

Data Structure and Algorithms

Syllabus Number

3B209

Basic Major Subjects

Elective Requisites 2
credit

ARAI, Masayuki

1. Course Description

We will learn the followings:

- (1) Time complexity and space complexity
- (2) String matching algorithm
- (3) Sort algorithm
- (4) Basic data structures
- (5) Shortest path algorithm
- (6) Minimum spanning tree algorithm

This course is related to DP4C and DP4M.

2. Course Objectives

The aim of this course is followings:

- (a) The learners comprehend basic data structures and apply them to programs.
- (b) The learners comprehend basic algorithm and use it appropriately.
- (c) The learners can evaluate algorithm from point of complexity.
- (d) The learners can select suitable algorithm and data structure according to the problem.

3. Grading Policy

The learners are assessed by the followings: reports 30%, mini tests 30%, a term-end examination 40%. The learners who get over 60% can get credits. For reexamination, examinees who get over 60% can get credits.

The learners can get feedback from the reports in which professors write comments and explanation for the mini tests.

4. Textbook and Reference

Textbook

Satoshi Fujita: "Algorithm and data structures," Graphic Information Library, Saiensu-sha Co., Ltd., 2013, ISBN978-4-901683-99-9.

5. Requirements(Assignments)

1. Overview of this course and complexity

Preparation: reading the section 1 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 1 (1.5 hours)

2. String matching algorithm: what are string matching problems

Preparation: reading the subsections 2.1 and 2.2 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 2 (1.5 hours)

3. String matching algorithm: KMP

Preparation: reading the subsections 2.3 and 2.4 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 2 and programming the algorithm 2.4 of the textbook (1.5 hours)

4. Sort algorithm: Selection sort and insert sort

Preparation: reading the subsections 3.1 ~ 3.3 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 3 (1.5 hours)

5. Sort algorithm: Divide-and-conquer method and quick sort

Preparation: reading the subsections 3.4 and 3.5 of the textbook carefully and confirm keywords (1.5 hours)

Review: programming the algorithm 3.4 of the textbook (1.5 hours)

6. Sort algorithm: Merge sort and radix sort

Preparation: reading the subsections 3.6 and 3.7 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 3 and programming the algorithm 3.5 of the textbook (1.5 hours)

7. Basic data structures: Queue and stack

Preparation: reading the subsections 4.1 and 4.2 of the textbook carefully and confirm keywords (1.5 hours)

Review: programming linked list and stack (1.5 hours)

8. Basic data structures: Binary tree and balanced tree

Preparation: reading the subsections 4.3 and 4.4 of the textbook carefully and confirm keywords (1.5 hours)

Review: programming the binary tree (1.5 hours)

9. Basic data structures: Heap and disjoint-set data structure

Preparation: reading the subsections 4.5 and 4.6 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 4 and programming the heap (1.5 hours)

10. Shortest path algorithm: How to express graph structures

Preparation: reading the subsections 5.1 and 5.2 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 5 (1.5 hours)

11. Shortest path algorithm: Dijkstra's algorithm

Preparation: reading the subsection 5.3 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 5 (1.5 hours)

12. Shortest path algorithm: Bellman-Ford algorithm

Preparation: reading the subsection 5.4 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 5 (1.5 hours)

13. Minimum spanning tree algorithm: Best-first search algorithm

Preparation: reading the subsections 6.1 and 6.2 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 6 (1.5 hours)

14. Minimum spanning tree algorithm: Kruskal's algorithm and Prim's algorithm

Preparation: reading the subsections 6.3 and 6.4 of the textbook carefully and confirm keywords (1.5 hours)

Review: solving the problems in the end of the section 6 (1.5 hours)

15. Summarization and examination

Preparation: summarizing this course (1.5 hours)

Review: reviewing the exam (1.5 hours)

6. Note

The learners sometimes write programs, therefore they should review the courses of Programming 1 and Programming 2.

The learner can use leaning materials uploaded to LMS.

This course is a required subject for JABEE and corresponds to 4.2 in the learning and educational objectives.

7. Schedule

- [1] Overview of this course and complexity
- [2] String matching algorithm: what are string matching problems
- [3] String matching algorithm: KMP
- [4] Sort algorithm: Selection sort and insert sort
- [5] Sort algorithm: Divide-and-conquer method and quick sort
- [6] Sort algorithm: Merge sort and radix sort
- [7] Basic data structures: Queue and stack
- [8] Basic data structures: Binary tree and balanced tree
- [9] Basic data structures: Heap and disjoint-set data structure
- [10] Shortest path algorithm: How to express graph structures
- [11] Shortest path algorithm: Dijkstra's algorithm
- [12] Shortest path algorithm: Bellman-Ford algorithm
- [13] Minimum spanning tree algorithm: Best-first search algorithm
- [14] Minimum spanning tree algorithm: Kruskal's algorithm and Prim's algorithm
- [15] Summarization and examination