# **Name: Abdurrahman Qureshi**

# **Roll No: 242466**

Practical No: 8

1) Demonstrate the use of basic list operations.

**CODE:**

main :: *IO*()

main = do

    putStrLn "Increment each element by 1:"

    print (map (+1) [1, 2, 3, 4, 5])

    putStrLn "Filter elements greater than 3:"

    print (filter (>3) [1, 2, 3, 4, 5])

    putStrLn "Filter even numbers from the list:"

    print (filter even [1, 2, 3, 4, 5])

    putStrLn "Filter odd numbers from the list:"

    print (filter odd [1, 2, 3, 4, 5])

    putStrLn "Get the length of the list:"

    print (length [1, 2, 3, 4, 5])

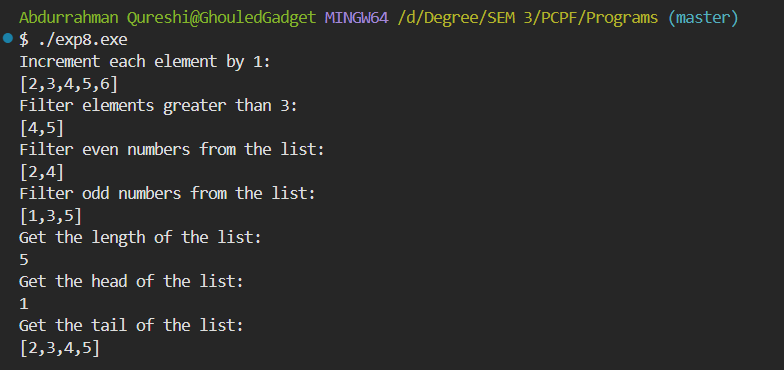
    putStrLn "Get the head of the list:"

    print (head [1, 2, 3, 4, 5])

    putStrLn "Get the tail of the list:"

    print (tail [1, 2, 3, 4, 5])

**OUTPUT:**



2) Demonstrate a program for Pattern Matching

**CODE:**

main :: *IO*()

main = do

    let numbers = [1, 2, 3, 4, 5]

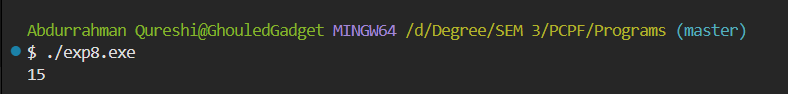
    print (sumList numbers)

sumList :: [*Int*] -> *Int*

sumList [] = 0

sumList (x:xs) = x + sumList xs

**OUTPUT:**



2) Demonstrate a program for List Comprehension

**CODE:**

main :: *IO*()

main = do

    putStrLn "Double elements greater than 2:"

    print ([x \* 2 | x <- [1..5], x > 2])

**OUTPUT:**

