

Fortranプログラミング入門

-反復処理(2)-

2.7.18 2.8

反復処理(do文)

☆文法 do

繰り返し実行したい文

end do

- do文の始値と終値を記述しない.
- 繰り返し実行したい文が無限に実行される.
- do文を抜け出す仕組みが必要!

☆文法 exit

exit文が実行されると処理は

exit文がある最内のdo文と対になるend doの次の文
が実行される.

```
do
...
if (条件) exit
...
end do
```



```
do
do
...
if (条件) exit
...
end do
...
end do
```

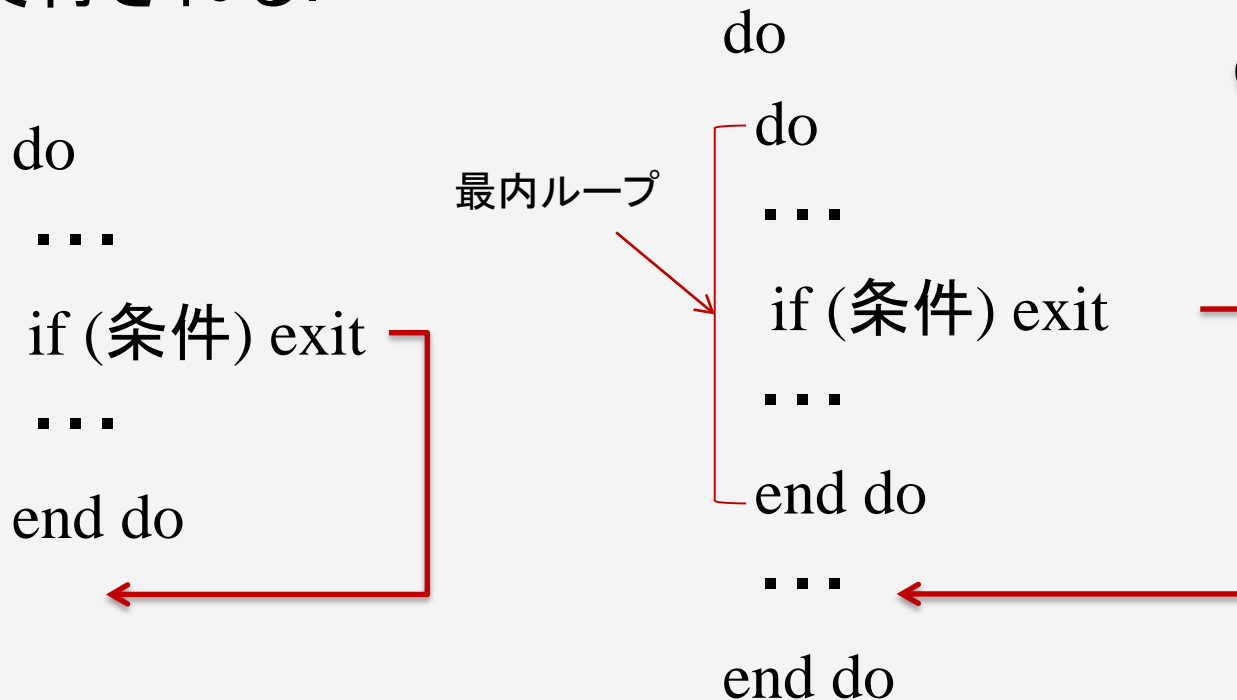
if文と組み合わせ
て使おう!!



反復処理(exit文)

☆文法 exit

exit文が実行されると処理は
exit文がある最内のループのend doの次の文
が実行される.



if文と組み合わせて使おう!!



☆文法 exit

exit文が実行されると処理は

exit文がある最内のループのend doの次の文

が実行される.

do

...

if (条件) exit

do

...

end do

...

end do

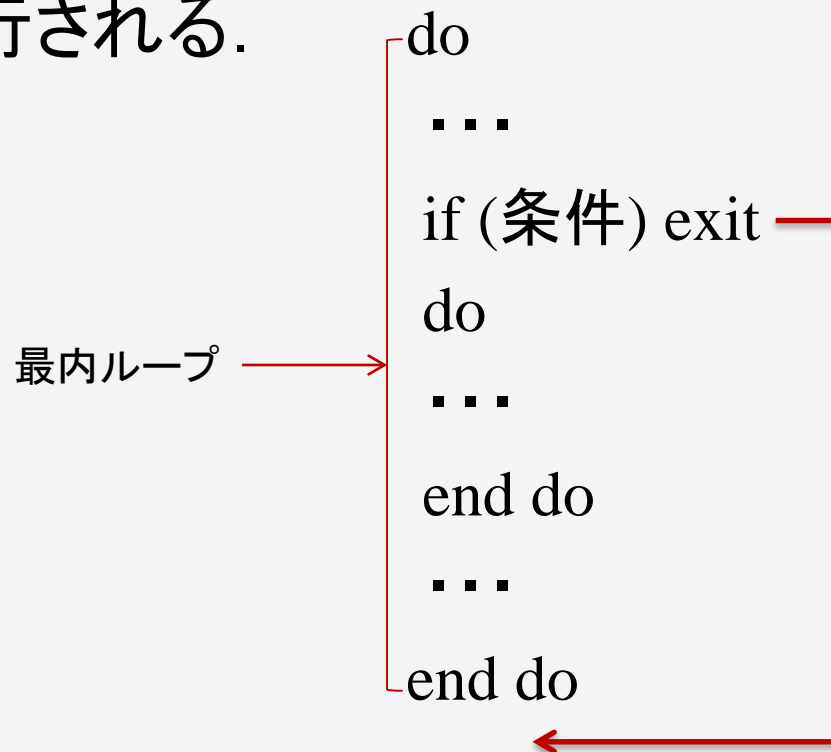
if文と組み合わせて使おう!!



反復処理(exit文)

☆文法 exit

exit文が実行されると処理は
exit文がある最内のループのend doの次の文
が実行される.



if文と組み合わせて使おう!!




反復処理(cycle文)

☆文法 cycle


cycle文が実行されると処理は

cycle文がある最内のdo文の次の文が実行される。

```
do
  ...
  if (条件) cycle
  ...
end do
```



```
do
  do
    ...
    if (条件) cycle
    ...
  end do
  ...
end do
```

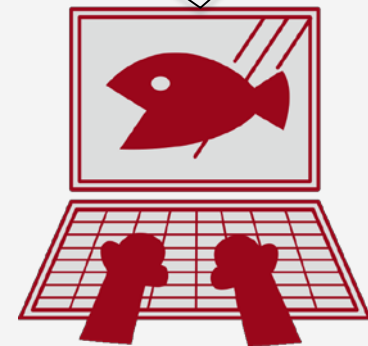


if文と組み合わせ
て使おう!!



整数型の変数 n を宣言し, read文で値を読み込む. n の値が0以下ならば $n \leq 0$ と表示し, 再度read文で n の値を読み込む. n が0よりも大きいならば $n > 0$ と表示し, 終了するプログラムを作成せよ.

無限ループを利用しよう.
終了はif文とexit文で書こう




```
program exp1
  implicit none
  integer :: n
  do
    write(*,*) 'Please input an integer number n'
    read(*,*) n
    if (n<=0) then
      write(*,*) 'n<=0'
      cycle
    else if (n>0) then
      write(*,*) 'n>0'
      exit
    end if
  end do
  stop
end program exp1
```

今回は省略可



→ cycle