**BTL01**

(Chiếm 30% điểm học phần)

Cho ngôn ngữ lập trình UPL với các đặc điểm như sau:

- Bắt đầu và kết thúc chương trình bằng cặp từ khóa **begin** và **end**

- Có hai kiểu dữ liệu là số nguyên (int) và boolean (bool) →>> Ko can quan tam o Parser

- id: bắt đầu bằng ký tự chữ, nếu có ký tự số thì phải ở cuối (có thể có nhiều) →> Scanner

[a-z]+ [0-9]\* aaaaaa000000

- Biến phải được khai báo trước khi sử dụng. Biến có thể được khởi tạo cùng với câu lệnh khai báo (sử dụng phép gán). → Semantic Analyzer

- Có 3 phép toán so sánh: lớn hơn (>), lớn hơn hoặc bằng (>=), bằng (==)

- Có phát biểu lựa chọn if then và if then else

- Có phát biểu lặp do ...while

- Có phát biểu in ra màn hình (print) với tham số là một biểu thức

- Các phép toán cộng (+) và nhân (\*) cho số nguyên. **\* được ưu tiên hơn** +. Lập trình viên có thể sử dụng **dấu ngoặc ()** để xác định lại thứ tự tính toán

- Có cơ chế comment như của Java (nghĩa là có cả /\*...\*/ và //…)

Ví dụ của một chương trình viết bằng UPL:

begin

int x;

int y=x+1;

/\* comments

cho nhiều dòng

\*/

bool a;//comment cho một dòng /

if x>a then{

int c=1;

}else{

y=x;

x=x+1;

}

print(a);

if x>=a then {

x=x+1;

}

bool x=a==b;

do{

int b=1;

b=b\*10;

a=(b+10)\*b;

}while(a>1);

print(a+1);

end

1) Đề xuất văn phạm phi ngữ cảnh (3 điểm)

2) Xây dựng bộ phân tích từ vựng (tối đa 5 điểm nếu sử dụng công cụ như Flex/JFlex, tối đa 6 điểm nếu viết từ đầu, tối đa 7 điểm nếu có cả hai phiên bản; nên có khả năng báo lỗi một cách chi tiết)

Nộp:

* Mã nguồn
* Báo cáo với các nội dung chính là thông tin thành viên của nhóm, những điểm chính trong việc thiết kế văn phạm và bộ phân tích từ vựng, cách chạy và kết quả của bộ phân tích từ vựng.