

- ۱- مفهوم الگوریتم: توضیح مختصر درباره اینکه ارایه دقیق تعریف الگوریتم و وجود آن برای مسایل مختلف، موضوعی پیچیده است و ایجاد انگیزه برای زمینه تخصصی نظریه محاسبه. توضیح مختصر تر چرچ تورینگ به زبان ساده. توضیح انواع مختلف مسایل و نیاز به روش های مختلف برای حل آنها از قبیل مسایل تصمیم (بله خیر)، شمارش، بهینه سازی و انواع دیگر با تأکید بر اینکه لازم است راه حل های مختلف را رده بندی کرد. ایجاد انگیزه برای زمینه ی پیچیدگی محاسبه و طراحی الگوریتم (مثال های مناسب ارایه شوند)
- ۲- الگوریتم های دقیق: ارایه مثال هایی مناسب که برای آنها الگوریتم دقیق و کارآ وجود دارند) مثال های مرتب کردن Sort، الگوریتم های بی از مسایل گراف، هندسه محاسباتی و انواع خاص حل پذیری معادلات منطقی نظیر SAT ۲ میتواند مفید باشند). ارایه مثال ی از بهینه سازی خطی حقیقی بسیار انگیزه بخش است، به ویژه بحث در مورد حالت های حقیقی و صحیح همراه با پیچیدگی های مربوط که در بخش های آینده میتواند راهگشا باشد.
- ۳- مسایل سخت: تأکید بر اینکه تعداد زیادی از مسایل وجود دارند که در عین مهم بودن، تاکنون راه حل کارایی برای آنها یافت نشده اند. تعریف (نه لزوماً دقیق) رده های P و NP در حدی که دانشجو بتواند اهمیت را درک کند و فهم صحیحی از این مفاهیم بیاید (به طور خواص، تأکید بر اختلاف این مدل با مدل پیچیدگی حساب ی مربوط به شمارش عملیات و اشاره به مفهوم Strong Polynomial-Time-Algorithm، با ارایه مثال هایی مناسب).
- ۴- ارایه روش های کلی: برنامه ریزی پویا Dynamic Programming، الگوریتم های حریصانه Greedy، برخط Online، ترتیبی Sequential یا موازی Parallel با ارایه مثال هایی مناسب (در صورت صلاح دید، میتوان از مسایل بهینه سازی ترکیبیاتی به نحوی مناسب استفاده کرد)
- ۵- الگوریتم های تقریبی: تعریف ضریب تقریب، مسأله TSP و اثبات تقریب ناپذیری آن و در صورت صلاح دید اشاره به قضیه PCP و نتایج آن. ارایه مثال هایی از مسایل سخت که میتوان آنها را با ضریب تقریب مناسبی حل کرد (با نظر استاد).
- ۶- الگوریتم های تصادفی: ارایه تعریف، رده های مربوط، بهیژه RP و ZP ارایه چند مثال (ترجیحاً از مسایل قبلی) که با روش های تصادفی قابل حل هستند و مقایسه الگوریتم ها (مثلاً مسأله Sort میتواند در این قسمت مناسب باشد). ارایه روش های طراحی الگوریتم های تصادفی در حد امکان.
- ۷- الگوریتم های هوشمند: تأکید بر وجود مسایل سخت که در حال حاضر حتی با روش های قبلی نیز راه حل های مناسب ندارند. ایجاد انگیزه برای زمینه تخصصی هوش مصنوعی. ارایه ایده های اصلی طراحی الگوریتم های ابتکاری (با ارایه مثال مطابق نظر استاد). تأکید بر لزوم وجود ابزار نظری مربوط به همگرایی سیستم های دینامیکی گسسته و طرح ایده های اصلی قضیه های شهباز گدیک (مثال زنجیر مارکوف و چگونگی کاربرد آن در طراحی الگوریتم برای مسائل شمارش و وجود، به ویژه الگوریتم های از نوع SA راهگشا هستند). ورود به موضوع محاسبات نرم (الگوریتم های نرم و داده های نرم) و ایجاد انگیزه های لازم در این مورد.
- ۸- مباحث منتخب برای تکمیل مفاهیم درس (با نظر استاد)

1. R. K. Ahuja, T. L. Magnanti, J. B. Orlin, Network Flows: Theory, Algorithms, and Applications, Prentice Hall, 1993.
2. T. H. Corman, C. E. Liserion, and R.L. Rimest, Introduction to Algorithms, MIT Press, 2009.
3. J. Hromkovic, Design and Analysis of Randomized Algorithms, Introduction to Design Paradigms, Springer 2005.
4. J. Hromkovic, Algorithms for Hard Problems, Springer 2001.
5. B. Korte and J. Vygen, Combinatorial Optimization, Theory and Algorithms, Springer-Verlag, 2008.
6. E. Tardos and J. Kleinberg, Algorithm Design, Pearson 2006.
7. V. V. Vazirani, Approximation Algorithms, Springer 2001.