Професионално направление 4.6 Информатика и компютърни науки

КОНСПЕКТ

за кандидатдокторантски изпит

1. Алгоритми – същност и неформално определение. Изчислителни задачи. Големина на входа на алгоритми. Коректност на алгоритми: доказателства по индукция за рекурсивни алгоритми и чрез инварианта на цикъла за итеративни алгоритми. Сложност по време в най-лошия случай, средна и в най-добрия случай. Асимптотични нотации. Сложност по памет.

2. Сортиране – приложения и важност. Елементарни сортиращи алгоритми: Selection Sort и Insertion Sort. Анализ на коректността им чрез инварианта на цикъла. Двоична пирамида. Построяване на пирамида чрез функции Heapify и Build Heap. Анализ на сложността им. Пирамидална сортировка Heapsort. Приоритетни опашки като абстрактен тип данни (АТД). Имплементация на приоритетни опашки чрез двоични пирамиди.

3. Рекурсивни алгоритми и рекурентни уравнения. Методи за решаване на рекурентни уравнения: развиване, дърво на рекурсията, индукция, Master Theorem, методът с характеристичното уравнение. Алгоритмична схема Разделяй и владей: същност, приложение и ограничения върху възможностите за прилагането ѝ. Сортиращ алгоритъм Mergesort. Анализ на коректността на Mergesort чрез инварианта на цикъла. Анализ на сложността по време на Mergesort.

4. Сортиращ алгоритъм Quicksort. Функцията Partition в Quicksort: варианти на Hoare и Lomuto. Анализ на коректността на Partition чрез инварианта на цикъла. Анализ на коректността, сложност по време и памет на Quicksort. Анализ на средната сложност по време на Quicksort.

5. Долни граници върху асимптотичната сложност по време на алгоритми. Дървета на решението (decision trees). Метод за доказване на долни граници, базиран дървета на решението. Долна граница върху асимптотичната сложност по време на сортиращи алгоритми, базирани на директно сравнение. Долна граница върху асимптотичната сложност по време на изчислителната задача Element Uniqueness. Доказване на асимптотични долни граници чрез редукции между изчислителни задачи.

6. Ориентирани и неориентирани графи. Представяния на графи чрез матрици на съседства и чрез списъци на съседства. Сравнителен анализ на предимствата и недостатъците на тези представяния. Обхождане в ширина (BFS) на графи. Дърво на обхождането в ширина. Видове ребра в обхождането в ширина. Обхождане в дълбочина (DFS) на графи. Дърво на обхождането в дълбочина. Видове ребра в обхождането в дълбочина при неориентираните и ориентираните графи.

7. Ориентирани ациклични графи (dags). Алгоритми за топологично сортиране на ориентирани ациклични графи. Анализ на сложността по време на тези алгоритми. Алгоритъм с линейна сложност по време за откриването на силно свързаните компоненти на ориентирани графи. Срязващи върхове (cut vertices) и срязващи ребра (bridges) в неориентираните графи. Двусвързани компоненти в неориентирани графи. Алгоритми с линейна сложност по време за откриване на срязващите върхове на граф и на срязващите ребра на граф.

8. Минимални покриващи дървета на неориентирани графи. Примери за приложения на минимални покриващи дървета. Обща алгоритмична схема за алгоритми за откриване на минимални покриващи дървета. Лакоми (greedy) стратегии. Алгоритми на Prim и на Kruskal за минимално покриващо дърво. Коректност и сложност по време на двата алгоритъма. Union-Find операции върху множества. Евристики Union by rank и Path compression. Анализ на сложността по време на Union-Find операциите при използването на двете евристики.

9. Най-къси пътища в ориентираните графи. Примери за приложения. Потенциални трудности при наличието на отрицателни тегла. Анализ на задачата и сравнението и със задачата за най-дълъг път в граф. Алгоритъм на Dijkstra за намиране на най-късите пътища от даден връх до всички останали върхове. Анализ на коректността и сложността по време на този алгоритъм. Алгоритъм на Floyd-Warshall.

10. Динамично програмиране – същност и приложения. Примери:
    - изчисляване на числа на Фибоначи и на биномни коефициенти.
    - оптимално верижно умножение на матрици (Chain Matrix Multiplication),
    - изчислителната задача Vertex Cover върху дърветата,
    - задачата за раницата (Knapsack Problem),
    - оптимална триангулация на изпъкнал полигон.
    - други примери.

11. Потоци в графи. Методът на Ford-Fulkerson. Резидуални капацитети и резидуални графи. Пътища за нарастване (augmenting paths). Срезове (cuts) в графи с потоци. Max-flow min-cut теорема. Алгоритъм на Edmonds-Karp. Анализ на този алгоритъм. Примери за приложението на потоци в графи: задачата за максимално съчетание (matching) в двуделни графи.

12. Съвпадение на подстрингове (String Matching). Наивен алгоритъм. Алгоритъм на Rabin-Karp. Намиране на съвпадение на подстрингове с крайни автомати; изчисляване на таблицата на преходите на автомата. Алгоритъм на Knuth-Morris-Pratt (KMP). Анализ на KMP.

13. Практически решими изчислителни задачи – клас на сложност **P**. Практически нерешими (intractable) задачи. Недетерминирани изчисления. Интерпретации в детерминиран модел. Клас на сложност по време **NP**: лесните за проверяване задачи. Полиномиални (Karp) редукции и Turing редукции. Частична наредба в **NP**. **NP**-трудност и **NP**-пълнота. Нерешеният въпрос $\mathbf{P} \stackrel{?}{=} \mathbf{NP}$ и неговото значение.

14. Теорема на Cook.

15. Основни **NP**-пълни задачи: 3sat, Independent Set, Vertex Cover, Dominating Set, Hamiltonian Cycle, Hamiltonian Path, Clique, Partition, 3D Matching, 3Coloring.

16. Клас на сложност **coNP**. Полиномиалната йерархия **PH**. Клас на сложност **PSPACE**. Алтерниране и игри. Пълни задачи за **PSPACE**: QSAT, Geography, Go.

17. Ефикасни апроксимиращи алгоритми за практически нерешими оптимизационни задачи. Фактор на апроксимируемост. Примери за ефикасни апроксимиращи алгоритми с константен фактор за **NP**-пълни задачи: Vertex Cover, Metric Steiner Tree, Metric TSP, Feedback Vertex Set, Shortest Superstring.

18. Апроксимиращи схеми в полиномиално време (PTAS) и апроксимиращи схеми в напълно полиномиално време (FPTAS). FPTAS за задачата Knapsack и PTAS за задачата Bin Packing. L-свеждания и апроксимиране. Клас на сложност MAXSNP. Пълни задачи за MAXSNP: MAX3SAT.

19. Условна невъзможност за апроксимиране (спрямо допускането, че $\mathbf{P} \neq \mathbf{NP}$). Невъзможност за апроксимиране с константен фактор на TSP, Independent Set и Clique. Невъзможност за PTAS за MAX3SAT.

20. Теория на параметризираната сложност: ефикасни алгоритми за практически нерешими задачи при малка стойност на параметър на задачата. Въведение чрез пример с Vertex Cover. Клас на сложност **FPT**.

Литература

[CLRS09] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, and Clifford Stein. *Introduction to Algorithms, Third Edition.* The MIT Press, 3rd edition, 2009.

[DF99] Rodney G. Downey and Michael R. Fellows. *Parameterized Complexity.* Springer-Verlag, 1999. 530 pp.

[FG06] J. Flum and M. Grohe. *Parameterized Complexity Theory (Texts in Theoretical Computer Science. An EATCS Series).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[GJ79] Michael R. Garey and David S. Johnson. *Computers and Intractability: A Guide to the Theory of NP-Completeness.* W. H. Freeman & Co., New York, NY, USA, 1979.

[Man12] Krasimir Manev. *Uvod v Diskretnata Matematika.* KLMN – Krasimir Manev, fifth edition, 2012.

[Pap94] Christos M. Papadimitriou. *Computational complexity.* Addison-Wesley, Reading, Massachusetts, 1994.

[SF96] Robert Sedgewick and Philippe Flajolet. *An introduction to the analysis of algorithms.* Addison-Wesley-Longman, 1996.

[Ski08] Steven S. Skiena. *The Algorithm Design Manual.* Springer Publishing Company, Incorporated, 2nd edition, 2008.

[SW11] Robert Sedgewick and Kevin Wayne. *Algorithms, 4th Edition.* Addison-Wesley, 2011.

[Vaz01] Vijay V. Vazirani. *Approximation Algorithms.* Springer-Verlag New York, Inc., New York, NY, USA, 2001.