

TIPPS FÜR DIE LERNPHASE

(Meine persönliche Erfahrung)

ALGORITHMEN UND DATENSTRUKTUREN

Keine Garantie, dass eure Prüfung so aussehen wird!
Soll aber im gleichen Stil wie letzten Jahre werden.

- Schriftliche Prüfung
 - Aufgabe 1: Multiple Choice Aufgaben, Kurzaufgaben
 - Aufgabe 2: Analyse von gesehenen Algorithmen / Design von einfachen Algorithmen
 - Aufgabe 3: Dynamic Programming
 - Aufgabe 4: Graphen (shortest paths, minimum spanning trees, graph layering)
- Programmierprüfung
 - Aufgabe 1: Graphen
 - Aufgabe 2: Dynamic Programming

SCHRIFTLICHE PRÜFUNG: AUFGABE 1

- Asymptotic Notation Quiz
- Search Trees (AVL, Binary, Heaps)
- Graph Quiz
- Algorithmen ausführen an gegebenen Graphen (BFS, DFS, Top. Sort, MST, Shortest Paths)
- Sorting Algorithms Quiz
 - Verständnis der Algorithmen und Invarianten sehr wichtig

SCHRIFTLICHE PRÜFUNG: AUFGABE 2

- Counting loop iterations
 - Hierfür sind Summenformeln sehr wichtig!
- Induktion
- Eigenschaften zeigen (von bestimmten Graphen oder Algorithmen)
 - Hierfür ist Verständnis der Algorithmen sehr wichtig, teilweise auch Herleitung
- Einfache Algorithmen entwerfen (meistens modifizierte BFS-/DFS- Varianten)

SCHRIFTLICHE PRÜFUNG: AUFGABE 3

- Fast immer modifizierte Variante eines DP-Algorithmus aus der Vorlesung, z.B. Subset Sum, LAT
- Im bekannten Schema Aufgabe ausfüllen (Size of table, meaning of an entry, computation of an entry, ...)
- Tipps hierfür:
 - Zuerst aus der Aufgabenstellung den Algorithmus aus der Vorlesung identifizieren und Rekursion ins Gedächtnis rufen (am besten aufschreiben auf anderes Blatt)
 - Abänderung zum gegebenen Problem identifizieren (z.B. weitere Einschränkungen im Vergleich zum Problem aus der Vorlesung, oder Ausweitung des Problems)
 - Modifikation in neue Rekursion einschließen (neue Base Cases, neues „ / „ in der Rekursion, ...)
 - Entworfenen DP-Algorithmus auf Korrektheit überprüfen (am Besten kurz an Beispiel durchgehen)

SCHRIFTLICHE PRÜFUNG: AUFGABE 4

- Involvieren meistens viel Text
- Problem als Text beschrieben, Modellierung als Graphenproblem gesucht
- Über mehrere Teilaufgaben werden verschiedene Probleme betrachtet, z.B. kürzeste Pfade oder Minimaler Spannbaum
- Graph-Layering als mögliche Lösung berücksichtigen
- Versucht Graph so zu modellieren, sodass Problem direkt mit einem in der Vorlesung vorgestellten Algorithmus gelöst werden kann

PROGRAMMIERPRÜFUNG AND

- DP Aufgabe
 - Zuerst auf Papier lösen! Erst implementieren, wenn man die Lösung auf dem Papier hat
- Graphen Aufgabe
 - Algorithmen aus Vorlesung implementieren können (vor allem die einfacheren)
 - Es wird viel DFS/BFS gebraucht (hierfür mit der Implementation vertraut sein, und zusätzliche Modifikationen implementieren können)
- Aufgabenstellung zuerst **genau** verstehen
- Zuerst Aufgabe im Kopf / auf dem Papier lösen, danach implementieren
 - Falls man zuerst falschen Ansatz implementiert, verliert man viel Zeit

EINFÜHRUNG IN DIE PROGRAMMIERUNG

- Theorieteil 1/3
 - Sehr ähnlicher Stil wie letzten Jahre, so gut wie die gleichen Aufgabentypen kommen immer dran
- Programmierteil 2/3
 - Immer im gleichen Stil
- Unterschätzt EProg nicht, auch wenn ihr schon programmieren könnt.
 - Es lohnt sich, viel für EProg zu üben!
 - Indem ihr viele alte Prüfungen löst, lernt ihr diesen Aufgabenstil (aus beiden Teilen) effizient zu lösen
- Beide Teile direkt hintereinander (wie in richtiger Prüfung) auf Zeit mit alten Prüfungen üben

DISKRETE MATHEMATIK

- 4 Aufgabenteile in jeder Prüfung
- Man muss nicht jedes Thema ins genaueste Detail verstehen in Diskrete Mathematik
 - Vor allem die schwierigeren Themen
- Es ist vor allem wichtig, die „Basics“ sehr gut zu können, und ein gutes allgemeines Verständnis von den Themen zu haben
- Übt mit alten Prüfungen, um typische Aufgaben effizient lösen zu können

ALLGEMEINE TIPPS

- Fangt möglichst früh an, zu lernen
- Verwendet nicht zu viel (aber auch nicht zu wenig) Zeit, die Theorie zu wiederholen
 - Verschwendet keine Zeit mit Schreiben von Zusammenfassungen
- Löst so viele Prüfungen wie möglich
 - Am besten alle, die zum aktuellen Stil der Kurse passen (pro Fach sind das mehr als 8)
 - Früh anfangen, alte Prüfungen zu lösen, da es sehr viele zum Üben gibt
- Löst die Prüfungen unter “Prüfungsbedingungen”
 - Also ohne zwischendurch die Lösung anzuschauen, und mit Zeitdruck
- Nicht zu spät die erste Altprüfung in einem Kurs lösen
 - Es ist sehr hilfreich wenigstens eine alte Prüfung recht am Anfang der Lernphase zu lösen, um zu verstehen, wie eine typische Prüfung aufgebaut ist, und welche Themen vor allem wichtig sind