# КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені ТАРАСА ШЕВЧЕНКА



## ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

## **Кафедра прикладних інформаційних систем**

**Звіт до лабораторної роботи №3**

# з курсу **«Функціональне програмування»**

*Студента 4 курсу*

*Групи ПП-41*

*спеціальності 122*

*«Комп'ютерні науки»*

*ОП«Прикладне програмування»*

*Селецького Віктора Романовича*

## Київ 2023

# Тема

# Замикання (локальні визначення)

# Мета

# Oзнайомитися з особливостями механізму функціонального програмування – замикання. Розширити навички використання анонімних функцій. Поглибити знання в області написання нетривіальних функцій вищого порядку.

# Завдання

# 

# Виконання

Пишемо код:

# 

module Main (main) where

import Data.IORef

updateCounter reference = do

let increment :: IO()

increment = do

modifyIORef' reference (+1)

increment

return reference

main :: IO ()

main = do

ref <- newIORef(0::Int)

counter <- updateCounter ref

counter <- updateCounter ref

counter <- updateCounter ref

counter <- updateCounter ref

counter <- updateCounter ref

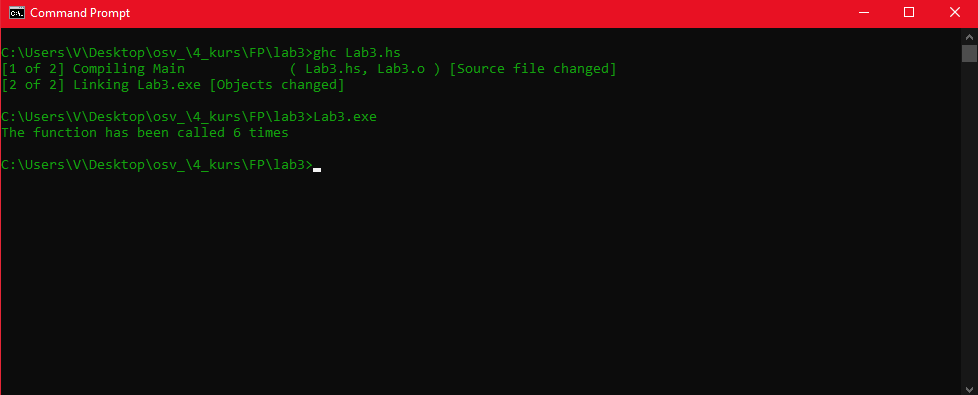
counter <- updateCounter ref

count <- readIORef counter

putStrLn $ "The function has been called " ++ show count ++ " times"

# 

Компілюємо та виконуємо:



**Висновок**

# Oзнайомився з особливістю механізму функціонального програмування – замикання, розширив навички використання анонімних функцій, поглибив знання в області написання нетривіальних функцій вищого порядку.