

Giới thiệu tổng quan môn CTDL&TT

Tại sao phải học CTDL&TT

- Khối lượng dữ liệu phải xử lý: càng ngày càng nhiều
 - ngày càng lớn và phức tạp,
 - dữ liệu tới từ nhiều nguồn: text, web, audio, image, video,...
- Tốc độ phải xử lý: phải nhanh hơn
 - Dữ liệu nhiều, tốc độ xử lý cần tăng lên
 - Xử lý dữ liệu một cách hiệu quả hơn
- Số lượng request – yêu cầu xử lý ngày một nhiều
 - Số lượng yêu cầu đồng thời lớn
 - Các yêu cầu ngày càng phức tạp hơn

Tại sao phải học CTDL&TT

- Học lập trình với 1 ngôn ngữ lập trình liệu đã đủ?
 - Chương trình chạy được và chạy một cách hiệu quả là hai khái niệm khác nhau
 - Chương trình liên tục được tối ưu để đáp ứng các yêu cầu liên tục thay đổi
 - Tốc độ xử lý, khả năng lưu trữ và kết nối của máy tính ngày càng thay đổi nhiều: cần tận dụng hết khả năng của máy tính
- Môn học này sẽ bổ trợ rất nhiều kiến thức quan trọng cho LTV để có thể xây dựng được chương trình hiệu quả hơn

Nội dung cần nắm được

- Khái niệm cơ bản về thuật toán, cấu trúc dữ liệu
- Phương pháp đánh giá hiệu quả thuật toán
- Một số kỹ thuật thiết kế thuật toán
 - Vét cạn, tham lam
 - Chia để trị, quy hoạch động
- Cấu trúc dữ liệu tuyến tính
 - Danh sách, Stack, Queue
- Cấu trúc dữ liệu cây

Nội dung cần nắm được

- Một số thuật toán tìm kiếm cơ bản và nâng cao
 - Tìm kiếm tuần tự, nhị phân
 - Cây nhị phân tìm kiếm, AVL, RB-Tree
 - Bảng băm
- Một số thuật toán sắp xếp cơ bản và nâng cao
 - Sắp xếp lựa chọn, chèn, nổi bọt
 - Sắp xếp nhanh, trộn, vun đống
 - Một số thuật toán sắp xếp đặc biệt
- Đồ thị và một số bài toán cơ bản
 - Đồ thị, ma trận kề, danh sách kề
 - Duyệt đồ thị, đường đi, cây khung

Đánh giá

- Gồm 2 điểm
 - Điểm quá trình : 40%
 - Điểm cuối kỳ: 60%
- Hình thức đánh giá: thi viết (ngoại trừ trường hợp đặc biệt)
- Extra ?
 - Bài tập làm thêm? (cho điểm quá trình)
- Ngôn ngữ lập trình: C/C++, Java, Python,..
 - Nên dùng các thư viện chuẩn có sẵn: VD. STL trong C/C++

Tài liệu tham khảo

1. Nguyễn Đức Nghĩa. Bài giảng CTDL>. DHBKHN
2. Đỗ Xuân Lôi. Cấu trúc dữ liệu và giải thuật. Nhà xuất bản ĐHQG Hà nội, 2005.
3. Alfred V. Aho, John E. Hopcroft, and Jeffrey D. Ullman. *Data Structures and Algorithms*. Addison-Wesley, 1983.
4. Robert Sedgewick. *Algorithms in C*. Third Edition. Addison-Wesley, 1998.
5. Robert Sedgewick. *Algorithms in C++, Parts 1-4: Fundamentals, Data Structures, Sorting, Searching*. 3th Edition, Addison-Wesley, 1999.
6. Robert Sedgewick. *Algorithms in C++ Part 5: Graph Algorithms (3rd Edition)*. 3th Edition, Addison-Wesley, 2002.
7. Michael T. Goodrich, Roberto Tamassia, David M. Mount, *Data Structures and Algorithms in C++*. 704 pages. Wiley, 2003.
8. T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein. *Introduction to Algorithms* . Second Edition, MIT Press, 2001.