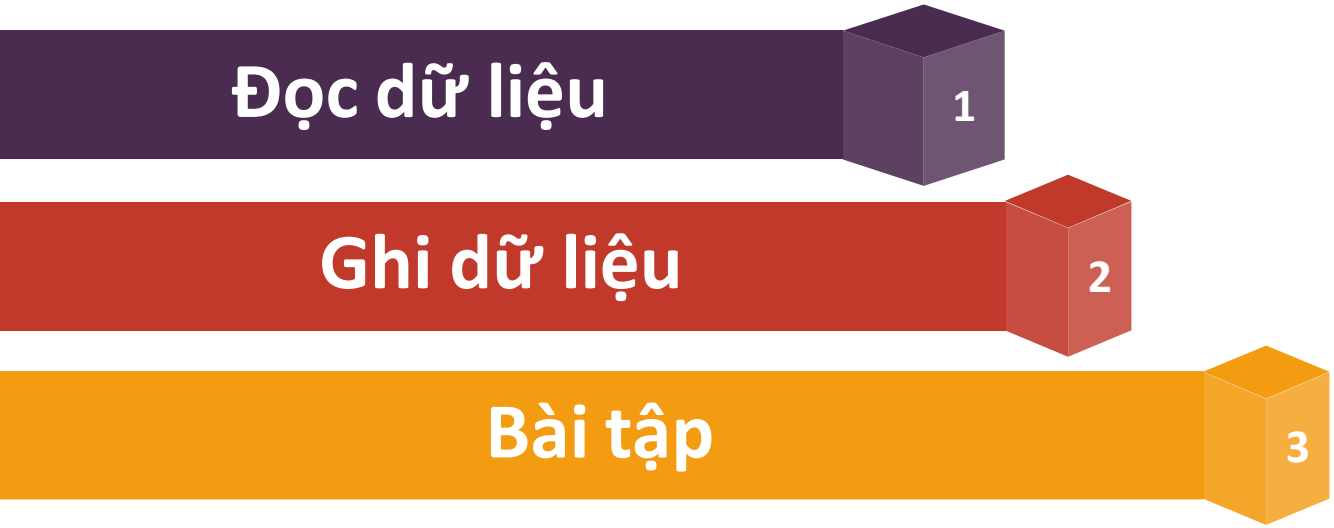
KHOA CÔNG NGHỆ THÔNG TIN KINH DOANH
School of Business Information Technology

Lập trình căn bản (ngôn ngữ R)

Fundamentals of Programming (R Programming)

Bài 4. Đọc ghi dữ liệu trong ngôn ngữ R

Nội dung



1. Đọc/ghi dữ liệu

- Một số chức năng đọc chính:
  - `read.table, read.csv`, for reading tabular data
  - `readLines`, for reading lines of a text file
  - `source`, for reading in R code files (inverse of `dump`)
  - `dget`, for reading in R code files (inverse of `dput`)
  - `load`, for reading in saved workspaces
  - `unserialize`, for reading single R objects in binary form

1. Đọc/ghi dữ liệu

- Một số chức năng ghi chính:
  - *write.table*, for writing tabular data to text files (i.e. CSV) or connections
  - *writeLines*, for writing character data line-by-line to a file or connection
  - *dump*, for dumping a textual representation of multiple R objects
  - *dput*, for outputting a textual representation of an R object
  - *save*, for saving an arbitrary number of R objects in binary format (possibly compressed) to a file.
  - *serialize*, for converting an R object into a binary format for outputting to a connection (or file)

2. Làm việc với tài liệu

- File text
- File csv
- File binary
- File json
- File xml
- File excel
- Webdata

2.1. File text: đọc file

```
## Create a connection to 'foo.txt'
con <- file("foo.txt")

## Open connection to 'foo.txt' in read-only
mode
open(con, "r")

## Read from the connection
 data <- read.csv(con)

## Close the connection
close(con)
```

- "r" read only mode
- "w" writing (and initializing a new file)
- "a" open a file for appending
- "rb", "wb", "ab" reading, writing, or appending in binary mode (Windows)

2.1. File text: đọc file readlines

```
## Create a connection to 'foo.txt'
con <- file("foo.txt")

## Open connection to 'foo.txt' in read-only
mode
open(con, "r")

## Read from the connection
n <- readLines(con, 1)
x <- readLines(con, n)

## Close the connection
close(con)
```

- "r" read only mode
- "w" writing (and initializing a new file)
- "a" open a file for appending
- "rb", "wb", "ab" reading, writing, or appending in binary mode (Windows)

2.1. File text: ghi file

```
url <- "D://MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"text.txt",sep="")
output <- paste(url,"out.txt",sep="")
## Create a connection to 'text.txt'
con <- file(input)

## Open connection to 'text.txt' in read-only mode
open(con, "r")

## Read from the connection
data <- read.csv(con)
min_data <- min(data)

## write data
# write.csv(min_data,output,row.names = FALSE)
write.table(min_data, output, sep=",",  col.names=FALSE, row.names =
FALSE)


## Close the connection
close(con)
```

2.1. File text: ghi file writelines

```
url <- "D://MonHoc//Lap trinh can ban R//Demo//"

input <- paste(url,"text.txt",sep="")

output <- paste(url,"out.txt",sep="")

## Create a connection to 'text.txt'

con <- file(input)
## Open connection to 'text.txt' in read-only mode
open(con, "r")

## Read from the connection
data <- read.csv(con)
min_data <- data > 0.6

## write data
writeLines(paste(data[min_data]), output)

## Close the connection
close(con)
con_output <- file(output)
open(con_output,  "a")
writeLines("End of file", con_output)
close(con_output)
```

2.2. File csv: *datatable.csv*

```
 1 ,t1,t2,t3,t4,t5,t6,t7,t8
 2 r1,1,0,1,0,<NA>,1,0,2
 3 r2,1    ,2,2,1,2,1,2,1
 4 r3,0,0,0,2,1,1,0,1
 5 r4,<NA>,0,1,1,2,<NA>,0,0
 6 r5,0,2,1,1,1,0,0,0
 7 r6,2,2,0,1,1,1,0,0
 8 #The following line is blank,,,,,,,,
 9 r7,2,2,0,1,1,1,0,1
10 r8,0,2,1,0,1,1,2,0
11 r9,1,<NA>,1,2,<NA>,1,0,1
12 r10, 1                  ,0,2,1,2,2,1,<NA>
```

2.2. File csv: đọc file

```
read.table(file, header = FALSE, sep = "", quote = "\"'",
           dec = ".", numerals = c("allow.loss", "warn.loss", "no.loss"),
           row.names, col.names, as.is = !stringsAsFactors,
           na.strings = "NA", colClasses = NA, nrows = -1,
           skip = 0, check.names = TRUE, fill = !blank.lines.skip,
           strip.white = FALSE, blank.lines.skip = TRUE,
           comment.char = "#",
           allowEscapes = FALSE, flush = FALSE,
           stringsAsFactors = default.stringsAsFactors(),
           fileEncoding = "", encoding = "unknown", text, skipNul = FALSE)

read.csv(file, header = TRUE, sep = ",", quote = "\"",
         dec = ".", fill = TRUE, comment.char = "", …)

read.csv2(file, header = TRUE, sep = ";", quote = "\"",
          dec = ",", fill = TRUE, comment.char = "", …)
```

https://www.rdocumentation.org/packages/utils/versions/3.6.2/topics/read.table

2.2. File csv: đọc file

```
url <- "D://Demo//datatable.csv"
d1 <- read.table(url, header=FALSE, sep=",")
print(d1)

d2 <- read.table(url, header=TRUE, sep="
,strip.white=TRUE)
print(d2)

d3 <- read.csv(url, row.names = 1, header=TRUE,
sep="," ,strip.white=TRUE)
print(d3)

d4 <- read.csv(url)
print(d4)
```

2.2. File csv: *phân tích csv*

```
url <- "D://Demo//datatable.csv"
data <- read.csv(url)

# Kiểm tra dataframe
print(is.data.frame(data))

# Xem tổng số dòng, số cột
print(ncol(data))
print(nrow(data))

# Lấy phần tử dòng 4, cột 2
data[[4,2]]
data$t1[4]

# Lấy phần tử lớn nhất
max(data1$t1)
```

2.2. File csv: *input.csv*

```
1 id,name,salary,start_date,dept
2 1,Rick,623.3,2012-01-01,IT
3 2,Dan,515.2,2013-09-23,Operations
4 3,Michelle,611,2014-11-15,IT
5 4,Ryan,729,2014-05-11,HR
6 5,Gary,843.25,2015-03-27,Finance
7 6,Nina,578,2013-05-21,IT
8 7,Simon,632.8,2013-07-30,Operations
9 8,Guru,722.5,2014-06-17,Finance
```

2.2. File csv: phân tích *input.csv*

```
# Create a data frame.
data <- read.csv("input.csv")

# Get the max salary from data frame.
sal <- max(data$salary)

# Get the person detail having max salary.
retval <- subset(data, salary == max(salary))
print(retval)

# Get all the people working in IT department
retval <- subset(data, dept == "IT")
print(retval)
```

2.2. File csv: phân tích *input.csv*

```
# Create a data frame.
data <- read.csv("input.csv")

# Get the max salary from data frame.
sal <- max(data$salary)

# Get the persons in IT department whose salary is
greater than 600
retval <- subset(data, salary > 600 & dept ==
"IT"))
print(retval)

# Get the people who joined on or after 2014
retval <- subset(data, as.Date(start_date) >
as.Date("2014-01-01"))
print(retval)
```

2.2. File csv: ghi file

```
url <- "D://Demo//"
# Create a data frame.
input <- paste(url,"input.csv",sep="")
output <- paste(url,"output.csv",sep="")

data <- read.csv(input)
retval <- subset(data,
as.Date(start_date) > as.Date("2014-01-01"))

# Write filtered data into a new file.
write.csv(retval,output)
# write.csv(retval,"output.csv", row.names =
FALSE)
newdata <- read.csv(output)
print(newdata)
```

2.3. File binary: ghi file

```
url <- "D://MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"mtcars.csv",sep="")
output1 <- paste(url,"mtcars_1.csv",sep="")
output2 <- paste(url,"binmtcars.dat",sep="")

data <- read.csv(input, header=TRUE, sep="," ,strip.white=TRUE)
mtcars <- data[c('cyl','am','gear')]
# Read the "mtcars" data frame as a csv file and store only the columns "cyl", "am"
and "gear".
write.table(mtcars, file = output1,row.names = FALSE, na = "",
   col.names = TRUE, sep = ",")

# Store 5 records from the csv file as a new data frame.
new.mtcars <- read.table(output1,sep = ",",header = TRUE,nrows = 5)

# Create a connection object to write the binary file using mode "wb".
write.filename = file(output2, "wb")

# Write the column names of the data frame to the connection object.
writeBin(colnames(new.mtcars), write.filename)

# Write the records in each of the column to the file.
writeBin(c(new.mtcars$cyl,new.mtcars$am,new.mtcars$gear), write.filename)

# Close the file for writing so that it can be read by other program.
close(write.filename)
```

2.3. File binary: đọc file

```
url <- "D://MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"binmtcars.dat",sep="")

# Create a connection object to read the file in binary mode
using "rb".
read.filename <- file(input, "rb")

# First read the column names. n = 3 as we have 3 columns.
column.names <- readBin(read.filename, character(),  n = 3)

# Next read the column values. n = 18 as we have 3 column names
and 15 values.
bindata <- readBin(read.filename, integer(),  n = 15)

# Print the data.
print(bindata)
```

2.3. File binary: đọc file

```
# Read the values from 1th byte to 5th byte which represents
"cyl".
cyldata = bindata[1:5]
print(cyldata)

# Read the values form 6th byte to 10th byte which represents
"am".
amdata = bindata[6:10]
print(amdata)

# Read the values form 11th byte to 15th byte which represents
"gear".
geardata = bindata[11:15]
print(geardata)

# Combine all the read values to a dat frame.
finaldata = cbind(cyldata, amdata, geardata)
colnames(finaldata) = column.names
print(finaldata)
```

2.4. File json: đọc file

- Sử dụng console để cài thư viện

```
install.packages("rjson")
```

- Mã nguồn

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"input.json",sep="")

# Load the package required to read JSON files.
library("rjson")

# Give the input file name to the function.
result <- fromJSON(file = input)

# Print the result.
print(result)
```

2.4. File json: đọc file dataframe

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban
R//Demo//"
input <- paste(url,"input.json",sep="")

# Load the package required to read JSON files.
library("rjson")

# Give the input file name to the function.
result <- fromJSON(file = input)

# Convert JSON file to a data frame.
json_data_frame <- as.data.frame(result)

print(json_data_frame)
```

2.5. File xml: đọc file

- Sử dụng console để cài thư viện

```
install.packages("XML")
```

- Mã nguồn

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"input.xml",sep="")

# Load the package required to read XML files.
library("XML")

# Also load the other required package.
library("methods")

# Give the input file name to the function.
result <- xmlParse(file = input)

# Print the result.
print(result)
```

2.5. File xml: đọc file lấy số lượng node

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban
R//Demo//"
input <- paste(url,"input.xml",sep="")

# Load the packages required to read XML files.
library("XML")
library("methods")

# Give the input file name to the function.
result <- xmlParse(file = input)

# Exract the root node form the xml file.
rootnode <- xmlRoot(result)

# Find number of nodes in the root.
rootsize <- xmlSize(rootnode)

# Print the result.
print(rootsize)
```

2.5. File xml: lấy node

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh  can ban R//Demo//"
input <- paste(url,"input.xml",sep="")

# Load the packages required to read XML files.
library("XML")
library("methods")

# Give the input file name to the function.
result <- xmlParse(file = input)

# Exract the root node form the xml file.
rootnode <- xmlRoot(result)

# Print the result.
print(rootnode[1])

# ID
print(rootnode[[1]][[1]])

# value of ID
print(rootnode[[1]][[1]][[1]])
```

2.5. File xml: dataframe

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban
R//Demo//"
input <- paste(url,"input.xml",sep="")

# Load the packages required to read XML files.
library("XML")
library("methods")

# Convert the input xml file to a data frame.
xmldataframe <- xmlToDataFrame(input)
print(xmldataframe)
```

2.6. File excel: đọc file

https://www.rdocumentation.org/packages/readxl/versions/0.1.1/topics/read_excel

- Sử dụng console để cài thư viện

```
install.packages("readxl")
```

- Mã nguồn

```
url <- "D://Drive nguyentt@UEH//MonHoc//Lap trinh can ban R//Demo//"
input <- paste(url,"input.xlsx",sep="")

# Load the library into R workspace.
library("readxl")

# Specify sheet by its name
my_data1 <- read_excel(input, sheet = 1)

# Specify sheet by its index
my_data2 <- read_excel(input, sheet = "city")

print(my_data1)
print(my_data2)
```

2.7. Webdata: đọc file xml

- Sử dụng console để cài thư viện

```
install.packages("httr")
```

- Mã nguồn

```
require(httr)
UA <- "Mozilla/5.0 (Windows NT 6.1; WOW64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2227.0 Safari/537.36"
my_url <- "https://vnexpress.net/rss/tin-moi-nhat.rss"
doc <- GET(my_url, user_agent(UA))

# Load the package required to read XML files.
library("XML")

# Also load the other required package.
library("methods")

# Give the input file name to the function.
result <- xmlParse(content(doc, "text"))

xmldataframe <- xmlToDataFrame(xmlRoot(result)[[1]][3])
print(xmldataframe)
```

2.7. Webdata: đọc file json

```
require(httr)
UA <- "Mozilla/5.0 (Windows NT 6.1; WOW64)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/41.0.2227.0
Safari/537.36"
my_url <-
"https://gist.githubusercontent.com/soxjke/63fcc6e39e294f9
1c855d9aa2f958bc2/raw/12d87132a635d40d3d4d959038fca7d8cacb
7e6b/Weather.json"
doc <- GET(my_url, user_agent(UA))

# Load the package required to read JSON files.
library("rjson")

# Give the input file name to the function.
result <- fromJSON(content(doc, "text"))

# Print the result.
print(result)
```

3. Bài tập

1. Tạo trước file csv chứa bốn cột tên STT, Toan, Ly, Hoa. Mỗi dòng là 3 điểm của học sinh. Tổng cộng có 5 học sinh.
   Viết chương trình đọc file csv trên. Hãy tính điểm trung bình 3 môn của mỗi học sinh, điểm trung bình mỗi môn. Lưu theo định dạng file csv mới như sau.

Input

	A	B	C	D
1	STT	Toan	Ly	Hoa
2	1	2.8	6.3	4.5
3	2	4.2	2.7	5.7
4	3	1.9	7.7	1
5	4	6	2	5.7
6	5	8.5	1.5	5.4

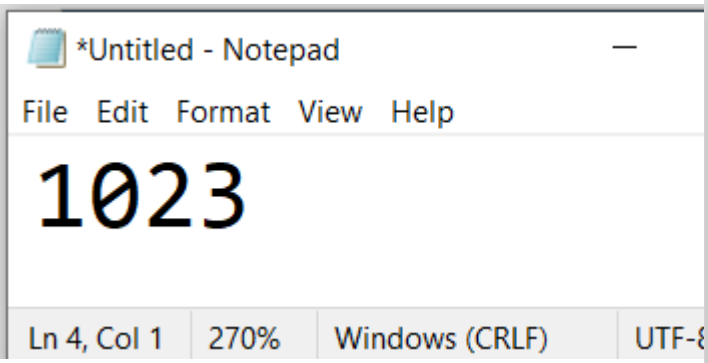
Output

	A	B	C	D	E
1	STT	Toan	Ly	Hoa	DTB hoc sinh
2	1	2.8	6.3	4.5	4.5
3	2	4.2	2.7	5.7	4.2
4	3	1.9	7.7	1	3.5
5	4	6	2	5.7	4.6
6	5	8.5	1.5	5.4	5.1
7	TB Mon	4.7	4	4.5	

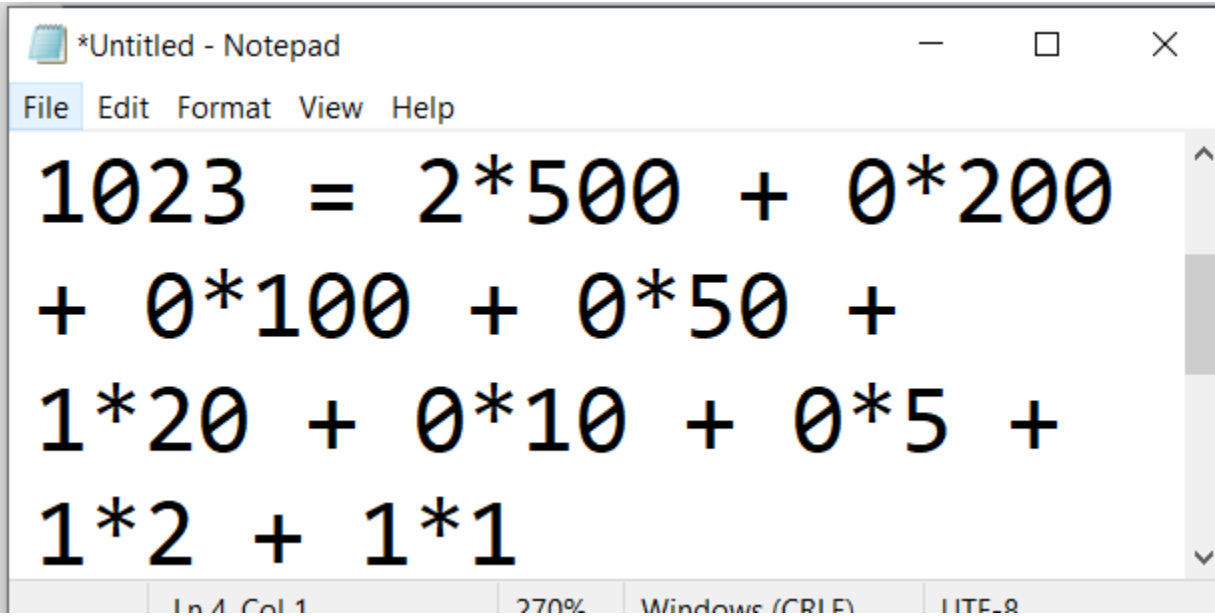
3. Bài tập

2. Tạo file text có 2 dòng, mỗi dòng là một số.
Hãy tính số tiền trên sẽ có bao nhiêu tờ tiền có mệnh giá 500, 200, 100, 50, 20, 10, 2, 1 và lưu vào file text mới theo định dạng mới (Ưu tiên tiền mệnh giá cao hơn)

Input



Output



3. Bài tập

3. Sử dụng mã nguồn lấy rss (file xml) từ trang báo Vnexpress. Lấy ra 10 tin đầu tiên lưu thành file csv theo định dạng phía dưới.

	title	pubDate	link
1	Dị ứng kháng sinh có nên tiêm vaccine Covid-19?	Mon, 30 Aug 2021 01:00:00 +0700	https://vnexpress.net/di-ung-khang-sinh-co-nen-tiem-vacci...
2	Concept ôtô màn hình khắp nơi của Hyundai	Mon, 30 Aug 2021 00:27:00 +0700	https://vnexpress.net/concept-oto-man-hinh-khap-noi-cua-...
3	Man Utd thoát hiểm ở Wolverhampton	Mon, 30 Aug 2021 00:23:45 +0700	https://vnexpress.net/man-utd-thoat-hiem-o-wolverhampto...
4	Đồ cưới đã sắm hết mà tôi còn nhiều băn khoăn	Mon, 30 Aug 2021 00:08:42 +0700	https://vnexpress.net/do-cuoi-da-sam-het-ma-toi-con-nhie...

THANK YOU!

QUESTIONS?