

情報処理概論

第15回 演習3

情報基盤研究開発センター 谷本 輝夫

横の並びを全部の行でチェック

```
do j = 1, n ! (前回、2に固定した部分を j に)
  count = 0
  do i =1, m
    if (board(i, j) == ball(side)) then
      count = count + 1
      if (count == 4) then
        write (*, *) "won!"
        stop
      end if
    else
      count = 0
    end if
  end do
end do
```

6	[]	[]	[o]	[*]	[]	[]	[]
5	[]	[*]	[o]	[*]	[]	[]	[]
4	[]	[o]	[*]	[o]	[]	[]	[]
3	[]	[*]	[o]	[*]	[]	[]	[]
2	[]	[o]	[o]	[o]	[o]	[]	[]
1	[*]	[o]	[*]	[o]	[*]	[]	[]
	1	2	3	4	5	6	7

→ i

縦は？

```
do j = 1, m
  count = 0
  do i = 1, n
    if (board(j, i) == ball(side)) then
      count = count + 1
      if (count == 4) then
        write (*, *) "won!"
        stop
      end if
    else
      count = 0
    end if
  end do
end do
```

6	[]	[]	[o]	[*]	[]	[]	[]
5	[]	[*]	[o]	[*]	[]	[]	[]
4	[]	[o]	[*]	[o]	[]	[]	[]
3	[]	[*]	[o]	[*]	[]	[]	[]
2	[]	[o]	[o]	[o]	[o]	[]	[]
1	[*]	[o]	[*]	[o]	[*]	[]	[]
	1	2	3	4	5	6	7

→ i

斜めは？(右上がり)

```
do i = 1-n+1, m
  count = 0
  do j = 1, n
    if ((1 <= i+j-1) .and. (i+j-1 <= m)) then
      if (board(i+j-1, j) == ball(side)) then
        count = count+1
        if (count == 4) then
          write(*, *) 'won!'
          stop
        end if
      else
        count = 0
      end if
    end if
  end do
end do
```



6						[]	[]	[o]	[*]	[]	[]	[]
5						[]	[]	[*]	[o]	[*]	[]	[]
4						[]	[]	[]	[o]	[*]	[o]	[]
3						[]	[]	[]	[]	[*]	[o]	[*]
2						[]	[]	[]	[]	[]	[o]	[o]
1						[]	[]	[]	[]	[]	[*]	[o]
	-4	-3	-2	-1	0	1	2	3	4	5	6	7

→ i

斜めは？(右下がり)

```
do i = 1-n+1, m
  count = 0
  do j = 1, n
    if ((1 <= i+j-1) .and. (i+j-1 <= m)) then
      if (board(i+j-1, n-j+1) == ball(side)) then
        count = count+1
        if (count == 4) then
          write(*, *) 'won!'
          stop
        end if
      else
        count = 0
      end if
    end if
  end do
end do
```



```
6  [ ][ ][ ][ ][ ][ ][ ][o][*][ ][ ][ ]
5   [ ][ ][ ][ ][ ][ ][*][o][*][ ][ ][ ]
4    [ ][ ][ ][ ][ ][o][*][o][ ][ ][ ]
3     [ ][ ][ ][ ][*][o][*][ ][ ][ ]
2      [ ][ ][o][o][o][o][ ][ ][ ]
1       [*][o][*][o][*][ ][ ][ ]
-4 -3 -2 -1 0 1 2 3 4 5 6 7
```

→ i

発展バージョン1：メインプログラム

```
program four
  implicit none
  integer, parameter :: m=7, n=6
  character(len=1), dimension(m, n) :: board
  integer :: step, side

  board = ''
  call show(m, n, board)

  do step = 1, m*n/2
    do side = 1, 2
      call drop(m, n, board, side)      ! dropサブルーチン中でshowもcheckも呼ぶ
      ! call show(m, n, board)          ! コメントアウト
      ! call check(m, n, board, side)   ! コメントアウト
    end do
  end do
  stop
end program
```

発展バージョン1：サブルーチン drop

```
subroutine drop(m, n, board, side)
implicit none
integer, intent(IN) :: m, n, side
character(len=1), dimension(m, n), intent(INOUT) :: board
integer :: x, y, done
character(len=1) :: mark
```

！ 中略

```
    call show(m, n, board, side)
    call check(m, n, board, side, x, y)
end subroutine
```

発展バージョン2：メインプログラム

```
program four
  implicit none
  integer, parameter :: m=7, n=6
  character(len=1), dimension(m, n) :: board
  integer :: step, side, x, y

  board = ' '
  call show(m, n, board)

  do step = 1, m*n/2
    do side = 1, 2
      call drop(m, n, board, side, x, y)
      call show(m, n, board)
      call check(m, n, board, side, x, y)
    end do
  end do
  stop
end program
```


発展バージョン2：サブルーチン drop

```
subroutine drop(m, n, board, side, x, y)
implicit none
integer, intent(IN) :: m, n, side
integer, intent(OUT) :: x, y
character(len=1), dimension(m, n), intent(INOUT) :: board
integer :: done
character(len=1) :: mark
```

！ 中略

```
end subroutine
```

勝敗判定サブルーチン

```
subroutine check(m, n, board, side, x, y)
  implicit none
  character(1), dimension(m, n), intent(INOUT) :: board
  integer, intent(IN) :: m, n, side, x, y
  integer :: i, j, count
  character(1), dimension(2) :: ball = ('*', 'o')

  ! 横
  ! 縦
  ! 右上がり
  ! 右下がり

endsubroutine check
```

勝敗判定 横

```
!横    (縦の位置はyに固定して良い)
count = 0
do i = 1, m
  if (board(i, y) == ball(side)) then
    count = count+1
    if (count == 4) then
      write(*, *) ball(side), ' is winner!'
      stop
    end if
  else
    count = 0
  end if
end do
```

レポート提出に関して

- ▶ 「プログラム」と「説明」はセットです。両方出ていないと採点されない場合があります。
- ▶ レポート課題01～10
 - ▶ 締め切り：7月31日13:00
- ▶ レポート課題11-1、11-2、13、14
 - ▶ 締め切り：8月15日13:00
 - ▶ 11-1、11-2は「(新)」に提出、「(旧)」に提出した人も「(新)」に再度提出してください

試験に関して

- ▶ 日時
 - ▶ 7月31日13:00～
- ▶ 場所
 - ▶ この教室
- ▶ 内容
 - ▶ Linux の基本的なコマンド
 - ▶ Fortran プログラミングの基本
 - ▶ 演習で扱ったプログラムに関する問題

授業アンケート