

## CHƯƠNG TRÌNH HỌC PHẦN

### 1. Thông tin chung về học phần

Tên học phần: Lập trình ứng dụng với Java (Applied Java Programming)

Mã học phần:

Số tín chỉ: 3

Học phần tiên quyết: Lập trình hướng đối tượng

Đào tạo trình độ: Đại học

Giảng dạy cho các ngành: Công nghệ Thông tin

Bộ môn quản lý: Mạng và Truyền thông

Phân bổ thời gian trong học phần:

- Nghe giảng lý thuyết: 22
- Làm bài tập trên lớp: 4
- Thảo luận: 4
- Thực hành, thực tập: 15 (x2)
- Tự nghiên cứu: 90

### 2. Mô tả tóm tắt học phần

Học phần trang bị cho người học kiến thức cơ bản về lập trình bằng ngôn ngữ Java: những khái niệm cơ sở, các lệnh điều khiển, cách xây dựng, tổ chức lớp các đối tượng trong các chương trình ứng dụng viết bằng ngôn ngữ Java. Người học sẽ được trang bị cách tiếp cận lập trình hướng đối tượng và cách thức thể hiện nó với Java. Cơ chế xử lý ngoại lệ, lập trình giao diện người dùng đồ họa (GUI) và cơ bản về lập trình client/server cũng sẽ được cung cấp trong học phần này. Kết thúc môn học sinh viên sẽ có khả năng cài đặt các chương trình ứng dụng độc lập/standalone hoặc ứng dụng client/server trên môi trường Internet.

### 3. Chủ đề và chuẩn đầu ra của học phần

#### 3.1. Danh mục chủ đề của học phần

1. Lập trình Java cơ bản
2. Lập trình hướng đối tượng với Java
3. Lưu trữ và xử lý danh sách các đối tượng
4. Lập trình nhập xuất
5. Xử lý ngoại lệ
6. Xây dựng ứng dụng với giao diện đồ họa
7. Lập trình socket

#### 3.2. Chuẩn đầu ra của quá trình dạy - học từng chủ đề của học phần

##### Chủ đề 1: Lập trình Java cơ bản

Nội dung	Mức độ
<b>Kiến thức</b>	
1. Đặc trưng của ngôn ngữ Java, lịch sử hình thành và phát triển, các loại ứng dụng Java	1
2. Công cụ và môi trường lập trình Java	2
3. Cấu trúc chương trình Java	2
4. Các kiểu dữ liệu, hằng và biến, toán tử và biểu thức trong Java	3
5. Các câu lệnh đơn giản, các cấu trúc điều khiển	3

6. Câu lệnh nhập, xuất dữ liệu đơn giản	2
<b>Thái độ</b> 1. Nắm vững các thành phần cơ bản trong một chương trình Java giúp lập trình viên có thể dễ dàng đọc hiểu cũng như thiết kế và cài đặt các chương trình Java từ cơ bản đến phức tạp 2. Sử dụng thành thục các công cụ phát triển mạnh mẽ như Netbeans, Eclipse sẽ làm tăng tốc độ viết, kiểm lỗi, triển khai các chương trình Java	
<b>Kỹ năng</b> 1. Cài đặt và sử dụng môi trường và các công cụ phát triển thông dụng (JDK, Jcreator, Netbeans, Eclipse,...) cho ngôn ngữ java. 2. Viết, biên dịch và chạy chương trình Java 3. Thiết kế và cài đặt các chương trình với chức năng nhập xuất đơn giản 4. Thiết kế và cài đặt chương trình có sử dụng các kiểu dữ liệu sơ cấp hay tham chiếu 5. Thiết kế và cài đặt các chương trình có sử dụng các cấu trúc điều khiển	2 3 2 3 3

### Chủ đề 2: Lập trình hướng đối tượng với Java.

Nội dung	Mức độ
<b>Kiến thức</b> 1. Mô hình hóa hướng đối tượng với ngôn ngữ UML 2. Tính đóng gói; gói và giao diện trong Java 3. Lớp (Class) và đối tượng (Object); phương thức (Method); cách thức xây dựng các lớp trong Java 4. Tính thừa kế; tính đa hình 5. Các gói và lớp thông dụng trong Java	2 2 3 2 2
<b>Thái độ</b> 1. Lập trình hướng đối tượng là một phương pháp mạnh mẽ, hiệu quả và phổ biến để xây dựng nên những chương trình ứng dụng lớn trên máy tính ngày nay 2. Java là ngôn ngữ lập trình thuần hướng đối tượng. Cần nắm vững các khái niệm, kỹ thuật hướng đối tượng trong Java để có thể xây dựng nhanh chóng các ứng dụng theo phương pháp hướng đối tượng	
<b>Kỹ năng</b> 1. Mô tả lớp và đối tượng bằng ngôn ngữ UML 2. Thiết kế và cài đặt các lớp trong Java theo đúng nguyên tắc hướng đối tượng 3. Thiết kế và cài đặt các chương trình theo nguyên tắc hướng đối tượng	2 3 2

### Chủ đề 3: Lưu trữ và xử lý danh sách các đối tượng.

Nội dung	Mức độ
<b>Kiến thức</b> 1. Mảng và lớp ArrayList 2. Lớp Vector và giao diện Enumeration	2 2

3. Danh sách và giao diện Lists	2
<b>Thái độ</b> <ol style="list-style-type: none"> <li>Mảng là cấu trúc dùng để lưu trữ, tổ chức và xử lý dữ liệu thông dụng trong các chương trình.</li> <li>Java cung cấp nhiều lớp hỗ trợ cho việc tổ chức lưu trữ và xử lý một tập các đối tượng một cách mạnh mẽ, linh hoạt. Việc vận dụng thành thực các lớp tiện ích này sẽ giúp xây dựng các chương trình một cách dễ dàng, nhanh chóng mà không cần cài đặt lại các cấu trúc dữ liệu cũng như các giải thuật xử lý dữ liệu (sắp xếp, tìm kiếm) thông dụng.</li> </ol>	
<b>Kỹ năng</b> <ol style="list-style-type: none"> <li>Sử dụng các cấu trúc mảng để lưu trữ dữ liệu trong chương trình</li> <li>Thiết kế và cài đặt các ứng dụng có sử dụng mảng và các thao tác trên mảng (sắp xếp, tìm kiếm)</li> <li>Thiết kế và cài đặt các ứng dụng có sử dụng danh sách để lưu trữ dữ liệu, đối tượng</li> </ol>	<p>3</p> <p>2</p> <p>2</p>

#### Chủ đề 4: Lập trình nhập xuất.

Nội dung	Mức độ
<b>Kiến thức</b> <ol style="list-style-type: none"> <li>Luồng nhập xuất cơ sở hướng Byte (InputStream, OutputStream và các lớp dẫn xuất từ hai lớp này)</li> <li>Luồng nhập xuất chuẩn trong gói java.lang.System</li> <li>Luồng lọc (Filter stream)</li> <li>Luồng nhập xuất hướng ký tự</li> <li>Lớp File, RandomAccessFile</li> <li>Các lớp hỗ trợ vào ra thông dụng trong gói java.io, java.util</li> </ol>	<p>3</p> <p>2</p> <p>2</p> <p>2</p> <p>2</p> <p>2</p>
<b>Thái độ</b> <ol style="list-style-type: none"> <li>Vào ra là các tác vụ phổ biến của chương trình, đặc biệt là trong các ứng dụng mạng</li> <li>Java cung cấp nhiều lớp hỗ trợ thực hiện tác vụ vào ra với các nguồn dữ liệu khác nhau một cách linh hoạt. Việc lựa chọn và sử dụng luồng vào ra thích hợp sẽ giúp tăng hiệu năng chương trình</li> </ol>	
<b>Kỹ năng</b> <ol style="list-style-type: none"> <li>Cài đặt các chương trình thực hiện tác vụ vào ra với các luồng nhập xuất cơ sở</li> <li>Thiết kế và cài đặt các chương trình thực hiện tác vụ vào ra với các ký tự</li> <li>Thiết kế và cài đặt các chương trình thực hiện tác vụ vào ra với tập tin</li> </ol>	<p>3</p> <p>2</p> <p>2</p>

#### Chủ đề 5: Xử lý ngoại lệ.

Nội dung	Mức độ
<b>Kiến thức</b> <ol style="list-style-type: none"> <li>Khái niệm về ngoại lệ (exception), các loại ngoại lệ</li> <li>Các lớp ngoại lệ trong Java</li> <li>Mô hình xử lý ngoại lệ (Các khối try, catch, throw, throws, finally)</li> </ol>	<p>2</p> <p>2</p> <p>3</p>

<b>Thái độ</b>	
<ol style="list-style-type: none"> <li>1. Khi viết chương trình, lập trình viên khó kiểm soát hết mọi lỗi có thể xảy ra, đặc biệt là các lỗi trong thời gian thực.</li> <li>2. Lỗi xảy ra trong thời gian thực, nếu không được kiểm soát, xử lý, có thể gây ngắt chương trình hoặc phá vỡ hệ thống.</li> </ol>	
<b>Kỹ năng</b>	
<ol style="list-style-type: none"> <li>1. Định nghĩa và xây dựng một lớp ngoại lệ</li> <li>2. Sử dụng mô hình xử lý ngoại lệ thích hợp cho các ứng dụng</li> </ol>	<p>2</p> <p>2</p>

### Chủ đề 6: Xây dựng ứng dụng với giao diện đồ họa.

Nội dung	Mức độ
<b>Kiến thức</b>	
<ol style="list-style-type: none"> <li>1. Đặc điểm và mô hình tổ chức của ứng dụng đồ họa</li> <li>2. Sự kiện và mô hình xử lý sự kiện</li> <li>3. Giao diện đồ họa người dùng (Graphic User Interface - GUI) với AWT và SWING</li> </ol>	<p>2</p> <p>2</p> <p>3</p>
<b>Thái độ</b>	
<ol style="list-style-type: none"> <li>1. Các ứng dụng đồ họa thường hỗ trợ việc kích hoạt các sự kiện từ chuột hoặc bàn phím và mỗi sự kiện phát sinh luôn cần có cơ chế xử lý thích hợp.</li> <li>2. Giao diện đồ họa giúp cho các ứng dụng trở nên thân thiện hơn với người sử dụng</li> <li>3. Thư viện AWT, đặc biệt là SWING giúp xây dựng giao diện đồ họa cho các ứng dụng một cách linh hoạt, nhanh chóng</li> </ol>	
<b>Kỹ năng</b>	
<ol style="list-style-type: none"> <li>1. Vận dụng mô hình lập trình hướng sự kiện vào vào các ứng dụng đồ họa</li> <li>2. Xây dựng giao diện đồ họa cho các ứng dụng với thư viện AWT và SWING</li> </ol>	<p>2</p> <p>2</p>

### Chủ đề 7: Lập trình socket

Nội dung	Mức độ
<b>Kiến thức</b>	
<ol style="list-style-type: none"> <li>1. Các khái niệm cơ bản về địa chỉ mạng, địa chỉ tiến trình, socket</li> <li>2. Các lớp trong Java API hỗ trợ lập trình socket</li> </ol>	<p>2</p> <p>2</p>
<b>Thái độ</b>	
<ol style="list-style-type: none"> <li>1. Socket là API hỗ trợ các phương thức kết nối giúp điều khiển quá trình truyền thông giữa các tiến trình trên cùng một máy tính hoặc giữa các máy tính khác nhau</li> <li>2. Sử dụng cơ chế truyền thông socket, người lập trình có thể dễ dàng xây dựng các ứng dụng mạng theo mô hình client/server mà không cần hiểu biết sâu về hạ tầng mạng vật lý bên dưới</li> </ol>	
<b>Kỹ năng</b>	
<ol style="list-style-type: none"> <li>1. Thiết kế và cài đặt các ứng dụng mạng cơ bản dựa trên TCP socket.</li> <li>2. Thiết kế và cài đặt các ứng dụng mạng cơ bản dựa trên UDP socket.</li> </ol>	<p>2</p> <p>2</p>

#### 4. Phân bổ thời gian chi tiết

Chủ đề	Phân bổ số tiết cho hình thức dạy - học					Tổng
	Lên lớp			Thực hành, thực tập	Tự nghiên cứu	
	Lý thuyết	Bài tập	Thảo luận			
1	3	2		6	15	26
2	4	2		4	18	28
3	3		1	4	12	20
4	3		1	4	12	20
5	3		1	4	12	20
6	3		1	4	12	20
7	3			4	9	16

#### 5. Tài liệu

(Tài liệu kê theo thứ tự ưu tiên sử dụng, kê tối thiểu 4 tài liệu)

TT	Tên tác giả	Tên tài liệu	Năm xuất bản	Nhà xuất bản	Địa chỉ khai thác tài liệu
1	Daniel Liang	Introduction to Java Programming, 6th Edition	2007	Pearson Prentice Hall	Thư viện
2	Đoàn Văn Ban	Lập trình hướng đối tượng với Java	2005	NXB Khoa học và Kỹ thuật	
3	Florence Tiu Balagtas	Introduction to Programming I - Version 1.3	2006	J.E.D.I	Internet
4	Cay S. Horstmann	Core Java V1,2	2008	Prentice Hall	Internet

#### 6. Đánh giá kết quả học tập

TT	Các chỉ tiêu đánh giá	Phương pháp đánh giá	Trọng số (%)
1	Tham gia học trên lớp (TGH): <i>tham dự đầy đủ các buổi học; tích cực thảo luận và phát biểu ý kiến xây dựng bài.</i>	Quan sát, điểm danh	50
2	Tự nghiên cứu: (TNC): <i>hoàn thành bài tập cá nhân; đọc hiểu tài liệu.</i>	Chấm báo cáo	
3	Hoạt động nhóm	<i>Trình bày báo cáo</i>	
4	Kiểm tra giữa kỳ (KT)	Viết	
5	Kiểm tra đánh giá cuối kỳ (KTCK)	Thực hành	50
6	Thi kết thúc học phần (THP)	Viết	

TRƯỞNG KHOA

TRƯỞNG BỘ MÔN