

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені
ТАРАСА ШЕВЧЕНКА**



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра прикладних інформаційних систем

Звіт до лабораторної роботи №7

з курсу

«Функціональне програмування»

*Студентки 4 курсу
групи ПП-41*

спеціальності 122 «Комп'ютерні науки»

ОП «Прикладне програмування»

Штось Софії Максимівни

*Викладач:
Пирог М. В.*

Київ – 2023

Тема роботи: Перетворення графів при виконанні програм.

Мета роботи: Ознайомитися з процесом перетворення графів при виконанні функціональних програм. Розширити навички представлення функціональних програм. Поглибити знання SECD-машини та графічного представлення функціональних програм.

Теоретичні відомості

Тема №7 «Представлення, інтерпретація та виконання функціональних програм. SECD-машина» має на меті розкриття основних аспектів представлення та виконання функціональних програм на базі мови Haskell. Для цього описуються особливості представлення функціональних програм, розглядається Eval/apply-інтерпретатор Маккарті, архітектура SECD-машини та особливості компіляції функціональних програм у ній. На самостійне опрацювання виносяться питання перетворення графів при виконанні програм, що відноситься до розділу графічного представлення функціональних програм.

Для поглиблення знань в області представлення функціональних програм, а саме в графічному представленні, слід особливу увагу, перш за все звернути на графічне представлення конструкцій розширеного лямбда-числення взагалі. Необхідно розглянути особливості процесу перетворення графів при виконанні функціональних програм. Для вірної побудови графу слід з'ясувати правила виконання редукцій при побудові графа а також розглянути функції-проектори та фіктивні вузли..

Завдання для виконання

Представити Y-комбінатор Каррі в графічному вигляді. Виконати редукції графа.

Хід роботи

Код програми (Haskell):

```
module Main (main) where

fix :: (a -> a) -> a
fix f = f (fix f)

fac' :: (Integer -> Integer) -> Integer -> Integer
```

```

fac' f n = if n == 0 then 1 else n * f (n-1)

fac :: Integer -> Integer
fac = fix fac'

main :: IO ()
main = do
    putStrLn "Обчислення факторіалу числа 5 за допомогою Y-комбінатора."
    putStrLn "Результат:"
    print (fac 5)

```

Результат роботи програми:

The screenshot shows a code editor with the following Haskell code in `Main.hs`:

```

1  module Main (main) where
2
3  fix :: (a -> a) -> a
4  fix f = f (fix f)
5
6  fac' :: (Integer -> Integer) -> Integer -> Integer
7  fac' f n = if n == 0 then 1 else n * f (n-1)
8
9  fac :: Integer -> Integer
10 fac = fix fac'
11
12 main :: IO ()
13 main = do
14     putStrLn "Обчислення факторіалу числа 5 за допомогою Y-комбінатора."
15     putStrLn "Результат:"
16     print (fac 5)

```

Below the code, the `TERMINAL` tab shows the execution output:

```

> v TERMINAL
Registering library for Lab7-0.1.0.0..
Обчислення факторіалу числа 5 за допомогою Y-комбінатора.
Результат:
120
(base) sofiiashon@Sofiias-MacBook-Air Lab7 %

```

Висновок: Отже, у ході цієї лабораторної роботи було проведено ознайомлення з процесом перетворення графів при виконанні функціональних програм. Було розширено навички представлення функціональних програм.

Контрольні запитання

1. Чим відрізняється модель виконання функціональних програм у вигляді обчислень на SECD-машині від моделі графічного представлення?

Модель виконання функціональних програм на SECD-машині відрізняється від моделі графічного представлення тим, що SECD-машина є абстрактною машиною для

виконання функціональних програм, використовуючи стек, середовище, континуум та динамічну пам'ять, тоді як графічне представлення базується на зображенні виразів як графів, де вузли представляють вирази, а ребра відносини між ними.

2. Яким чином лямбда-вираз може бути представлений у вигляді графа?

Лямбда-вираз може бути представлений у вигляді графа, де вузли відповідають абстрактним синтаксичним об'єктам, а ребра вказують на залежності між ними, відображаючи структуру виразу та його зв'язки.

3. В чому сутність правил виконання редукцій при побудові графа та яким чином проходить процес перетворення графу виразу?

Сутність правил виконання редукцій при побудові графа полягає в заміні частин виразу за допомогою еквівалентних частин, спрощуючи та оптимізуючи граф. Процес перетворення графу виразу включає в себе послідовність редукцій, що зводять граф до найпростішої форми.

4. Яку необхідність зумовлює вирішення проблеми поділу виразів в представленні виразів у вигляді графа?

Проблема поділу виразів у представленні графа виникає з необхідності ефективного управління обчисленням та забезпеченням правильного порядку виконання підвиразів. Це стає важливим при обробці великих та складних виразів.

5. Що являють собою функції-проектори та навіщо утворюються фіктивні вузли?

Функції-проектори в графічному представленні виразів є вузлами, які відображають абстрактні функції та їх аргументи. Фіктивні вузли утворюються для забезпечення правильної структури графа та позначення обчислень, що ще не завершилися.