

Đề kiểm tra giữa kỳ

Môn kiểm tra: Nguyên Lý Ngôn Ngữ Lập Trình
Thời gian: 65 phút

Ngày kiểm tra: 06-10-2018

☐ Sinh viên được phép sử dụng tài liệu

☐ Sinh viên không được sử dụng tài liệu

Mã đề: 1182

- Sinh viên phải ghi tên và mã số sinh viên trên đề kiểm tra và giấy làm bài. Khi nộp bài, sinh viên phải nộp cả đề kiểm tra và giấy làm bài. Nếu không nộp đề kiểm tra, sinh viên sẽ bị điểm 0 cho toàn bộ bài kiểm tra.
- Sinh viên làm tất cả câu hỏi trên giấy làm bài.

I. Phần câu hỏi lập trình (cùng dùng để tính điểm BTL 1 theo công thức trung bình điều hoà): (4 điểm)

- Viết các mô tả thích hợp (token và biểu thức chính quy) trên ANTLR để mô tả token INTLIT, biểu diễn một hằng nguyên thập phân trên ngôn ngữ C. INTLIT là một chuỗi các ký số từ 0 đến 9 nhưng không bắt đầu bằng ký số 0 trừ trường hợp hằng 0 (chỉ ghi một số 0). Ví dụ 1023, 900, 0 là các chuỗi hợp lệ ứng với token INTLIT trong khi các chuỗi sau không hợp lệ: 00 (không bắt đầu là số 0 nên chuỗi này sẽ tạo ra 2 token INTLIT(0)), 0123 (không bắt đầu là số 0 nên chuỗi này sẽ tạo 2 token INTLIT(0) và INTLIT(123)), 123a4 (không có ký tự a nên sẽ tạo ra token INTLIT(123) và báo lỗi ở a). Sinh viên chỉ cần mô tả cho token INTLIT, không cần xử lý lỗi.
- Viết các mô tả thích hợp (token và biểu thức chính quy) trên ANTLR để mô tả token STRING, biểu diễn cho một hằng chuỗi trên ngôn ngữ X. STRING là một chuỗi bắt đầu bằng 2 dấu ", kết thúc cũng bằng 2 dấu ", và ở giữa là một chuỗi (có thể rỗng) các ký tự bất kỳ nhưng không được phép có 2 dấu " liên tiếp hoặc 1 dấu " nằm ở cuối. Ví dụ các chuỗi hợp lệ: "" abc " de" f "" , "" ?!" abc " mdm" dfg "" . Các chuỗi sai "" abc "" (chỉ có "" abc "" là tạo thành STRING), "" abc "" (chỉ có "" abc "" là tạo thành STRING).
- Cho biểu thức có các phép toán nhị phân @, ^, và ? trong đó phép @ có độ ưu tiên cao nhất và có tính kết hợp phải, phép toán ^ có độ ưu tiên trung bình và không có tính kết hợp, phép toán ? có độ ưu tiên thấp nhất và có tính kết hợp trái. Toán hạng nguyên tố là số nguyên (INT) và dấu () có thể được dùng để thay đổi độ ưu tiên và tính kết hợp. Hãy viết văn phạm cho biểu thức dùng dạng BNF?
- Một kiểu trên ngôn ngữ TEST có thể là int, float hoặc kiểu array. Một kiểu array trên TEST bắt đầu bằng từ khoá array theo sau là một danh sách miền, kế đến là từ khoá of và cuối cùng là một kiểu. Một danh sách miền bắt đầu bằng dấu '[', kết thúc bằng một dấu ']' và ở giữa là các miền cách nhau bằng dấu ','. Phải có ít nhất một miền trong danh sách miền. Mỗi miền được viết bằng 2 số nguyên cách nhau bởi hai dấu chấm liên tiếp '..' Một số nguyên có thể là số dương (chỉ gồm các ký số) hoặc số âm (bắt đầu bằng dấu '-' theo sau là các ký số).

Ví dụ:



`array [1 .. 3, -2 .. 4, 15 .. 20] of array [-13 .. 15] of int`

biểu diễn một array có 3 miền con với kiểu thành phần là một array có 1 miền với kiểu thành phần là int.

Hãy viết văn phạm và các token cần thiết cho một kiểu array trên TEST dùng ANTLR?

II. Phần câu hỏi lý thuyết: (6 điểm)

5. (a) Cho các lớp A, B (lớp con của A), C và Visitor được khai báo như sau:

```
class A:
    def accept(self, v):
        v.visitA(self)
class B(A):
    def accept(self, v):
        v.visitB(self)
class C:
    def accept(self, v):
        v.visitC(self)

class Visitor:
    def visitA(self, x):
        return x.accept(self)
    def visitB(self, x):
        pass
    def visitC(self, x):
        pass
```

Hỏi lệnh gọi `x.accept(self)` trong phương thức `visitA` của class `Visitor` có thể gọi (gián tiếp) đến các phương thức nào của class `Visitor`? **Giải thích**

- (b) Hãy giải thích giải pháp giải quyết "vấn đề kim cương" (diamond problem) trên Python 3 và minh họa cho giải thích trên thông qua xác định chuỗi tìm kiếm lớp cha của đoạn mã Python.

```
1 class A:
2     def foo(self):
3         print("A")
4 class B(A):
5     pass
6 class C(B):
7     pass
8 class D(A):
9     def foo(self):
10        print("D")

11 class E(A):
12     pass
13 class F(D, E):
14     def foo(self):
15         print("F")
16 class G(C, F, D):
17     pass
18 x = G()
19 x.foo() #1
```

6. (a) Thế nào là closure trên lập trình hàm? Cho một ví dụ để giải thích?

- (b) Viết hàm `subsum` nhận vào một danh sách `lst` và trả về một danh sách mà mỗi phần tử là tổng của phần tử tương ứng trong `lst` với tất cả các phần tử đứng trước nó trong `lst`. Ví dụ `subsum([1,3,4,5])` sẽ trả về `[1,4,8,13]` trong đó $1 = 1$, $4 = 1 + 3$, $8 = 1 + 3 + 4$, và $13 = 1 + 3 + 4 + 5$. Yêu cầu dùng một trong hai phương pháp: dùng quy hoặc dùng hàm bậc cao?

HẾT

