

Functional Programming in Haskell

Mid-semester Examination, Semester I, 2023–2024

Date : October 4, 2023

Time : 0930–1130

Marks : 100

Weightage : 30%

This paper has two parts. Each Part A question is worth 5 marks. For Part A, provide short answers. For Part B, provide crisp and short programs. Syntax should more or less correspond to Haskell, but minor inaccuracies like wrong indentation will be ignored. You can assume standard library functions, but if you use something not taught in class, you must explain briefly what it does.

Part A

1. What is the result of the following expression?

```
filter (≥ 5) (reverse [(-10)..10] ++ [(-10)..10])
```

2. What is the result of the following expression?

```
(length . takeWhile (≥ -6)) (reverse [0..10] ++ [(-10)..10])
```

3. Suppose $(++)$ is defined as follows:

```
[] ++ ys = ys  
(x:xs) ++ ys = x: xs ++ ys
```

How many times is the second line of the definition invoked in the computation of the following expression?

```
foldr (++) [] [[0,1], [2..5], [6..13]]
```

4. How many times is the second line of the definition (of $(++)$) invoked in the computation of the following expression?

```
foldl (++) [] [[0,1], [2..5], [6..13]]
```

5. What is the position of $(4, 1)$ in the following list (counting positions from 0)?

```
[(j,i) | i<-[0..9], j<-[(i+1)..9]]
```