Q1 Multichoice Questions (1100 and 1130)

20 Points

Australian National University

COMP1100/COMP1130 Mid-Semester Quiz, Semester 1 2024

This file contains instructions for all COMP1100 and COMP1130 students on how to participate in the COMP1100/COMP1130 midsemester quiz, and also contains multiple choice questions for both COMP1100 and COMP1130 students to attempt. More detailed comments regarding the purpose of this quiz can be found in these lecture slides.

There are eight multiple choice questions in this document, worth 2.5 marks each, for a total of 20 marks. Incorrect or missing answers earn 0 marks, without further mark penalty. Each question is intended to have one best answer.

There is another document of multiple choice questions for COMP1130 students only to attempt, worth a total of 10 marks.

Marks for multiple choice questions will be released when the quiz is closed on Wednesday 27 March.

There are also six programming questions, worth 5 marks each, for a total of 30 marks, for both COMP1100 and COMP1130 students to attempt. **Haskell files can be downloaded by clicking this link**. These files contain complete information about the task required. Doctests are provided to help you write your code but are not necessarily comprehensive. You can submit these files to the separate assignments on the Gradebook dashboard. Your code will immediately be run against our tests, which will not match the provided doctests, and your mark out of 5 will be the

number of tests passed. You may resubmit as often as you wish.

For the purpose of self assessment you can total your marks from each assessment to get a mark out of 50 (for COMP1100 students) or 60 (for COMP1130 students).

Q1.1 Sets and Functions
2.5 Points

Suppose we wish to define a polymorphic mathematical function (not a Haskell program) f whose

- codomain is A, where A is any set, and
- domain is the set of functions $\mathbb{Z} \longrightarrow A$, where $\mathbb{Z}$ is the set of integers.

Which of the following supposed definitions for f are correct for this domain and codomain?

$f(x) = 0$

$f(x) = x(0)$

$(f(x))(y) = x$

$(f(x))(y) = y$

Save Answer

Q1.2 Basic Types
2.5 Points

Suppose we had a function with name and type

```
thud :: String -> Char -> Int
```

Which of the following is a valid use of `thud`?

- `thud 'a' 'b'`
- `thud 'a' "b"`
- `thud "a" 'b'`
- `thud "a" "b"`

Save Answer

Q1.3 Algebraic Datatypes
2.5 Points

Suppose that `Foo` is a finite type defined to have three different elements. How many elements are there in the following type:

```
data Qux = Bar Foo | Baz Foo Foo
```

- 2
- 9
- 12
- 18
- 27

Save Answer

Q1.4 Case expressions
2.5 Points

In which of these situations is it in general most advisable to use a guarded expression, rather than some other similar construction such as a `case`?

- Pattern matching on a `Bool`
- Pattern matching on an input
- Pattern matching on a number
- Pattern matching on a pair

Save Answer

Q1.5 Errors, Warnings, and Exceptions
2.5 Points

Suppose that you are programming with a `Maybe` type, and you accidentally write `just` instead of `Just`. Which of the following will happen?

- When you try to compile there will be an error
- When you compile there will be a warning
- The code will compile, but when you run it you will get an exception
- The code will compile, and when you run it this small typo will not have any bad effect

Save Answer

Q1.6 Lists
2.5 Points

Which of the following patterns, such as you might use to the left of an `->` in a `case` statement, will **not** match the list `[4,7]`?

- `_`
- `[x, y]`
- `x : y`
- `x : [y, z]`
- `x : y : z`

Save Answer

Q1.7 Recursion
2.5 Points

Consider the following code, with line numbers to the left:

```
1 pluto :: Int -> Bool
2 pluto x
3   | y > 0      = not (pluto (y - 1))
4   | otherwise = True
5   where y = abs x
```

Which line of code gives the **base case**?

- 1
- 2
- 3
- 4
- 5

Save Answer

Q1.8 Parametric Polymorphism
2.5 Points

There is a polymorphic data type constructor in Haskell's Prelude library called `Either`. Its definition is

```
data Either a b = Left a | Right b
```

Which of the following is a valid element of an instantiation of an `Either` type?

- `Left 'a' 7 :: Either Char Int`
- `Left 4 :: Either Int Double`
- `Left False :: Either Integer Bool`
- `Left "hello" "world" :: Either String String`

Save Answer

Save All Answers

Submit & View Submission ›