



Metal part2

前回の補足

- for文

- for 変数名=初期値 to 終わりの値 step 増分

- stepはあってもなくても良い。省略した場合は増分は+1となる。

- 例：変数countを0から10まで2つつ増やす

```
for count=0 to 10 step 2
```

- for文とnext文は対応しなければならない。

- if文

- 条件が成り立っているかどうかを調べ、その結果により処理の内容を変える。

- 例：マウスカーソルのx位置が偶数のときは“even”、奇数のときは“odd”と表示する

```
if getmousex mod 2=0 then
```

```
  print “even”
```

```
else
```

```
  print “odd”
```

```
endif
```

無限ループ

- C,C++,Java等では while文やfor文でも実現可能

例:

```
while(1) {...}
```

```
for(;;) {...}
```

- METALの場合はgoto文を用いる(方が素直)。
 - goto文の書式はgoto ラベルとなり、ラベルのところにジャンプする。
 - 例: 1秒待つ

```
t=timer
```

```
again:
```

```
if timer-t<=1 then goto again
```

注: 構造化プログラミングという見地からは、流れが見えにくくなる為、goto文の使用は基本的にNG(絶対にNGという訳ではない)。

構造化プログラミングとgoto文

意味としては同値なプログラム

例：goto文を用いた場合

```
i=0
forLoop:
  print "i="; i
  i=i+1
  if i<=10 then goto forLoop
print "done."
```

例：for文を用いた場合

```
for i=0 to 10
  print "i="; i
next I
print "done."
```

注：for文の方が情報を多く含んでいる為、見てすぐに理解できる。

goto文の使用は慎重に。

グラフィックスーその1

- 表示Windowの位置とサイズの指定
`resize console left, top, right, bottom`
- 関連する関数
`screen width`
`screen height`
画面の解像度を返す
- 例: Windowを画面の中央に400×300ピクセルで表示する
`w=400`
`h=300`
`x=(screen width-w)/2`
`y=(screen height-h)/2`
`resize console x, y, x+w, y+h`

グラフィックスーその2

- 画面の消去
 - cls文(Clear Screen)

```
print "Click to clear."  
wait button  
cls
```

グラフィックスーその3

- 直線の描画

- line文: line x0,y0,x1,y1

- サンプルプログラム(インタラクティブに)線を引く

```
rem --- sample program single buffer---
```

```
w=400
```

```
h=300
```

```
x=(screen width-w)/2
```

```
y=(screen height-h)/2
```

```
resize console x, y, x+w, y+h
```

```
cls
```

```
wait button
```

```
startX=getmousex
```

```
startY=getmousey
```

```
again:
```

```
  cls
```

```
  endX=getmousex
```

```
  endY=getmousey
```

```
  line startX, startY, endX, endY
```

```
if button then goto again
```