Final exam

Started: Dec 8 at 7:14am

Quiz Instructions

Welcome to CS152 section 5 final exam for fall 2022 semester. The final is completely online to be taken any time during 12/8/22. Submit by the end of the day, 11:59pm California time. No late submissions will be accepted. Use any resources available to you (computer, lecture slides, lecture videos, quiz reviews, search engines). Good luck!

Question 1 — 1 pts

Functional programming is characterized by ...

- ○ Message passing between objects
- ◉ Function calls with local scope operations
- ○ Knowledge base for the rules and the facts in the program
- ○ Divide and conquer approach

Question 2 — 1 pts

Select all the hallmarks of functional programming

- ☐ Rules
- ☐ Objects
- ☐ List comprehension
- ☑ Referential transparancy
- ☑ Pure functions
- ☑ Lazy evaluation

- [ ] Predicates
- [x] First-class functions
- [ ] Functions as units of execution
- [x] Recursion
- [x] Higher-order functions

Question 3

1 pts

You have a function that performs a calculation and returns the result to the user. As a part of its computation, it also prints out some information to the user on the screen. Is this function pure?

- ( ) Yes
- (•) No

Question 4

1 pts

Can a pure function depend on a global variable?

- ( ) Yes
- (•) No

Question 5

1 pts

You wrote a function that will perform some computation and return the result as a return value. In your program you make a call to this function several times over the course of the application execution. Each time you use the same input parameter

values. The function executes and returns different values for each execution because the values of an outer scope variable it depends on changes over the course of the application execution. Is your function pure?

- ○ Yes
- ◉ No

Question 6 — 1 pts

Same scenario as in question 5 (previous question). Is your function referentially transparent?

- ○ Yes
- ◉ No

Question 7 — 1 pts

Impure functions are always an artifact of bad/erroneous programming practices.

- ○ True
- ◉ False

Question 8 — 1 pts

A base case in recursion is optional and will not be present in every recursive solution.

- ○ True

◉ False

Question 9 1 pts

Which is more efficient?

○ Non-tail recursion

◉ Tail recursion

Question 10 1 pts

The following is a Python code for printing Fibonacci series. This solutions utilized recursion. Identify if this is tail recursion or non-tail recursion.

```
def fibonacci_series(x):
    if x <= 1: # base condition
        return x
    else:
        return(fibonacci_series(x-1) + fibonacci_series(x-2))
```

◉ Tail recursion

○ Non-tail recursion

Question 11 1 pts

Is the following function referentially transparent?

```
def addTwo(a, b):
    return a + b
```

- (●) Yes
- ( ) No

Question 12 — 1 pts

Is the following function pure?

```
def computeAddition(a, b, c):
    result = a + b + c
    print("The result of the computation is: "+str(result)
    return result
```

- ( ) Yes
- (●) No

Question 13 — 1 pts

Which functions are treated as values?

- ( ) Pure functions
- ( ) Impure functions
- (●) First-class functions
- ( ) Higher-order functions
- ( ) Referentially transparent functions

Question 14 — 1 pts

In your code you are passing in a function as an input parameter into another function. What kind of function is the function you are passing in as an input parameter?

- ○ Referentially transparent function
- ○ Impure function
- ○ Pure function
- ○ Higher-order function
- ◉ First-class function

Question 15

1 pts

In your code you have a function, which returns another function as its return value. What kind of function is this function? (the question is about the function that returns, not the one being returned)

- ○ Pure function
- ○ First-class function
- ◉ Higher-order function
- ○ Referentially transparent function
- ○ Impure function

Question 16

1 pts

In your program you have the following code. Match the elements of your program with the correct terms.

```
def compute_result(value, func):
  return func(value)

def linear_equation(x):
  return 2*x+3

result = compute_result(4, linear_equation)

print(result)
```

compute_result	higher order function
linear_equation	first class function
result	variable
func	object referencing a function

Question 17

1 pts

In your program you have the following code. Match the elements of your program with the correct terms.

```
def square_result(f):
  def result(x):
    return f(x)*f(x)
  return result

plus_three = lambda x: x + 3
total = square_result(plus_three)
print(total(7))
```

square_result	higher order function
plus_three	first class function

total	closure
result	object that represents a func

Question 18

1 pts

Which package in Java provides functionality to implement higher order functions?

- (●) java.util.function
- ( ) java.util.closure
- ( ) java.util.declarative
- ( ) java.util.functional

Question 19

1 pts

What does a closure include? (select all that apply)

- [x] function definition
- [ ] input values for the function
- [x] lexical environment

Question 20

1 pts

What is the result the following Python code will produce?

```
myfunc = lambda x: list(range(x))
[ele for ele in reversed([myfunc(e) for e in range(5)])]
```

- ○ [[0, 1, 2, 3, 4], [0, 1, 2, 3], [0, 1, 2], [0, 1]]
- ○ [[], [0], [0, 1], [0, 1, 2], [0, 1, 2, 3]]
- ◉ [[0, 1, 2, 3], [0, 1, 2], [0, 1], [0], []]
- ○ [[], [0], [0, 1], [0, 1, 2], [0, 1, 2, 3], [0, 1, 2, 3, 4]]
- ○ [[0, 1, 2, 3, 4], [0, 1, 2, 3], [0, 1, 2], [0, 1], [0]]
- ○ [[0, 1, 2, 3, 4], [0, 1, 2, 3], [0, 1, 2], [0, 1], [0], []]

Question 21

1 pts

Closures provide performance benefits in interpreted programming.

- ◉ True
- ○ False

Question 22

1 pts

What are the lambda functions?

- ○ Inner functions that can be returned as objects from higher order functions
- ◉ Anonymous functions
- ○ Pure functions
- ○ Functions that do not return a return value
- ○ The only functions we can use with closures

Question 23

1 pts

Dataflow languages are conceptualized around what flowing through sequence of operations?

- ○ Closures
- ○ Variables
- ◉ Data
- ○ Other operations
- ○ Functions

Question 24

1 pts

How do dataflow programming languages perform computations?

- ◉ Using operation graphs
- ○ Using first-class functions
- ○ Using data units
- ○ Using SQL

Question 25

1 pts

Your program follows a reactive framework. In your program you have the following instructions.

b = 10

c = 5

a = b + c

d = a / 2

e = a + d

f = 3

g = e + f - 1

These lines of code already executed and later on in your program the value of **c** gets changed to **8**. What is the value of **g** now?

- (●) 34
- ( ) 32
- ( ) 29.5
- ( ) 28.5

Question 26 1 pts

Which system is responsible for managing everything about a database?

- ( ) SQL
- ( ) Structured data
- ( ) Persistant storage
- (●) DBMS
- ( ) Storage area

Question 27 1 pts

Can your JavaScript code access a database directly using an appropriate library?

- [x] No
- [ ] Yes

Question 28

1 pts

Which SQL command would you use to view records in your database table?

- [ ] PRINT
- [ ] INSERT
- [ ] VIEW
- [x] SELECT
- [ ] DISPLAY
- [ ] EXTRACT

Question 29

1 pts

Anonymous classes in Java are equivalent to anonymous functions in other programming languages.

- [ ] True
- [x] False

Question 30

1 pts

Every list comprehension can be rewritten with loops and every loop can be rewritten with list comprehension.

- ○ True
- ◉ False

Question 31 — 1 pts

Which code is syntactically correct?

- ○ lambda x, y, z: (x + y) / z
- ○ def lambda x, y, z: (x + y) / z
- ◉ lambda (x, y, z): (x + y) / z
- ○ lambda x, y, z: return (x + y) / z

Question 32 — 1 pts

You are given these two versions of a recursive solution to cumulative sum problem. Is one of them a tail recursive solution?

Code A:

```
function recsum(x) {
    if (x === 0) {
        return 0;
    } else {
        return x + recsum(x - 1);
    }
}
```

Code B:

```
function recsum(x, running_total = 0) {
    if (x === 0) {
        return running_total;
    } else {
        return recsum(x - 1, running_total + x);
    }
}
```

- ○ Both code A and B are tail recursive solutions
- ◉ Code B is tail recursive
- ○ Code A is tail recursive
- ○ Neither A nor B is a tail recursive solution

Question 33

3 pts

Utilize map() **Python** function to implement a mapping for a list of integers to produce a new list in which each element is the result of the following functions for each corresponding element in the original list:

$$f(x) = 4x^2 + 3x - 5$$

Example of your code execution:

original_list = range(10)

new_list = list(map(mapping of the original list to the function above))

print(new_list)

Edit View Insert Format Tools Table

12pt Paragraph B I U A T²

```
original_list = range(10)

new_list = list(map(lambda x: 4*x**2 + 3*x -5, original_list))

print(new_list)
```

p 20 words </>

Question 34

5 pts

Implement a function called **adder()** in **Python**. This function should be a higher order function that returns a function object, which takes in a single value and adds to it a value that the outer function accepts. Create a closure called **add_five** for the adder() function, with an input parameter value set to **5**. Create a different closure called **add_ten** for the adder() function, with an input parameter value set to **10**. Now create a function called **invoke_closure()**, which accepts two parameters: a list of integers, and a closure which to apply to each element of the list. Call this function twice, once using add_five and once using add_ten closure. Each time use the same list of integers generated by range(10). Your final two lines of code should look like this:

print(invoke_closure(add_five, list(range(10))))

print(invoke_closure(add_ten, list(range(10))))

The output of the above two lines of code should be the following:
[5, 6, 7, 8, 9, 10, 11, 12, 13, 14]
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19]

Edit View Insert Format Tools Table

12pt Paragraph B I U A T²

```
add_five = adder(5)
```

```
add_ten = adder(10)


def invoke_closure(function, list_):
        return list(map(function, list_))
print(invoke_closure(add_five, list(range(10))))
print(invoke_closure(add_ten, list(range(10))))
```

p 39 words </>

Question 35 1 pts

Is the following code valid in **JavaScript**?

let a = "Some text"

let a = 14

- ○ Yes
- ◉ No

Question 36 2 pts

Write a function called **pounds_to_killos()** in **JavaScript**, which accepts the value of pounds, converts that value to value in kilograms, and returns the result.

Edit View Insert Format Tools Table

12pt Paragraph B I U A T²

```
function pounds_to_kilos(p){
```

```
        return p * 0.453592
}
```

p

6 words

Question 37

5 pts

You are programming in **JavaScript**. In your program you have student objects with the following attributes:

- name (string)
- id (numeric)
- gpa (numeric)

Example: {name: "Jim Smith", id: 101, gpa: 3.7}

Write a function **called sort_students()**, which accepts an array of student objects and sorts that array based on the specified field, sorts the array of objects accordingly, and returns the sorted array. For example:

const students = [...];

students_sorted_by_name = called sort_students(students, "name")

students_sorted_by_gpa = called sort_students(students, "gpa")

students_sorted_by_id = called sort_students(students, "id")

Edit View Insert Format Tools Table

12pt Paragraph

```
function sort_student(students, field) {
```

```
function sort_student(students, field) {
    if(field === "name"){
        return students.sort(function(a, b){ return b.name - a.name});
    }else if(field === "gpa"){
        return students.sort(function(a, b){ return b.gpa - a.gpa});
    } else if(field === "id"){
        return students.sort(function(a, b){ return b.id - a.id});
    }
```

p 42 words </>

Question 38 5 pts

In **JavaScript**, define a class called **Person**. Each person has the following attributes:

- **name**
- **age**

The class **Person** should have the following functionality:

- **constructor** that initializes all the attributes
- method **print_info()** which prints out information about the person

Define a class called **Artist**, which inherits from class **Person**. Each artist should have the following attribute:

- specialty

The class Artist should have the following functionality:

- **constructor** that initializes all the attributes (and delegates to the parent as needed)
- method **artist_bio()**, which prints out all the information about the artist (name, age, specialty)

Add code that creates an array of artist instances (attribute values are up to you).

Iterate over the array of artists and invoke artist_bio() method on each instance.

```
            this.specialty=specialty;
        }
        artist_info = () => {
            return this.name + " " + this.age+" "+this.specialty ;
        }
}
const artists = [new Artist ("Test Artist 1", 21,"singing"), new Artist ("Test Artist 2", 29 , "singing" ), new Artist ("Test Artist 3", 22 , "dancing" ) ]
artists.forEach(artist => console.log(artist.artist_info()));
```

p 67 words </>

Question 39

4 pts

Implement **Scheme** code that applies mapping of the following function to each element of the specified list:

$$f(x) = 2x^3 - 5x + 1$$

Apply this mapping to the following list: **5, 6, 2, 9, 2, 8**

Apply **display()** function to print the results of the mapping operation.

Edit View Insert Format Tools Table

12pt Paragraph B I U A T²

```
(display (map (lambda(x)(+ (- (* 2 ( expt x 3)) (* x 5)) 1)) '(5 6 2 9 2 8) ))
```

p 18 words </>

Question 40

3 pts

In **Scheme** define a function called **compute_product()**, which accepts two input parameters and multiplies them by each other. **Print/display** the result of applying this function to values **3** and **5**.

Edit View Insert Format Tools Table

12pt Paragraph B I U A T²

```
(define (compute_product a b)
        (display (* a b))
)
(compute_product 3 5)
(newline)
```

p

11 words

Quiz saved at 8:17am

Submit Quiz