# Hands-on Experiment # 5: Worksheet

# (Do not submit this document file)

To submit this homework, zip all your .java files into 1 zip file and submit as attachment on MyCourseville assignment page. Make sure you have completely uploaded your file!

### Part A: Practice the use of decision in Java (10 marks)

Write your own class *MyMath.java*, implementing the working of function f that satisfies an equation below (use int as input parameter type) using if statements. The value of x is received from the keyboard. The result (as double) is then calculated and printed on screen.

$$f(x) = \begin{cases} x+2 & x \le 0 \\ 13 & x == 1 \\ x^2 - 2x + 3 & 1 < x \le 10 \\ x^3 - 3x^2 & otherwise \end{cases}$$

An example run is shown below:

E:\Dropbox\teaching\2190101\2020\Lab\Lab05\_2020\Solution>java MyMath Enter value of x.
-2

The result is 0.0.

The table below shows sample answers:

VALUE OF X	VALUE OF RESULT
-5	-3.0
1	13.0
3	6.0
9	66.0
11	968.0

## Part B: Implementing Switch statement (10 marks)

You are to write a character encoder, which reads a character and prints an output character according to the following table:

input character	output
а	В
А	В
b	С
В	С
С	D
С	D

other character	Z

Example: if a user inputs 'a', your program will print out 'B'.

if a user inputs 'E', your program will print out 'Z'.

Assume that the user always types in 1 character when a scanner is used, create and write your program in **Encoder.java**.

#### YOU MUST USE THE SWITCH STATEMENT, OTHERWISE THE SCORE IS 0!

HINT: you can use s.chatAt(i) to return the i<sup>th</sup> character inside the string s. The first character is at position 0<sup>th</sup>.

Example runs are shown below:

# Part C: Rock, paper, scissors (10 marks)

You are to write a program that simulate a game of rock-paper-scissors. Your program should

- a. ask for your move (to be typed in from keyboard).
- b. the legal moves are only "R", "P", and "S"("r", "p", and "s" also work). If illegal move is read, print message "Illegal move, please re-run the program" and exit the program.
- c. ask for the opponent's move (to be typed in from keyboard).
- d. If illegal move is read, print message "Illegal move, please re-run the program" and exit the program.
- e. Then the program evaluates the moves and print "You win", "You lose", or "Draw" according to this rule:
  - i. same moves will result in a draw
  - ii. "R" beats "S"
  - iii. "P" beats "R"
  - iv. "S" beats "P"

Here is a sample screenshot when an illegal move is read:

```
E:\Dropbox\teaching\2190101\2020\Lab\Lab05_2020\Solution>java RPS
What is your move?
p
What is the opponent's move?
d
Illegal move, please re-run the program
```

Here is a sample screenshot when moves are legal:

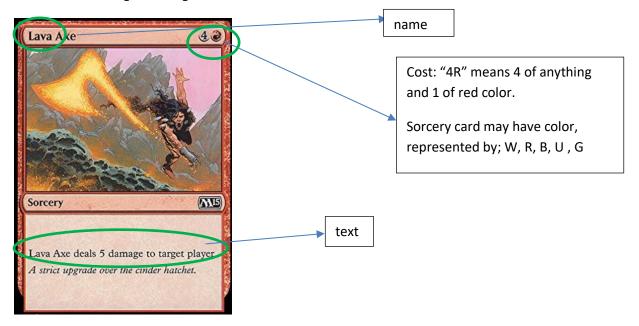
```
E:\Dropbox\teaching\2190101\2020\Lab\Lab05_2020\Solution>java RPS
What is your move?
p
What is the opponent's move?
s
You lose
```

Write your program as RPS.java It must give correct result for all possible combinations of inputs.

# Part C: Comparing Card in Magic: The Gathering (8 marks)

A Magic: The Gathering! Sorcery card has the following properties (the card details used in this question are simplified from the real game):

- name: a string displaying the card name.
- cost: a string displaying casting cost.
- text: a string describing what the card does.



Two cards are considered equal if all their properties are the same.

A class MTGCard is given. A program that tests a few cards for their equality (TestMTG.java) is also given.

TestMTG can run now, but its use of "equality testing" may not be what we want.

- Write code in method **equals** of class MTGCard and use the method to do comparison.
- Modify TestMTG.java so that it gives correct result for each comparison (according to our definition of equal).

You must include both *MTGCard.java* and *TestMTG.java* in your submission.