATE(NN(NMAX, NMAX))
  DO I=1, NMAX
    DO J=1, NMAX
      NN(I, J)=I*J
    END DO
  END DO

  DO I=1, NMAX
    WRITE(*, *) 'I=', I
    DO J=1, NMAX
      WRITE(*, *) NN(I, J)
    END DO
  END DO
  DEALLOCATE(NN)

  STOP
END PROGRAM array_nn
```

ポインタとは

変数や配列などが記録されている記憶領域の番地そのものを表す変数をポインタという。動的に大きさが変化する部分配列やリスト構造を扱う場合に役立つ。マシン語でプログラムで書くときはメモリ (記憶装置) の番地を指定し、データの取り出し、格納をおこなう。

- 「&変数名」を用いて「変数の格納場所 (番地)」を表す。
(例) x が変数のとき、&x は「x に割り当てられた番地」
- 「*番地」で「ある番地に格納されている変数の値」を表す。
(例) a が番地のとき、*a は「a 番地の内容」
- 「番地」を「ポインタ」と呼ぶ。
- &x は「変数 x に対するポインタ」
- *a は「ポインタ a が指す変数の値」

ポインタ	変数の値
&x	x
a	*a

pointer-plus.c

```
#include <stdio.h>
int main(void){
    int *a, *b, *c;
    int ia, ib, ic;
    a=&ia; b=&ib; c=&ic;
    *a=2;
    *b=3;
    *c=*a+*b;
    printf("a=%d b=%d c=%d\n", *a,*b,*c);
    printf("a=%p b=%p c=%p\n", a,b,c);
    return 0;
}
```

いずれか 1 問を提出

- 明解 C 言語 10 章の演習問題。
- C 言語のポインタを用いて nn.f90 を C 言語で書く。(ヒント : malloc, free 関数を使う。明解 C(中級))