

BÁO CÁO BÀI TẬP LỚN MÔN LẬP TRÌNH MẠNG

HỌC KỲ I NĂM HỌC 2018 - 2019

Nhóm: 01 _ STT: 15

Thành viên:

- | | |
|------------------------------|--------------------------|
| 1. Họ và tên: Lê Tuấn Hiệp | Mã sinh viên: B15DCCN209 |
| 2. Họ và tên: Đỗ Viết Duy | Mã sinh viên: B15DCCN164 |
| 3. Họ và tên: Nguyễn Nam Anh | Mã sinh viên: B15DCCN021 |

Nội dung

1. Giới thiệu sơ lược chủ đề
 - a. Mục tiêu:
 - Công cụ hỗ trợ chấm tự động mã nguồn Java.
 - Công cụ cho phép gửi/nhận mã nguồn java và đánh giá mã nguồn:
 - + Hoạt động đúng input/output (đọc từ file txt)
 - + Thiết kế đúng yêu cầu (đọc từ file xml)
 - b. Kết quả đã đạt được
 - Chương trình hoạt động theo mô hình client/server thiết kế đa luồng.
 - Hỗ trợ kiểm tra thiết kế file Java theo yêu cầu (file mẫu có sẵn)
 - Hỗ trợ kiểm tra kết quả (tự động điền input và check output của file gửi lên so với output cho trước.)
 - c. Hạn chế, hướng phát triển
 - Hạn chế:
 - o Không lưu trữ lại bài của Client gửi lên.
 - o Thời gian chưa được tối ưu.
 - Hướng phát triển:

- Tính thời gian thực hiện của file Client gửi lên.
- Hỗ trợ Client gửi package.
- Tích hợp Git.
- Lưu trữ thời gian, bài làm của Client.
- Bảng xếp hạng điểm số hiện tại.

2. Phân công công việc

STT	Họ tên	Các nội dung thực hiện	Đánh giá
1.	Lê Tuấn Hiệp	Module gửi nhận mã nguồn theo mô hình client/server theo thiết kế đa luồng và input output.	
2.	Đỗ Viết Duy	Module phân tích cấu trúc mã nguồn Java và xuất ra cây cấu trúc xml và chấm kết quả.	
3.	Nguyễn Nam Anh	Module so sánh mã nguồn Java và chấm thiết kế.	

3. Quá trình phát triển

STT	Phiên bản	Vấn đề	Xử lý	Tự đánh giá	Link tải
1.	1.0 12/09	1. Chức năng chấm thiết kế chỉ hiển thị đúng, sai không hiển thị được cụ thể lỗi nào 2. Chức năng phân tích cấu trúc mã nguồn Java thiếu một vài tham số. 3. Chưa có mô hình gửi nhận mã nguồn.	1. Sử dụng DocumentBuilder để đọc XML sau đó chấm thiết kế 2. Sử dụng lớp Parameter để lấy ra tên của kiểu tham số. 3. Sử dụng TCP Socket để truyền dữ liệu. - Đọc file bằng FileInputStream. - Gửi nhận file bằng DataOutputStream, DataInputStream.	- OK	https://github.com/hieplt1018/PhanTichCodeJava/blob/master/BaiTapLon_v1.0.rar
2.	2.0 14/10	1. Sinh ra file .class từ file .java 2. Gửi nhận mã nguồn chưa hỗ trợ đa luồng. 3. Chức năng chấm kết quả chưa hoàn thiện.	1. Dùng ProcessBuilder để chạy lệnh javac ClassName.java 2. Xử lý đa luồng:	- OK	https://github.com/hieplt1018/PhanTichCodeJava/blob/master/ver2.0.rar

		4.Chức năng chấm thiết kế hiển thị được chi tiết lỗi	<ul style="list-style-type: none"> - Cài đặt lớp Server sử dụng <i>ExecutorService</i> xây dựng <i>Thread Pool</i>. - Cài đặt lớp <i>WorkerThread</i>. 3. Dùng <i>ProcessBuider</i> để chạy lệnh <code>java fullClassName <input.txt</code> + Sử dụng <i>BufferedReader</i> để đọc kết quả từ output của cmd và file <code>result.txt</code> + So sánh 2 chuỗi 4. Sử dụng <i>DocumentBuilder</i> để đọc XML sau đó chấm thiết kế 		
3.	3.0 8/11	<ul style="list-style-type: none"> 1. Chương trình gửi file phía client chưa có giao diện. 2. Chấm file class đã nộp lên của Client đến trước thay vì Client vừa gửi. 3. Client gửi nhầm tên bài. 	<ul style="list-style-type: none"> 1.Sử dụng <i>JavaSwing</i> tạo giao diện phía Client. 2. Cài đặt hàm <code>deleteFile()</code> sử dụng method <code>delete</code> của <i>File</i> để xóa file của Client đến trước tránh sự nhầm lẫn. 3. Bắt lỗi cho hàm <code>compileToClassFile()</code> 	- OK	https://github.com/hieplt1018/PhanTichCodeJava/blob/master/ver3.0.rar