

105-2: EE4052
計算機程式設計
Computer Programming

Unit 11: 動作與動畫

連 豐 力

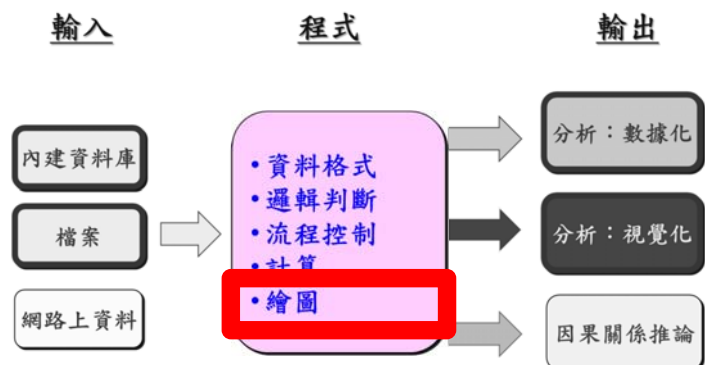
臺大電機系

Feb 2017 - Jun 2017

課程主題進度

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

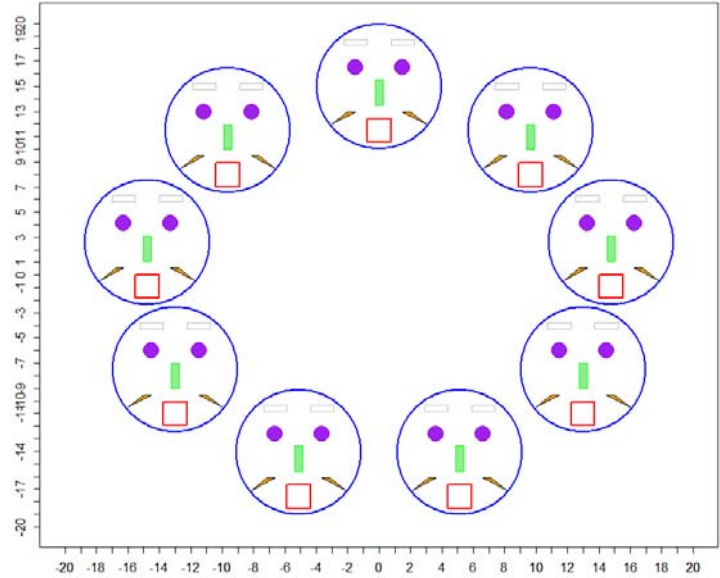
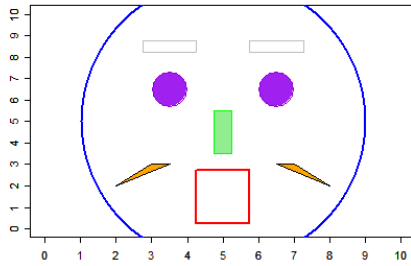
- **U01:** 課程介紹：討論主題，作業，報告，進行方式
- **U02:** 設定軟體 R 與 Rstudio
- **U03:** 數據處理與繪圖指令功能
- **U04:** 資料類別與基本運算
- **U05:** 邏輯判斷與流程控制
- **U06:** 函數：計算與排序
- **U07:** 多維度資料格式
- **U08:** 檔案資料輸入與輸出
- **U09:** 繪圖功能與文字
- **U10:** 多重繪圖與顏色
- **U11:** 函數：動畫與動作
- **U12:** 探索性資料分析
- **U13:** 資料前置處理
- **U14:** 資料連結分析



Unit 11: 繪圖 + 流程 -> 動畫

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

```
■ for( index in index.set ) {  
    statement  
}
```



- 3

大綱

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

- 畫一個臉
- 行走路徑與迴圈控制
- 畫一個動畫
- 數學函數繪圖

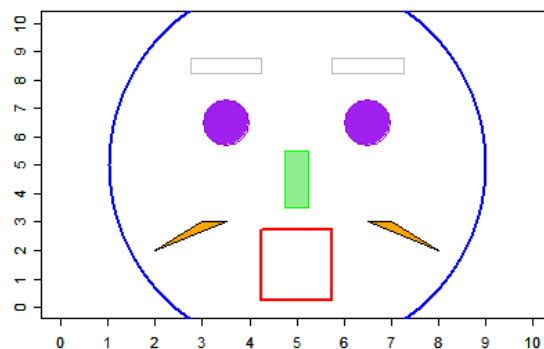
畫一個臉

5

圖形元件 - 圓形 矩形 多邊形

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

- `symbols()` # 圓形 矩形
- `polygon()` # 多邊形



- 6

- `windows(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 4, 4, 2) + 0.1)`

- `plot(0:10, 0:10, xlab = "", ylab = "", pch = 1:11, col = 1:8, type = "n")`
- `axis(side = 1, at = 0:10)`
- `axis(side = 2, at = 0:10)`

- # 臉輪廓
- `symbols(5, 5, circles = 4, fg = "blue", lwd = 2, inches = FALSE, add = TRUE)`

- 7

- # 眼睛
- `symbols(3.5, 6.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE)`
- `symbols(6.5, 6.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE)`

- # 眉毛
- `symbols(3.5, 8.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE)`
- `symbols(6.5, 8.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE)`

- 8

- # 鼻子
- `symbols(5, 4.5, rectangles = matrix(c(0.5, 2), nrow = 1), fg = "green", bg = "lightgreen", inches = FALSE, add = TRUE)`
- # 嘴巴
- `symbols(5, 1.5, squares = 1.5, fg = "red", lwd = 2, inches = FALSE, add = TRUE)`
- # 鬍子
- `polygon(c(3.5, 3, 2), c(3, 3, 2), col = "orange")`
- `polygon(c(6.5, 7, 8), c(3, 3, 2), col = "orange")`
- `par(old.par)`

- 9

繪圖函數 - myFace()

```
# draw face
```

```
myFace <- function( cx, cy ) {
```

```
# 臉輪廓
```

```
symbols( cx+0, cy+0, circles = 4, fg = "blue", lwd = 2, inches = FALSE, add = TRUE )
```

```
# 眼睛
```

```
symbols( cx-1.5, cy+1.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE )
```

```
symbols( cx+1.5, cy+1.5, circles = 0.5, fg = "dark violet", bg = "purple", inches = FALSE, add = TRUE )
```

```
# 眉毛
```

```
symbols( cx-1.5, cy+3.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE )
```

```
symbols( cx+1.5, cy+3.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg = "gray", inches = FALSE, add = TRUE )
```

- 10

繪圖函數 - myFace()

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

鼻子

```
symbols( cx+0, cy-0.5, rectangles = matrix(c(0.5, 2), nrow = 1), fg =  
"green", bg = "lightgreen", inches = FALSE, add = TRUE )
```

嘴巴

```
symbols( cx+0, cy-3.5, squares = 1.5, fg = "red", lwd = 2, inches = FALSE,  
add = TRUE )
```

鬍子

```
polygon( c( cx-1.5, cx-2, cx-3), c( cy-2, cy-2, cy-3), col = "orange" )  
polygon( c( cx+1.5, cx+2, cx+3), c( cy-2, cy-2, cy-3), col = "orange" )
```

```
}
```

- 11

繪圖函數 - myFaceOff()

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

draw face

```
myFaceOff <- function( cx, cy ) {
```

臉輪廓

```
symbols( cx+0, cy+0, circles = 4, fg = "white", lwd = 2, inches = FALSE,  
add = TRUE )
```

眼睛

```
symbols( cx-1.5, cy+1.5, circles = 0.5, fg = "white", bg = "white", inches =  
FALSE, add = TRUE )
```

```
symbols( cx+1.5, cy+1.5, circles = 0.5, fg = "white", bg = "white", inches  
= FALSE, add = TRUE )
```

眉毛

```
symbols( cx-1.5, cy+3.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg =  
"white", inches = FALSE, add = TRUE )
```

```
symbols( cx+1.5, cy+3.5, rectangles = matrix(c(1.5, 0.5), nrow = 1), fg =  
"white", inches = FALSE, add = TRUE )
```

- 12

鼻子

```
symbols( cx+0, cy-0.5, rectangles = matrix(c(0.5, 2), nrow =1), fg =  
"white", bg = "white", inches = FALSE, add = TRUE )
```

嘴巴

```
symbols( cx+0, cy-3.5, squares = 1.5, fg = "white", lwd = 2, inches =  
FALSE, add = TRUE )
```

鬍子

```
polygon( c( cx-1.5, cx-2, cx-3), c( cy-2, cy-2, cy-3), col = "white", border =  
"white" )
```

```
polygon( c( cx+1.5, cx+2, cx+3), c( cy-2, cy-2, cy-3), col = "white", border  
= "white" )
```

```
}
```

- 13

行走路徑與迴圈控制

行走路徑

- `x <- c(-15:15, rep(15, 31), 15:-15, rep(-15, 31))`
- `y <- c(rep(15, 31), 15:-15, rep(-15, 31), -15:15)`

- `x <- sin(1:90/45*pi)*15`

- `y <- cos(1:90/45*pi)*15`

-

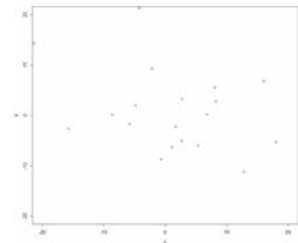
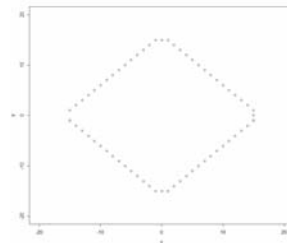
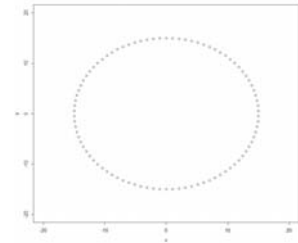
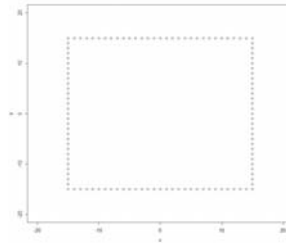
- `x <- c(-15:15, 15, 15:-15)`

- `y <- c(1:15, 15, 15:1, 0, -1:-15, -15, -15:-1)`

-

- `x <- rnorm(20) * 10`

- `y <- rnorm(20) * 10`



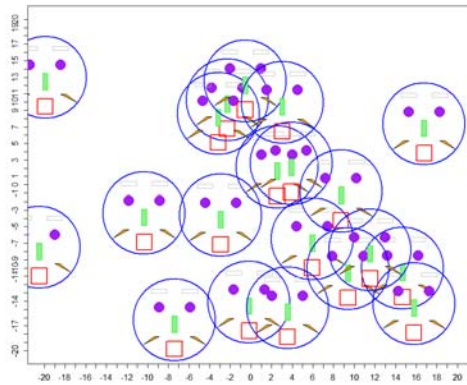
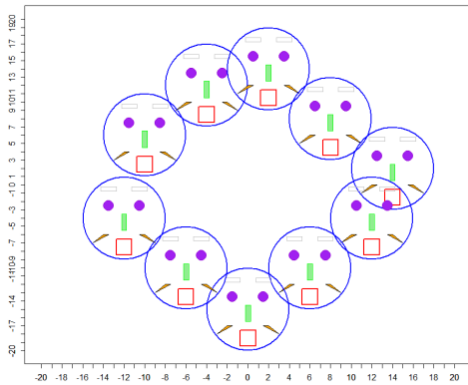
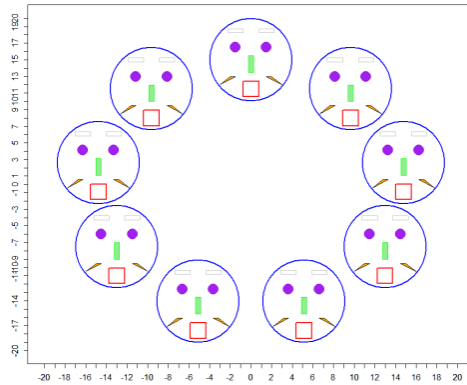
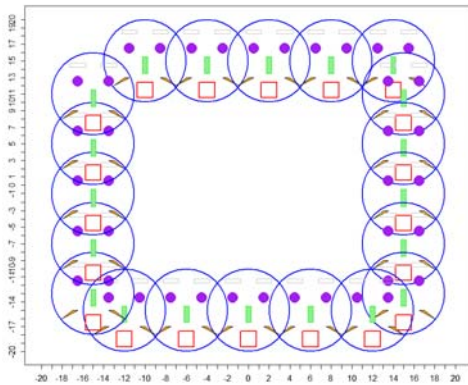
- 15

迴圈控制

- `Num <- length(x)`
- `for (i in 1:Num){`
- `myFace(x[i], y[i])`
- `#line <- readline()`
- `Sys.sleep(0.1)`
- `myFaceOff(x[i], y[i])`
- `#line <- readline()`
- `}`
- `for (i in 1:Num){`
- `myFace(x[i], y[i])`
- `}`

- 16

行走路徑

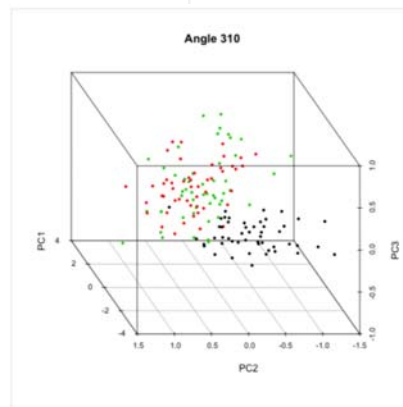
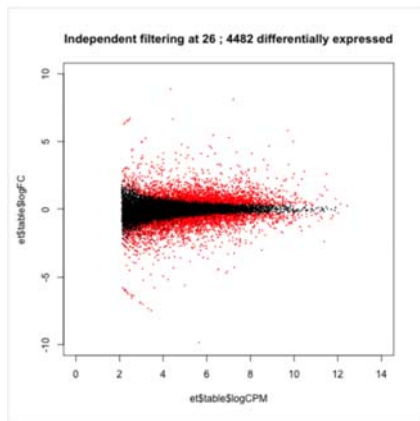
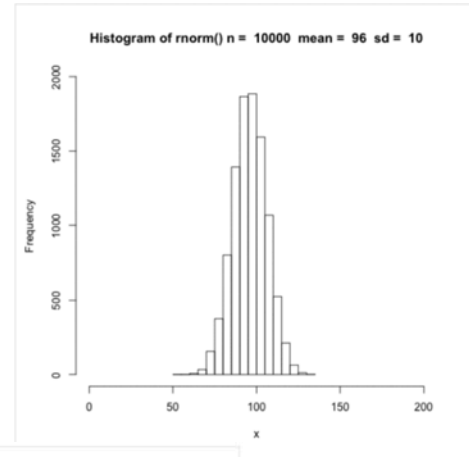
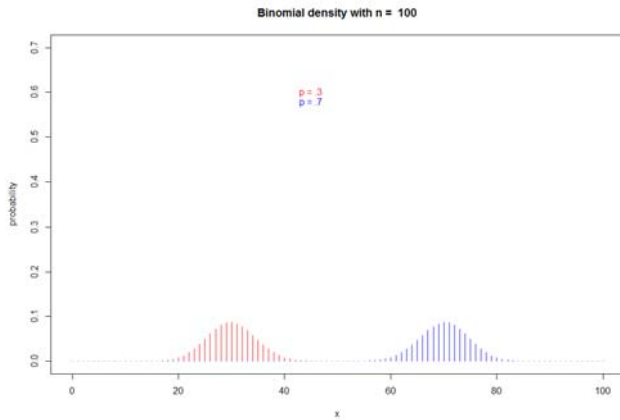


- 17

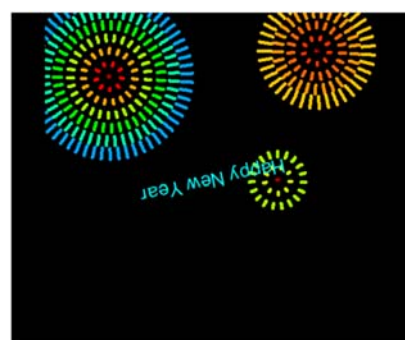
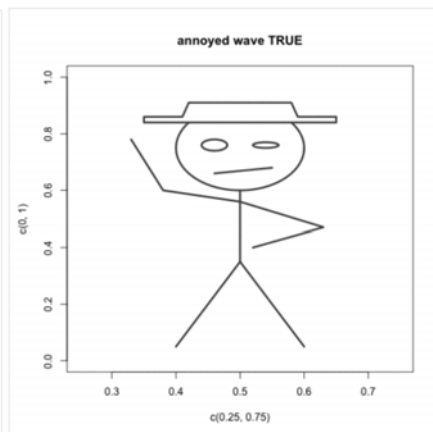
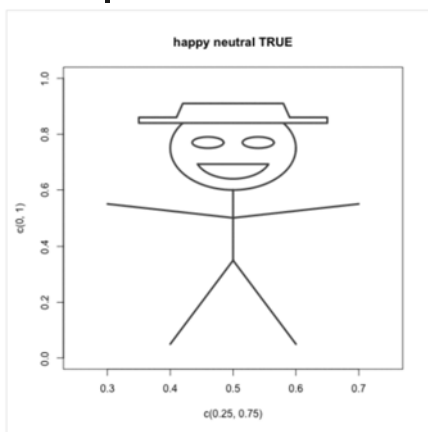
大綱

畫一個動畫

機率分布圖



畫一個動畫



- Distribution shifts according to the mean
- Visualize filtering threshold
- Rotating 3D scatter plot
- Stick figures
- <https://davetang.org/muse/2015/02/12/animated-plots-using-r/>

- Binomial Density
- <https://www.r-bloggers.com/animated-plots-with-r/>

- Happy New Year with R
- <http://sixf.org/en/2015/02/happy-new-year-with-r/>

- 21

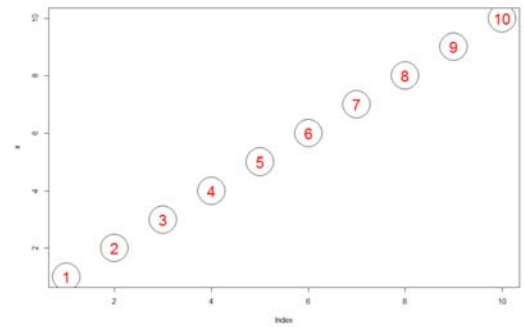
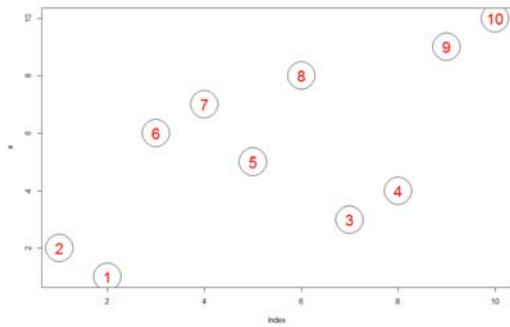
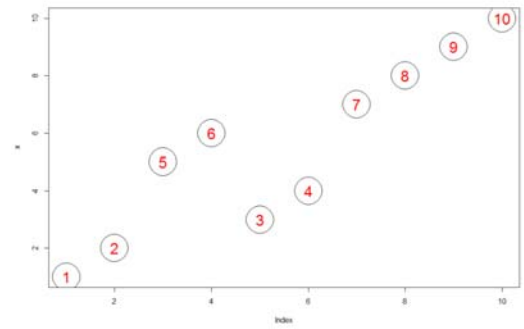
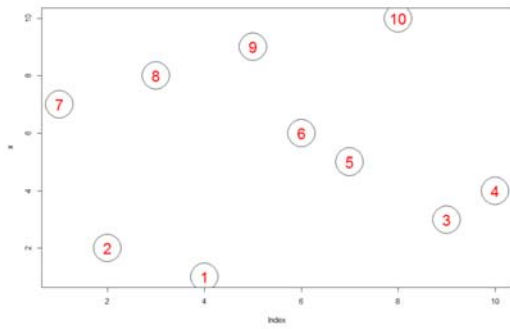
數據排序

```
■ mySort <- function( x ) {  
  itemCount <- length( x )  
  repeat {  
    hasChanged <- FALSE  
    itemCount <- itemCount - 1  
    if ( itemCount >= 1 ){  
      for( k in 1 : itemCount ) {  
        if ( x[ k ] > x[ k+1 ] ) {  
          t <- x[ k ]  
          x[ k ] <- x[ k+1 ]  
          x[ k+1 ] <- t  
          hasChanged <- TRUE  
        }  
      }  
    }  
    if ( !hasChanged ) break;  
  }  
  return( x )  
}
```

22

搜尋演算法的過程

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE



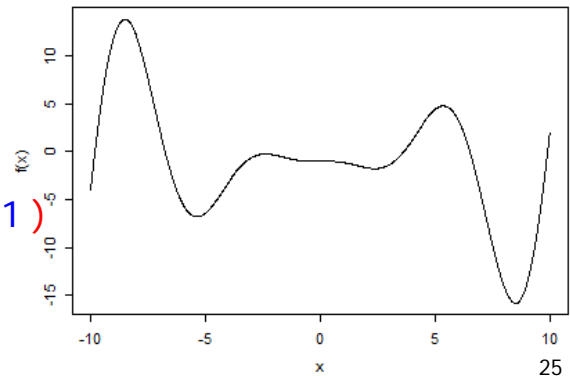
- 23

大綱

計算機程式設計 - 2017S
U11: 動作與動畫
Feng-Li Lian @ NTU-EE

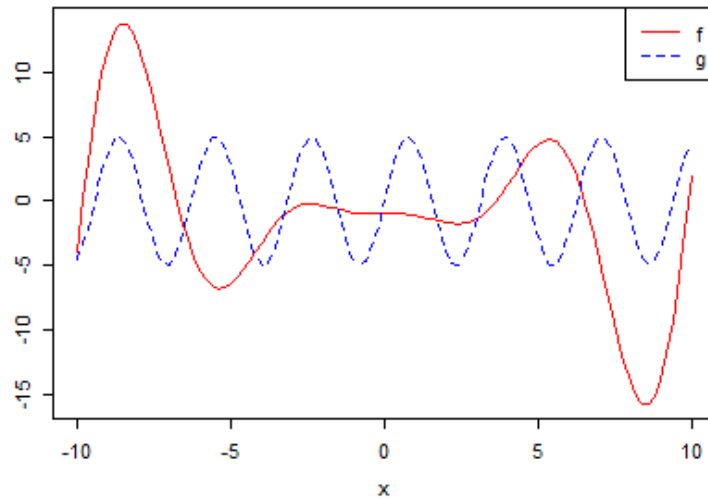
數學函數繪圖

- `curve()` # 數學函數繪圖
- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `win.graph(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `curve(f, from = -10, to = 10)`
- `curve(f, from = -10, to = 10, n = 1001)`
- `par(old.par)`



- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `g <- function(x) 10 * cos(x) * sin(x)`
- `win.graph(width = 4.5, height = 3.3, pointsize = 8)`
- `old.par <- par(mex = 0.8, mar = c(5, 5, 4, 2) + 0.1)`
- `curve(f, from = -10, to = 10, lty = 1, col = "red", ylab = "")`
- `curve(g, add = TRUE, lty = 2, col = "blue")`
- `legend("topright", legend = c("f", "g"), lty = 1:2, col = c("red", "blue"))`
- `par(old.par)`

- `f <- function(x) 0.01 * x^3 * cos(x) - 0.2 * x^2 * sin(x) + 0.05 * x - 1`
- `g <- function(x) 10 * cos(x) * sin(x)`



27

作業

28

HW10：動作與動畫

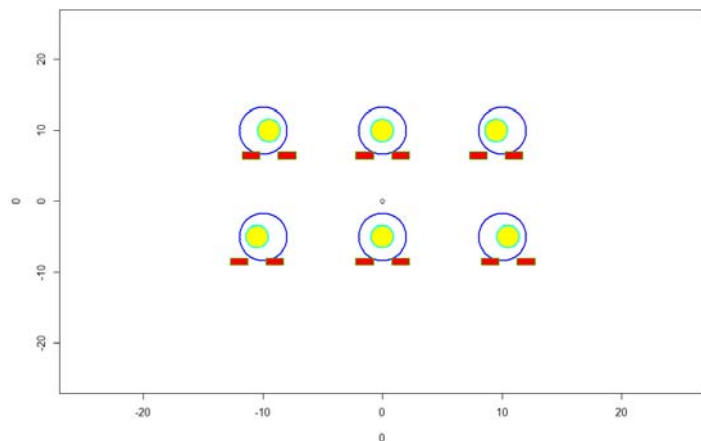
計算機程式設計 - 2017S

U11: 動作與動畫

Feng-Li Lian @ NTU-EE

On 5/16, 2017

- 參考範例程式， [HW10_B01921001_Animation.R](#)
- 使用 `Ball_0()`, `Ball_1()`, `Ball_2()`
可以繪製下圖的六個不同姿態（腳的站姿）的主角
- 也可以使用 `Ball_0_OFF()`, `Ball_1_OFF()`, `Ball_2_OFF()`
把這這個主角消除掉。
- 請參考範例程式，寫一個動畫程式，
讓這個主角自由自在地遊走在您的螢幕上！



29

HW10：動作與動畫

計算機程式設計 - 2017S

U11: 動作與動畫

Feng-Li Lian @ NTU-EE

On 5/16, 2017

- 繳交下面檔案，檔案名稱：[HW10_學號_關鍵字.xxx](#)
 - 主要指定檔案：[HW10_B01921001_Animation.R](#)
將有進行改變的哪幾行程式碼以及對應的註解解釋，
報告檔案：[HW10_B01921001_Animation.pdf](#) 或者 [.pptx](#)
程式執行之後所產生的過程或完成圖，
試著解釋您的規劃的動作為何？
或者是：[R Markdown](#) 等整合式的檔案，[.Rmd](#) 與 [.pdf](#)
- 繳交方式與期限：
 - E-mail 上面兩個檔案到：ntucp105s@gmail.com
 - E-mail 主旨：[HW10_B01921001_Animation](#)
(就是，作業編號_您的學號_關鍵字)
 - 繳交期限：**5/21 (Sun), 2017, 11pm 以前**
- 學習方式：請至下面網址輸入此次的學習方式所花的時間：
 - <https://goo.gl/L157kQ>

30