COS10009/COS60006 Test 2 - 2020 S1

Assessment Details

- **Assessment**: Test 2 – Sem 1 2020 V1
- **Unit Codes**: COS10009/COS60006
- **Duration**: 2 hours in total. 90 minutes to complete, 30 minutes to submit.

Student Name: Georges Youssef

Student ID: 103070491

Instructions to Candidates

- **You must complete all the questions.**
- You must submit this test booklet to Doubtfire within 2 hours of the test becoming available.
- The test answers must be your own work. By submitting your answers to Doubtfire you are declaring the work as your own.

Section 1. Coding - Answer the following questions:

Q1.1) Write code using a **while loop** in the Ruby programming language that will **print out** the 5 times table. The output should look as follows:

```
1 x 5 = 5
2 x 5 = 10
3 x 5 = 15
4 x 5 = 20
5 x 5 = 25
6 x 5 = 30
7 x 5 = 35
8 x 5 = 40
9 x 5 = 45
10 x 5 = 50
11 x 5 = 55
12 x 5 = 60
```

Write your Ruby code here:

```
def main
        num = 5
        i = 1
        while (i<=12)
        mult=num*i
        puts "#{num} * #{i} = #{mult}"
        i+=1
end
```

Q1.2) Write a Ruby function that takes two **integer arguments**, and **returns** the **difference** between the value of both arguments.

Write your function here:

```
def diff(a, b)
        (a - b).abs
end
```

Q1.3) The following 'Code to Be Corrected' generates an error as indicated below. The expected output is also provided.

Indicate the incorrect line(s) of code and write replacement line(s) of code in the box provided under the code below. In your answer only write lines that need to be changed to correct the program.

Expected output:

```
Enter your age in years:
23
Enter your name:
Fred
Fred you were born in: 1997
```

Actual output:

```
Enter your age in years:
23
Enter your name:
Fred
debug.rb:23:in `print_year_born': undefined local variable or method `today' for main:Object (NameError)
        from debug.rb:31:in `main'
        from debug.rb:34:in `<main>'
```

Code to be corrected:

```ruby
require 'date'

# Asks the user to enter their age and returns an integer age
def get_age()
  puts "Enter your age in years: "
  age_in_years = gets
  return age_in_years.to_i
end

# takes a prompt and displays it to the user then returns the
# entered string
def get_string(prompt)
  puts prompt
  s = gets
  return s
end

# Calculate the year born based on the parameter age and print that out.
def print_year_born(name, age)
  year_born = today.year -  age
  puts name.chomp + " you were born in: " + year_born.to_s
end

def main
  age = get_age()
  name = get_string("Enter your name: ")
  print_year_born(name, age)
end

main
```

Write your corrected line(s) of code here (put the line number to indicate which line):

Line 20: year_born = Date.today.year - age

Q1.4) Arrays.

Complete the following code so that it prints out the array correctly. The output should look as follows:

```
Fido
Rex
Mandy
Bitza
Gypsy
Bluey
```

```
1
2   DOGS = ["Fido", "Rex", "Mandy", "Bitza", "Gypsy", "Bluey"]
3
4   # print out all the dogs in the array DOGS
5   def main()
6
7       index =  _____________
8
9       while (index <  _________________ )
10
11          puts( ____________________________ )
12
13          index += 1
14      end
15  end
16
17  main()
18
```

Write the missing sections for each line out here:

Line 7: 0

Line 9: 1

Line 11: DOGS

Section 2: Theory Questions

(select the boxes or indicate your answer by highlighting or an X)

Q2.1)

Is Sequence important in the following:

- Get out of bed
- Have a shower
- Get dressed
- Have breakfast

☒ **True**

☐ **False**

Q2.2) Write under each of the following sentences which of the terms listed best matched the sentence:

Type, Statement, Keyword, Evaluation, Assignment, Operator,

1. a syntactic unit of an imperative programming language that expresses some action to be carried out**.**

   Statement

2. a symbol that tells the compiler or interpreter to perform specific mathematical, relational or logical operation and produce a final result.

   Operator

3. determines the format in which data is stored.

   Type

Q2.3)

If you want the code in the body of the loop to execute at least once (i.e regardless of any conditional test), you use a post-test loop.

☐ **True**

☒ **False**

Q2.4)

Which of the following statements are true:

☒ In Ruby we can represent complex types using classes

☐ In Ruby we can represent complex types using structs

☐ In Ruby we can represent custom types using modules

☐ In Ruby complex types can be components of other complex types

☐ Once you define a type you can assign values to the fields

End of Task

Submit your completed answer sheet to Doubtfire Task 10T within 2 hours of the test starting.