Затверджено

Наказ Міністерства освіти і науки,

молоді та спорту України

29 березня 2012 р. №384

Форма№Н-5.05

Київський національний університет ім. Тараса Шевченка

|  |  |  |
| --- | --- | --- |
| **ОКР** |  | бакалавр |
| **Спеціальність** |  | 121 Інженерія програмного забезпечення |
| **Семестр** |  | сьомий |
| **Навчальна дисципліна** |  | Функціональне програмування |

# ЕКЗАМЕНАЦІЙНИЙ БІЛЕТ № 1

1. **Теоретичне запитання (5 балів).**

Поняття, характеристика, особливості парадигм декларативного та імперативного програмування, різниця між цими парадигмами. Приклади програм, що написані мовою імперативного та функціонального програмування

1. **Написати програму мовою функціонального програмування (20 балів)**

Розв’язати нелінійне рівняння *x* = cos(*x*) будь-якими чисельними методами (бісекції, перебору, хорд, дотичних) визначивши інтервал [*a, b*], на якому існує рішення рівняння. Значення *a, b* інтервалу вибрати самостійно. Порівняти результати розв’язків двома методами.

1. **Виконати Тест1 (Scheme+Lisp) або Test2 (Haskell+Scheme) на вибір студентв, що подані в Google Forms за посиланням (10 балів):**

**Тест1 (Scheme+Lisp):**

<https://docs.google.com/forms/d/e/1FAIpQLSfK_xEI9FajWuIO6Hp24aGDxBUIJQ0Mr37jUVhJhMCozZO6CQ/viewform>

**Test2 (Haskell+Scheme):**

<https://docs.google.com/forms/d/e/1FAIpQLSfCxpOpST5ugw2bt5shUEz-YUY5MCZjq3EQMKDzxUV4uFFYHw/viewform>

1. **Пояснити семантику поданого нижче коду (на вибір студента код мовою SCHEME або HASKELL), відредагувати код у випадку необхідності, скомпілювати та виконати його. Проінтерпретувати отримані результати (5 балів).**

|  |  |
| --- | --- |
| SCHEME | HASKELL |
| (define (follow int directions)  (if (pair? directions)  (if (pair? int)  (case (car directions)  ((left) (follow (car int) (cdr directions)))  ((straight) (follow (cadr int) (cdr directions)))  ((rigth) (follow (caddr int) (cdr directions))))  #f)  int))  (follow '(a b c) '(rigth)) ; right turn goes to c  (follow '(a (b c (d e f)) ; straight to (b c (d e f)) ;right to (d e f)  g) '(straight rigth left)) ; left to d | data Expr = Num Int | Add Expr Expr | Mul Expr Expr  deriving (Eq,Show)  data Ex = NumEx Int | BinEx Op Ex Ex  data Op = AddOp | MulOp  convert (Num n) = NumEx n  convert (Add e1 e2) = BinEx AddOp (convert e1) (convert e2)  convert (Mul e1 e2) = BinEx MulOp (convert e1) (convert e2) |

Затверджено на засіданні кафедри програмних систем і технологій, протокол №2 від «11» вересня 2021р.

Завідувач кафедри Бичков О.С.

Екзаменатор Ковалюк Т.В.