



105-1: EE4052
計算機程式設計
Computer Programming

Unit 04: 資料類別與基本運算

連 豐 力

臺大電機系

Sep 2016 - Jan 2017



大綱

計算機程式設計 - 2016F
Chap 04: 資料與運算
Feng-Li Lian @ NTU-EE

- 資料表示法
- 基本運算與函數
- 文字處理

資料表示法

3



資料表示法

計算機程式設計 - 2016F
Chap 04: 資料與運算
Feng-Li Lian @ NTU-EE

■ R程式語言的資料類別 (data type) (六類) :

- 文字 (character) "a", "b", "H", "beautiful", "cat"
- 整數 (integer) 25, -8
- 實數或數值 (numeric) 3.5824, -0.03, exp(2)
- 複數 (complex) $2 + 5i$, $-3 + 4i$
- 邏輯 (logic) TRUE, FALSE
- 原始資料 (raw)

- 4

- `x <- 3` # 1 個分量的原型向量
- `y = 3` # 1 個分量的原型向量
- `3 -> z` # 1 個分量的原型向量

- `x[1]` # x 向量的第1個分量或元素
- `x[2]` # x 向量的第2個分量或元素

- `yy <- 1:30` # 30 個分量的原型向量
- `1:30 -> zz` # 30 個分量的原型向量

- `y[1]` # y 向量的第1個分量或元素
- `y[2]` # y 向量的第2個分量或元素
- `y[18]` # y 向量的第18個分量或元素

- `y[35]` # y 向量的第35個分量或元素

- `c()` # 連接函數，(concatenation)，
連接若干向量或元素

- `c(3, 7, 5)` # 一個向量，放了 3, 7, 5 的三個分量或元素

- `x <- c(3, 7, 5)`

- `y <- beautiful`

- `y <- "beautiful"`

- `y <- c("beautiful", "handsome")`

- `z <- c(x, y)`

- `class()` # 資料類別, (class)
- `class(x)`
- `class(y)`
- `class(z)`
- `class(z[1])`
- `class(z[1:3])`

- 7

- `seq()`
- `seq(from = 1, to = 1, by = ((to - from)/(length.out - 1)), length.out = NULL, along.with = NULL, ...)`
- `seq(from = 1, to = 9, by = 2)` # matches 'end'
- `seq(1, 9, by = 2)` # matches 'end'
- `seq(from = 0, to = 1, length = 11)`
- `seq(from = 0, to = 1, by = 0.1)`
- `seq(from = 1.575, to = 5.125, by = 0.05)`
- `seq(from = 1, to = 6, by = 3)`
- `seq(from = 1, to = 9, by = pi)` # stays below 'end'
- `seq(17)` # same as 1:17, or even better `seq_len(17)`

- 8

- `rep()`
- `rep(x, times = ?)`
- `rep(x, length = ?)`

- `rep(3, times = 5)`
- `rep(1:4, 2)`

- `rep(1:4, each = 2)` # not the same.
- `rep(1:4, c(2,2,2,2))` # same as second.
- `rep(1:4, c(2,1,2,1))`

- `rep(1:4, each = 2, length = 4)` # first 4 only.
- `rep(1:4, each = 2, length = 10)` # 8 integers plus two recycled 1's.

- `rep(1:4, each = 2, times = 3)` # length 24, 3 complete replications

- `is.numeric()`
- `is.integer()`
- `is.character()`

- `x <- c(2, 4, 6)`

- `is.numeric(x)`
- `is.integer(x)`
- `is.character(x)`

- `class(x)`

- `class(c(2, 5, 8))`
- `class(c("beautiful", "handsome"))`
- `class(c(2, 5, "beauty"))`

- `as.numeric()`
- `as.integer()`
- `as.character()`

- # 產生相同的實數類別物件
- `u1 <- c(2.2, 4.4, 6.6)`
- `u2 <- as.numeric(c(2.2, 4.4, 6.6))`
- `u3 <- as.vector(c(2.2, 4.4, 6.6), mode = "numeric")`

- # 產生相同的實數類別物件
- `v1 <- c(2, 4, 6)`
- `v2 <- as.numeric(c(2, 4, 6))`
- `v3 <- as.vector(c(2, 4, 6), mode = "numeric")`

- `as.numeric()`
- `as.integer()`
- `as.character()`

- # 產生相同的整數類別物件
- `x1 <- as.integer(c(2, 4, 6))`
- `x2 <- as.vector(c(2, 4, 6), mode = "integer")`

- # 產生相同的文字類別物件
- `y1 <- c("beautiful", "handsome")`
- `y2 <- as.character(c("beautiful", "handsome"))`
- `y3 <- as.vector(c("beautiful", "handsome"), mode = "character")`

- `as.numeric()`
- `as.integer()`
- `as.character()`

- # 不同類別物件的轉換，有的可以，有的不行
- `as.integer(c(-1.2, 0, 3.4))`

- `as.character(c(2, 4, 6))`

- `as.integer(c("beautiful", "cool", "handsome"))`

- # 下面四種方法，可以產生相同的內容

- `x <- c(1.2, 3.4, 5.6)`

- `y <- as.numeric(c(1.2, 3.4, 5.6))`

- `z <- numeric()`
- `z[1:3] <- c(1.2, 3.4, 5.6)`

- `w <- 1.2`
- `w[2] <- 3.4`
- `w[3] <- 5.6`

- 流程控制：
 - function
 - if
 - else
 - repeat
 - while
 - for
 - in
 - next
 - break
- 特殊符號：
 - TRUE
 - FALSE
 - NULL
 - Inf
 - NaN Not a Number
 - NA Not Available
 - NA_integer_
 - NA_real_
 - NA_complex_
 - NA_character
- 內建常數：
 - pi
 - LETTERS
 - letters
 - month.abb
 - month.name

- 0 / 0
- 1 / 0
- -1 / 0

- x <- c(1, NA, 3)
- x[2]
- x[2] + x[2]
- x[1] + x[2]

- a <- Inf; b <- Inf
- a + b
- a - b
- a * b
- a / b

基本運算與函數

17



基本數學運算子

計算機程式設計 - 2016F
Chap 04: 資料與運算
Feng-Li Lian @ NTU-EE

- \wedge # 次方
- $+$ $-$ # 加減運算
- $*$ $/$ # 乘除運算
- $+$ $-$ # 單運算元，取正或取負
- $\% / \%$ # 整數除法 取商
- $\% \%$ # 整數除法 取餘數

- 18

- $3.6 + 1.25$

- $x <- 3.6; y <- 1.25$
- $x + y$
- $x - y$
- $x * y$
- x / y

- $x <- 1:10; y <- 1:10 * 2$
- x
- y
- $x + y$
- $y - x$
- $(y - x) / 2$

- $a <- 5; b <- 2$

- $a \%/% b$
- $a \% b$

- $x <- 2; y <- 0:4$
- x
- y
- x^y
- y^x

表 4.1: 常見數學函式

函式	說明
?	Help
<-	Left assignment, binary
->	Right assignment, binary
\$	List subset, binary
:	Sequence, binary
:	In model formulae: interaction
~	Tilde, used for model formulae, can be either unary or binary
-	Substraction, can be unary or binary
+	Addition, can be unary or binary
!	Unary not
*	Multiplication, binary
/	Division, binary
^	Exponentiation, binary
%%	Modulus, binary
%/%	Integer divide, binary
round(x, digits = 0)	its first argument to the specified number of decimal places
signif(x, digits = 6)	rounds the values to the specified number of significant digits
trunc(x)	the integers by truncating 'x' toward '0'
ceiling(x)	the smallest integers not less than 'x'
floor(x)	the largest integers not greater than 'x'
sign(x)	$sign(x)$, the sign of a real number is 1, 0, or -1
	if the number is positive, zero, or negative, respectively.
abs(x)	$ x $, absolute value of x
sqrt(x)	\sqrt{x}
exp(x)	e^x
expm1(x)	computes $\exp(x) - 1$ accurately also for $ x \ll 1$.
log(x)	$\log(x)$
log10(x)	$\log_{10}(x)$
log2(x)	$\log_2(x)$
logb(x, base = z)	$\log_z(x)$
log1p(x)	computes $\log(1 + x)$ accurately also for $ x \ll 1$.

資料來源：
 台北大學統計學系林建甫教授

gamma(x)	$\Gamma(x) = (x-1)! = \int_0^\infty t^{(x-1)} \exp(-t) dt$
lgamma(x)	$\log_e[\Gamma(x)]$
beta(a, b)	$B(a, b) = (\Gamma(a)\Gamma(b)) / (\Gamma(a+b)) = \int_0^1 t^{(a-1)}(1-t)^{(b-1)} dt$
lbeta(a, b)	$\log_e[B(a, b)]$
digamma(x)	$\frac{d}{dx} \log_e[\Gamma(x)]$
trigamma(x)	$\frac{d^2}{dx^2} \log_e[\Gamma(x)]$
psigamma(x, deriv = 0)	$\frac{d^p}{dx^p} \log_e[\Gamma(x)]$
choose(n, k)	$\frac{n!}{k!(n-k)!}$
lchoose(n, k)	$\log_e(\text{choose}(n, k))$
factorial(x)	$x! = \Gamma(x+1)$
lfactorial(x)	$\log(x!) = \log_e[\Gamma(x)]$
sin(x) cos(x) tan(x)	trigonometric functions
asin(x) acos(x) atan(x)	inverse functions
sinh(x) cosh(x) tanh(x)	hyperbolic functions
asinh(x) acosh(x) atanh(x)	inverse hyperbolic functions

資料來源：
 台北大學統計學系林建甫教授

表 4.2: 常見之排序函式

函式	說明
rev(x)	reverse order
rank(x)	Returns the sample ranks of the values Default argument "ties.method" = "average"
sort(x)	Sort a vector or factor (partially) into ascending (or descending) order.
order(x)	Returns a permutation which rearranges its first argument into ascending or descending order, breaking ties by further arguments.

表 4.4: 常見敘述性統計函式

格式	說明	
sum(x)	summation	$y = \sum_i x_i$
cumsum(x)	cumulative sum	$z_j = \sum_{i \leq j} x_i$
diff(x)	$x[i+1] - x[i]$	$z_i = x_{i+1} - x_i$
prod(x)	product	$y = \prod_i x_i$
cumprod(x)	cumulative product	$z_j = \prod_{i \leq j} x_i$
mean(x)	mean	$\bar{x} = \frac{1}{n} \sum_i x_i$
median(x)	median	0.5 quantile, 50 th percentile
var(x)	variance, covariance	$s^2 = \frac{1}{n-1} \sum_i (x_i - \bar{x})^2$
sd(x)	standard deviation	$s = \sqrt{s^2}$
range(x)	range	$[\min(x), \max(x)]$
min(x)	minimum	
max(x)	maximum	
quantile(x)	percentile	
fivenum(x)	five-number summary	$[\max, Q_1, \text{median}, Q_3, \max]$
sample(x)	random sample	

資料來源：
 台北大學統計學系林建甫教授

- log(10)
- log10(100)
- log2(8)
- exp(1.2)

- `x <- 0:7 * pi / 16`
- `sx <- sin(x)`
- `x1 <- asin(sx)`

- `x1 - x`

- `cx <- cos(x)`
- `x2 <- acos(cx)`
- `x2 - x`

- `tx <- tan(x)`

- `x3 <- atan(tx)`
- `x3 - x`

- `factorial(4)`

- `ceiling(2.55)`

- `floor(2.55)`

- `trunc(2.55)`

- `round(3.555, digits = 2)`

- `signif(3.555, digits = 2)`

- `x <- c(1.2, 3.5, -4.7, 0)`

- `min(x)`
- `max(x)`
- `range(x)`

- `length(x)`

- `sum(x)`
- `mean(x)`
- `sd(x)`
- `var(x)`
- `median(x)`

- `x1 <- 1:6`
- `x2 <- rep(4, 6)`
- `x3 <- 6:1`

- `min(x1, x2, x3)`
- `max(x1, x2, x3)`
- `pmin(x1, x2, x3)`
- `pmax(x1, x2, x3)`

- `x <- c(1.22, 3.555, -4.75, 0)`
- `ceiling(x)`
- `floor(x)`
- `trunc(x)`
- `round(x, digits = 1)`
- `signif(x, digits = 2)`

- `x <- c(1.22, 2.55, 3.78, -4.75)`
- `x[2]`
- `x[c(1, 3)]`
- `x[-2]`
- `x[-c(2, 4)]`
- `x[-length(x)]`
- `x[c(1, 0, 2)]`
- `x[c(-1, 0, -2)]`
- `x[c(1, 0, -2)]`
- `x[c(1.2, 3.4)]`
- `x[c(-1.2, -3.4)]`

- $x \leftarrow c(1, 0, -1, 2)$
- $y \leftarrow c(-2, 1, 0, -2)$

- $x * y$

- $2 * x$
- $x * 2$
- $c(2, 2, 2, 2) * x$

- $x * (1:2)$
- $x * c(1, 2, 1, 2)$

- $\text{sum}(x * y)$

文字處理

- `look <- c("beautiful", "handsome", "cool")`
- `substr(look, start = 1, stop = 4)`
- `substring(look, first = 1, last = 4)`
- `look.more <- c(look, "pretty")`
- `look.more`
- `paste(look, "people")`
- `paste("They look like", look, "people", collapse = ". ")`

- `letters[1:10]`
- `LETTERS[1:10]`
- `month.name[1:12]`
- `month.abb[1:12]`

作業

35

HW03：資料類別與基本運算

計算機程式設計 - 2016F
Chap 04: 資料與運算
Feng-Li Lian @ NTU-EE

On 10/11, 2016

- 編輯一組程式於 .R 檔，以完成下面結果：
 - 產生一個數列 i ，從 1 到 10 的 10 個數值
 - 產生一個數列 t ，從 0.1 到 1.0 的 10 個數值
 - 產生一個數列 x ，從 1 到 10 的幾個奇數數值
 - 產生一個數列 y ，從 1 到 10 的幾個偶數數值
 - 讓 t 中的每一個數值，乘上 3
 - 計算 t 的最大值與最小值，平均值，標準差，範圍，與數列長度
 - 讓 x 中的第一個數值，乘上 y 中的第一個數值，
讓 x 中的第二個數值，乘上 y 中的第二個數值，以此類推，
就是讓每一個 x 乘上對應的每一個 y
 - 計算 i 中的每一個數值取 $\log_2()$ ，
計算 t 中的每一個數值乘上 $2 * \pi$ ，之後的每一個數值再取 $\cos()$ ，
最後，讓每一個 i 取完 $\log_2()$ 的數值，再乘上對應的每一個 t ，乘上 $2 * \pi$ ，取
完 $\cos()$ 之後的數值
 - 建立一個數列：`mymonth`，包含 12 個月的英文單字，
然後，利用 `substring`, `paste` 的函數，取出前三個字母，並加上句點 "."，
變成月份的縮寫形式，指定到：`mymonthshort`
- 把執行的過程，以及產生的數據等，整理到報告檔 (pptx or pdf)。

HW03：資料類別與基本運算

計算機程式設計 - 2016F

Chap 04: 資料與運算

Feng-Li Lian @ NTU-EE

On 10/11, 2016

- 繳交下面檔案，檔案名稱：[HW03_學號](#)
 - 主要指定檔案：[HW02_B01921001.R](#)
 - 報告檔案：[HW02_B01921001.pptx](#) 或者 [.pdf](#)
- 繳交方式與期限：
 - E-mail 上面四個檔案到：ntucp105f@gmail.com
 - E-mail 主旨：[HW03_B01921001](#) (就是，作業編號_您的學號)
 - 繳交期限：[10/12 \(Wed\), 2016, 11pm 以前](#)
 - [HW03](#)，每位繳交自己的作業，或者兩人繳交一份
 - 兩人完成作業者，請註明工作模式，例如：
 - A 進行指令測試與報告編輯，然後給 B 重複測試
 - A 進行指令測試，B 進行報告編輯
 - A 進行指令測試與報告編輯，B 僅看過報告，但沒有實際測試
 - A 進行指令測試與報告編輯，B 沒有參與指令測試，也沒有看過報告