



دانشگاه صنعتی شریف
دانشکده مهندسی کامپیوتر

برنامه درسی و ریز مواد دروس کارشناسی ارشد مهندسی کامپیوتر

گرایش نرم افزار

* مخصوص ورودی‌های ۱۳۹۳ به بعد *

آخرین ویرایش: شهریور ۱۳۹۸

فهرست مطالب

۴	مقررات عمومی گرایش نرم افزار.....
۵	دروس جبرانی.....
۵	دروس زیرگرایش ها.....
۶	دروس سمینار و پروژه.....
۷	دروس زیرگرایش الگوریتم ها و محاسبات.....
۸	شبکه های اجتماعی و اقتصادی (۴۰۶۴۲).....
۱۰	پردازش موازی (۴۰۶۴۷).....
۱۲	الگوریتم های تصادفی (۴۰۶۸۵).....
۱۴	الگوریتم های داده های حجیم (۴۰۶۹۶).....
۱۶	هندسه محاسباتی (۴۰۷۳۵).....
۱۸	الگوریتم های پیشرفته (۴۰۷۶۵).....
۲۰	نظریه پیچیدگی (۴۰۷۷۵).....
۲۲	بهینه سازی ترکیبیاتی (۴۰۷۸۵).....
۲۳	الگوریتم های تقریبی (۴۰۸۳۴).....
۲۵	نظریه الگوریتمی بازی ها (۴۰۸۳۵).....
۲۷	دروس زیرگرایش سیستم ها.....
۲۸	سیستم های عامل پیشرفته (۴۰۵۳۳).....
۳۰	سیستم های پشتیبانی تصمیم گیری (۴۰۵۴۸).....
۳۲	معماری سازمانی فناوری اطلاعات (۴۰۶۳۶).....
۳۴	مدل سازی و تحلیل شبکه های کامپیوتری (۴۰۶۵۷).....
۳۶	نظریه سیستم های توزیع شده (۴۰۶۶۱).....
۳۸	درستی بای سیستم های واکنشی (۴۰۶۶۵).....
۴۰	وب معنایی (۴۰۶۹۴).....
۴۲	داده کاوی (۴۰۷۱۴).....
۴۴	ارزیابی کارایی کامپیوترها (۴۰۸۲۴).....
۴۶	طراحی پایگاه داده ها ۲ (۴۰۹۳۸).....
۴۸	دروس زیرگرایش مهندسی نرم افزار.....
۴۹	معماری نرم افزار (۴۰۶۴۶).....
۵۱	تکامل نرم افزار (۴۰۶۷۹).....
۵۳	تولید برنامه از توصیف رسمی (۴۰۶۸۴).....
۵۶	مهندسی نیازمندی های نرم افزار (۴۰۶۸۸).....
۵۸	متدولوژی های ایجاد نرم افزار (۴۰۷۲۴).....
۶۰	توصیف و واریسی برنامه ها (۴۰۷۴۵).....
۶۲	الگوها در مهندسی نرم افزار (۴۰۷۸۷).....
۶۴	آزمون نرم افزار (۴۰۸۲۸).....
۶۶	مهندسی نرم افزار ۲ (۴۰۹۲۴).....

برنامه کارشناسی ارشد مهندسی کامپیوتر- گرایش نرم افزار

* مخصوص ورودی‌های ۱۳۹۳ به بعد *

برنامه کارشناسی ارشد مهندسی کامپیوتر - گرایش نرم افزار

ویژه دانشجویان ورودی ۱۳۹۳ به بعد*

(این نسخه‌ی روزآمدشده در تاریخ ۹۸/۴/۵ به تصویب گروه نرم افزار رسیده است)

مقررات عمومی گرایش نرم افزار

- طول دوره: طول دوره ۲ سال است.
- زیرگرایش‌ها: گرایش نرم افزار دارای سه زیرگرایش به قرار زیر است (دروس متعلق به زیرگرایش‌ها در جداول ۲ تا ۴ آورده شده‌اند):
 - زیرگرایش الگوریتم‌ها و محاسبات
 - زیرگرایش سیستم‌ها
 - زیرگرایش مهندسی نرم افزار
- واحدها: تعداد واحدهای دوره برای دانشجویان پژوهش-محور ۲۹ واحد، و برای دانشجویان آموزش-محور ۳۲ واحد است. دانشجویان حق ندارند دروسی را که قبلاً در دوره کارشناسی گرفته‌اند مجدداً اخذ نمایند. گذراندن دروس زیرگرایش‌ها، سمینار و پروژه باید مطابق قواعد زیر انجام شود:
- دروس: حداقل ۴ درس از مجموعه دروس گذرانده شده دانشجوی پژوهش-محور باید همگی متعلق به یک زیرگرایش (از زیرگرایش‌های فوق) باشند، که زیرگرایش تخصصی دانشجو محسوب خواهد شد. حداقل ۲ درس از مجموعه دروس گذرانده شده نباید متعلق به زیرگرایش تخصصی دانشجو بوده و باید از زیرگرایش‌های دیگر گرایش نرم افزار انتخاب شده باشند؛ درس‌های غیرتخصصی می‌توانند همگی از یک زیرگرایش انتخاب شوند (محدودیتی در این زمینه وجود ندارد). در صورت موافقت استاد راهنما و مدیر گروه، حداکثر ۲ درس از دروس غیرتخصصی می‌توانند از گرایش‌ها یا رشته‌های دیگر دانشکده یا دانشگاه انتخاب شوند.
 - در مورد دانشجویان آموزش-محور، حداقل ۶ درس از مجموعه دروس گذرانده شده باید همگی متعلق به زیرگرایش تخصصی دانشجو بوده، و حداقل ۳ درس باید متعلق به زیرگرایش‌های دیگر نرم افزار باشند. سایر مقررات مانند دانشجویان پژوهش-محور خواهد بود.
- سمینار کارشناسی ارشد: ۲ واحد - مطابق جدول ۵
- پروژه کارشناسی ارشد: ۶ واحد - مطابق جدول ۵ (مختص دانشجویان پژوهش-محور)
- تصویب پروژه کارشناسی ارشد: مهلت تصویب تعریف پروژه کارشناسی ارشد، پایان نیمسال دوم است.
- ارائه سمینار کارشناسی ارشد: دانشجوی پژوهش-محور باید در نیمسال سوم و پس از تصویب پروژه، سمیناری در زمینه پروژه ارائه دهد که گزارش پیشرفت پروژه محسوب می‌شود. سمینار دانشجویان آموزش-محور نیز باید در نیمسال سوم و پس از تعیین موضوع و تأیید آن توسط گروه اخذ شود.
- همکاری با دانشکده: همه دانشجویان نوبت اول باید به صورت دستیار آموزشی یا با انجام امور ارجاعی دیگر (معادل ۲ واحد) با دانشکده همکاری کنند؛ همکاری با دانشکده برای دانشجویان آموزش-محور نیز اجباری است.
- استاد راهنما: مدیر گروه تا قبل از انتخاب و تصویب پروژه، استاد راهنمای دانشجویان است. پس از تصویب تعریف پروژه، استاد راهنمای پروژه هر دانشجو مسئول تأیید و راهنمایی وی در اخذ واحدهای درسی نیز می‌باشد.
- واحدهای جبرانی: در صورتی که گذراندن واحدهای جبرانی به تشخیص مدیر گروه برای دانشجو ضروری باشد، می‌بایستی حداکثر تا پایان سال اول تحصیل با تأیید گروه حداکثر چهار درس از دروس جدول ۱ را بگذرانند. اخذ ۸ واحد جبرانی یا بیشتر امکان افزایش حداکثر یک نیمسال تحصیلی را به سنوات تحصیلی دانشجو فراهم می‌کند. حداقل نمره قبولی برای دروس جبرانی ۱۲ است.

* ورودی‌های قبل از ۱۳۹۳ می‌توانند در صورت موافقت کمیته تحصیلات تکمیلی دانشکده، دروس خود را طبق این برنامه بگذرانند.

دروس جبرانی

جدول ۱. فهرست دروس جبرانی گرایش نرم افزار

ردیف	شماره	تعداد واحد	نام درس (فارسی)	نام درس (لاتین)
۱	۴۰۲۵۴	۳	ساختمان داده ها و الگوریتمها	Data Structures and Algorithms
۲	۴۰۳۵۴	۳	طراحی الگوریتمها	Design of Algorithms
۳	۴۰۴۱۴	۳	طراحی کامپایلرها	Compiler Design
۴	۴۰۴۱۵	۳	نظریه زبانها و ماشینها	Theory of Machines and Languages
۵	۴۰۴۲۴	۳	سیستمهای عامل	Operating Systems
۶	۴۰۴۷۴	۳	مهندسی نرم افزار	Software Engineering

دروس زیرگرایشها

جدول ۲. فهرست دروس زیرگرایش الگوریتمها و محاسبات

ردیف	شماره	تعداد واحد	نام درس (فارسی)	نام درس (لاتین)
۱	۴۰۶۴۲	۳	شبکههای اجتماعی و اقتصادی	Social and Economic Networks
۲	۴۰۶۴۷	۳	پردازش موازی	Parallel Processing
۳	۴۰۶۸۵	۳	الگوریتمهای تصادفی	Randomized Algorithms
۴	۴۰۶۸۶	۳	الگوریتمهای دادههای حجیم	Massive Data Algorithms
۵	۴۰۷۳۵	۳	هندسه محاسباتی	Computational Geometry
۶	۴۰۷۶۵	۳	الگوریتمهای پیشرفته	Advanced Algorithms
۷	۴۰۷۷۵	۳	نظریه پیچیدگی	Complexity Theory
۸	۴۰۷۸۵	۳	بهینه سازی ترکیبیاتی	Combinatorial Optimization
۹	۴۰۸۳۴	۳	الگوریتمهای تقریبی	Approximation Algorithms
۱۰	۴۰۸۳۵	۳	نظریه الگوریتمی بازیها	Algorithmic Game Theory
۱۱	۴۰۷۹۵	۳	مفاهیم پیشرفته علوم کامپیوتر	Advanced Topics in Computer Science
۱۲	۴۰۷۹۶	۳	مفاهیم پیشرفته در علوم کامپیوتر ۲	Advanced Topics in Computer Science 2
۱۳	۴۰۷۹۷	۳	مفاهیم پیشرفته علوم نظری کامپیوتر	Advanced Topics in Theoretical Computer Science

جدول ۳. فهرست دروس زیرگرایش سیستمها

ردیف	شماره	تعداد واحد	نام درس (فارسی)	نام درس (لاتین)
۱	۴۰۵۳۳	۳	سیستمهای عامل پیشرفته	Advanced Operating Systems
۲	۴۰۵۴۸	۳	سیستمهای پشتیبانی تصمیم گیری	Decision Support Systems
۳	۴۰۶۳۶	۳	معماری سازمانی فناوری اطلاعات	IT Enterprise Architecture
۴	۴۰۶۵۷	۳	مدل سازی و تحلیل شبکههای کامپیوتری	Computer Networks Modeling and Analysis
۵	۴۰۶۶۱	۳	نظریه سیستمهای توزیع شده	Theory of Distributed Systems
۶	۴۰۶۶۵	۳	درستی یابی سیستمهای واکنشی	Reactive Systems Verification
۷	۴۰۶۹۴	۳	وب معنایی	Semantic Web
۸	۴۰۷۱۴	۳	داده کاوی	Data Mining
۹	۴۰۸۲۴	۳	ارزیابی کارایی کامپیوترها	Computer Performance Evaluation
۱۰	۴۰۹۳۸	۳	طراحی پایگاه دادهها ۲	Database Design 2
۱۱	۴۰۹۲۶	۳	مفاهیم پیشرفته سیستمهای نرم افزاری	Advanced Topics in Software Systems

جدول ۴. فهرست دروس زیرگرایش مهندسی نرم افزار

ردیف	شماره	تعداد واحد	نام درس (فارسی)	نام درس (لاتین)
۱	۴۰۶۴۶	۳	معماری نرم افزار	Software Architecture
۲	۴۰۶۷۹	۳	تکامل نرم افزار	Software Evolution
۳	۴۰۶۸۴	۳	تولید برنامه از توصیف رسمی	Formal Program Development
۴	۴۰۶۸۸	۳	مهندسی نیازمندی های نرم افزار	Software Requirements Engineering
۵	۴۰۷۲۴	۳	متدولوژی های ایجاد نرم افزار	Software Development Methodologies
۶	۴۰۷۴۵	۳	توصیف و واریسی برنامه ها	Program Specification and Verification
۷	۴۰۷۸۷	۳	الگوها در مهندسی نرم افزار	Patterns in Software Engineering
۸	۴۰۸۲۸	۳	آزمون نرم افزار	Software Testing
۹	۴۰۹۲۴	۳	مهندسی نرم افزار ۲	Software Engineering 2
۱۰	۴۰۷۳۷	۳	مفاهیم پیشرفته معماری نرم افزار	Advanced Topics in Software Architecture
۱۱	۴۰۹۲۵	۳	مفاهیم پیشرفته مهندسی نرم افزار	Advanced Topics in Software Engineering

دروس سمینار و پروژه

جدول ۵. فهرست دروس سمینار و پروژه

ردیف	شماره	تعداد واحد	نام درس (فارسی)	نام درس (لاتین)
۱	۴۰۹۰۰ (گروه ۱)	۲	سمینار کارشناسی ارشد	M.Sc. Seminar
۲	۴۰۶۶۰	۶	پایان نامه کارشناسی ارشد	M.Sc. Thesis

دروس زیرگرایش الگوریتم‌ها و محاسبات



گرایش: نرم‌افزار	مقطع: تحصیلات تکمیلی
واحد: ۳	نوع درس: زیرگرایش الگوریتم‌ها و محاسبات
پیش‌نیاز: -	هم‌نیاز: -

اهداف درس

این درس برای کسب یک دید سطح بالا از مدل‌ها و تکنیک‌هایی که برای مطالعه و تحلیل شبکه‌های اجتماعی و اقتصادی به کار برده می‌شوند ارائه شده است. شبکه‌های اجتماعی و اقتصادی چند مشخصه اصلی دارند:

- ۱- شبکه پیچیده (Complex Network) هستند و بسیاری از مشخصه‌های اصلی آن را دارند.
- ۲- در اثر تعامل عوامل باهوش و خودخواه ایجاد می‌شوند.
- ۳- تصادفی هستند و در بسیاری موارد نمی‌توان به صورت قطعی ویژگی‌های آن‌ها را مشخص کرد.

بنابراین مدل‌ها و تکنیک‌هایی که در این درس با آن‌ها سر و کار داریم از هر سه دسته بالا نشأت گرفته و در بسیاری موارد نیز با هم ترکیب می‌شوند. به همین منظور در این درس با مدل‌هایی سر و کار داریم که با استفاده از نظریه احتمال، نظریه بازی و نظریه گراف و ترکیبیات ساخته می‌شوند.

ریز مواد

۱. پیش‌زمینه‌ها و مبانی تحلیل شبکه (۵ جلسه)
 - a) مقدمه - مثال‌هایی از شبکه‌های اجتماعی و اقتصادی و کاربردهای آن در علوم مختلف
 - b) نمایش و اندازه‌گیری شبکه‌ها - بعضی مقدمات از تئوری گراف - مشخصات عمومی شبکه‌ها
 - c) مشاهدات تجربی در مورد شبکه‌های اجتماعی و اقتصادی
۲. مدل‌های شکل‌گیری شبکه‌ها (۵ جلسه)
 - a) گراف‌های تصادفی
 - b) مدل‌های تصادفی رشد شبکه
 - c) مدل‌های استراتژیک تشکیل شبکه
۳. انتشار در شبکه‌ها (۳ جلسه)
۴. یادگیری در شبکه‌ها (۳ جلسه)
۵. بازی‌های تحت شبکه (۳ جلسه)
۶. بازارهای شبکه‌ای (۳ جلسه)
۷. نظریه بازی‌های تعاونی و قوانین تخصیص منابع در شبکه‌ها (۳ جلسه)
۸. اندازه‌گیری تعاملات اجتماعی و اقتصادی (۳ جلسه)

ارزیابی

- آزمون (۶۰ درصد نمره)

- تمرین (۲۰ درصد نمره)

- سمینار (۲۰ درصد نمره)

مراجع اصلی

- M.O. Jackson, *Social and economic networks*. Princeton University Press, 2010.
- D. Easley and J. Kleinberg. *Networks, crowds, and markets: Reasoning about a highly connected world*. Cambridge University Press, 2010.



Parallel Processing

دانشکده مهندسی کامپیوتر

طراح درس: محمد قدسی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف اصلی از پردازش موازی انجام محاسبات یا طراحی مدارهای «تپنده» به کمک چندین پردازنده‌ی کوچک یا بزرگ است تا بتوان کارایی و تسریع بالایی را کسب کرد. دانش‌جویان در این درس، با مباحث نظری پردازش موازی و طراحی و تحلیل الگوریتم‌های موازی بر روی معماری‌های موازی مختلف و نیز مدل انتزاعی «پی‌رم» آشنا می‌شوند و نیز با برنامه‌نویسی موازی مبتنی بر پردازنده‌های گرافیکی GPU، مدل «نگاشت-کاهش» (Map-Reduce) و «چندریسه‌ای» آشنا می‌شوند و کار عملی انجام می‌دهند.

ریز مواد

۱. معرفی (۱ جلسه)

- نیاز به پردازش موازی [از کتاب‌های پرهامی و لایتون]
- انواع سیستم‌ها و پردازش موازی و واژه‌های علمی مورد استفاده
- موانع پردازش موازی
- پیچیدگی محاسبات موازی و رده‌ی NC

۲. آشنایی با الگوریتم‌های موازی [از کتاب‌های پرهامی و لایتون] (۲ جلسه)

- چند مسئله‌ی ساده (انقباض موازی، محاسبه‌ی پیشوندی موازی، مرتب‌سازی، داده‌پراکنی)
- چند معماری موازی (آرایه‌ی خطی، توری، ساختار درختی، گراف کامل)
- حل مسئله‌های فوق بر روی هر ساختار و تحلیل آن (کران پایین الگوریتم‌ها)
- آشنایی با «سیستم‌های تپنده» (systolic) و چند مسئله‌ی ساده (عملیات حسابی، محاسبات بیتی و کلمه‌ای، کانولوشن)

۳. مدل «پی‌رم» (PRAM) و الگوریتم‌های پایه‌ای [از کتاب پرهامی] (۲ جلسه)

- تعریف و فرضیات مدل پی‌رم
- حل چند مسئله و تحلیل (داده‌پراکنی، انقباض و پیشوند موازی، ترتیب عناصر در لیست، ضرب ماتریس‌ها)
- مسائل دیگر (انتخاب موازی، مرتب‌سازی، پوسته‌ی محدب نقاط)

۴. الگوریتم‌های موازی در سطح مدار [از کتاب لایتون] (۴ جلسه)

- قضیه‌ی retiming برای تبدیل مدارها و الگوریتم‌های هم‌گام به تپنده: استفاده از این قضیه در مسئله‌ی حل معادلات خطی، اجزای هم‌بند و بستر تراگذاری
- شبکه‌های مرتب‌ساز (Batcher، زوج-فرد)
- جست‌وجو و عملیات بر روی فرهنگ‌داده‌ای
- محاسبات پیشوندی، FFT

۵. الگوریتم‌های موازی مبتنی بر توری [از کتاب لایتون] (۴ جلسه)

- الگوریتم‌های مرتب‌سازی: Shearsort

- (b) الگوریتم‌های پردازش تصویر و هندسه‌ی محاسباتی
 - (c) مسیریابی بسته‌ها (packet routing)
 - (d) عملیات ماتریسی (حل معادلات خطی)
 - (e) الگوریتم‌های گراف
۶. معماری‌های با قطر کم (خانواده‌ی فوق مکعب) [از کتاب لایتون] (۴ جلسه)
- (a) ساختارهای «توری از درخت‌ها»، ابرمکعب، پروانه‌ای، برش تعویض
 - (b) جاده‌ی ساختارهای ساده در فوق مکعب
 - (c) الگوریتم‌های مختلف (مرتب‌سازی، ماتریسی، ...)
 - (d) الگوریتم‌های گراف
 - (e) مسیریابی و داده‌پراکنی
 - (f) الگوریتم‌های نرمال بر روی این ساختارها
 - (g) شبیه‌سازی الگوریتم‌های موازی از یک مدل به مدل دیگر
۷. مدل «نگاشت-کاهش» (Map-Reduce) و مبتنی بر GPU (۲ جلسه)
۸. برنامه‌سازی چندریشه‌ای [از کتاب CLRS] (۲ جلسه)

ارزیابی^۱

- آزمون: آزمون‌های میان‌نیمسال (۲۵ تا ۳۰ درصد نمره)، و پایان‌نیمسال (۳۰ درصد نمره)
- تمرین: ۵ یا ۶ تمرین نظری (۱۵ تا ۲۰ درصد نمره)؛ ۲ تمرین عملی با استفاده از GPU/CUDA (۱۵ درصد نمره)
- ارائه: ارائه مطالب اضافی (با نمره اضافی)
- پژوهش: به هر دانش‌جو مقاله‌ی عمیقی از مفاهیم درس داده می‌شود. او باید مقاله را بخواند و در انتهای درس در جلسه‌ای آن را ارائه نماید و نشان دهد که با جزئیات کامل آن‌را فهمیده است. اگر بر روی این موضوع کار پژوهشی جدی انجام دهید، امتیاز ویژه و اضافی خواهید گرفت (۱۵ درصد نمره).

مراجع اصلی

- F.T. Leighton, *Introduction to Parallel Algorithms and Architectures: Arrays, Trees, Hypercubes*. Morgan Kaufmann, 1992.
- B. Parhami, *Introduction to Parallel Processing: Algorithms and Architectures*. Plenum Press, 2000.
- D.B. Kirk and W.W. Hwu, *Programming Massively Parallel Processors: A Hands-on Approach*. Morgan Kaufmann, 2010 (supplementary text for multicore programming in CUDA).
- T. Cormen, C. Leiserson, R. Rivest, and C. Stein (CLRS), *Introduction to Algorithms*, 3rd ed. MIT Press, 2009.

^۱ درصدها بسته به کیفیت کار انجام شده تغییر می‌کنند.



Randomized Algorithms

طراح درس: محمدعلی آبام

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف از این درس، آشنایی دانش‌جویان با الگوریتم‌های تصادفی و تحلیل آن‌هاست.

ریز مواد

۱. احتمال و اثبات با روش احتمالاتی (۵ جلسه)
 - (a) تعاریف و قضایای پایه‌ای
 - (b) امید ریاضی و نامساوی مارکوف
 - (c) انحراف معیار و نامساوی چبیشف
 - (d) نامساوی چرنف
 - (e) مسئله توپ و ظرف‌ها و مسئله جمع‌آوری کوپن‌ها
۲. الگوریتم‌های تصادفی پایه‌ای (۲ جلسه)
 - (a) الگوریتم‌های لاس‌وگاس و مونتوکارلو: مسئله میانه تقریبی
 - (b) الگوریتم‌های مقایسه‌ای: انتخاب، مرتب‌سازی، پیچ و مهره
۳. جایگشت تصادفی و کاربردهای آن (۳ جلسه)
 - (a) مسئله‌ی استخدام
 - (b) مسئله‌ی مسیریابی در ابرمکعب
 - (c) الگوریتم‌های افزایشی: کوچک‌ترین دایره محیطی و Binary space partition
۴. کران پایین الگوریتم‌های تصادفی (۲ جلسه)
 - (a) قضیه Yao's minmax: بررسی مسئله پوشش نقاط با بازه‌های واحد در مدل آنلاین
 - (b) game tree evaluation
۵. ساختمان داده (۲ جلسه)
 - (a) Skip lists
 - (b) Treaps
۶. قدم‌زدن تصادفی (۲ جلسه)
 - (a) قدم‌زدن تصادفی در گراف‌های بدون جهت و کاربرد آن در مسئله‌های SAT₂ و SAT₃
 - (b) فرایند مارکوف (قدم‌زدن تصادفی در گراف‌های جهت‌دار با توزیع داده‌شده) و محاسبه احتمال بودن در یک گره در بی‌نهایت
۷. روش مونتوکارلو (۳ جلسه)
 - (a) معرفی روش مونتوکارلو: تخمین عدد π
 - (b) معرفی الگوریتم‌های تقریبی تصادفی کاملاً چندجمله‌ای: DNF counting problem
 - (c) نمونه‌برداری تقریبی و شمارش تقریبی: مسئله شمارش مجموعه‌های مستقل در گراف بصورت تقریبی

- (d) روش نمونه‌برداری مونتوکارلو با استفاده از فرایند مارکوف
۸. روش‌های جبری (۱ جلسه)
- (a) روش اثرانگشت: تساوی چندجمله‌ای‌ها و تطابق رشته‌ها
۹. الگوریتم‌های گراف (۲ جلسه)
- (a) برش کمینه
- (b) درخت پوشای کمینه
۱۰. آنترپی (۲ جلسه)
- (a) تابع آنترپی و خواص آن
- (b) آنترپی و ضرایب دوجمله‌ای
- (c) آنترپی: یک معیار تصادفی بودن

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۷۰ درصد نمره)
- ارائه و گزارش پژوهشی (۱۵ درصد نمره)
- تمرین تئوری (۱۵ درصد نمره)

مراجع اصلی

- R. Motwani and P. Raghavan, *Randomized Algorithms*. Cambridge University Press, 1995.
- J. Matoušek and J. Vondrák, *The Probabilistic Method*. Lecture Notes, Department of Applied Mathematics, Charles University, Prague, 2001.
- M. Mitzenmacher and E. Upfa, *Probability and Computing: Randomized Algorithms and Probabilistic Analysis*. Cambridge University Press, 2005.



Massive Data Algorithms

طراح درس: محمدعلی آبام

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

با افزایش چشمگیر حجم داده‌های ارزشمند و محدودیت پردازشی و ظرفیتی حافظه‌های سریع، نیاز به طراحی داده‌ساختارها و الگوریتم‌های ویژه برای داده‌های حجیم روزبه‌روز محسوس‌تر می‌شود. در این درس، مدل‌های رایج برای داده‌های حجیم مطرح و برای برخی مسائل پایه‌ای، الگوریتم‌های بهینه ارائه می‌شود

ریز مواد

۱. Cache-aware Models (۱۰ جلسه)

- (a) معرفی مدل
- (b) الگوریتم‌های مرتب‌سازی و کران پایین آن
- (c) داده‌ساختارهای جست‌وجو: Buffer tree و B-tree, Weighted B-tree, Persistent B-tree
- (d) داده‌ساختارهای هندسی: KD-tree و Range tree, Interval tree, Priority search tree
- (e) یک نمونه الگوریتم جاروب
- (f) الگوریتم‌های گراف: BFS, DFS, List ranking, Algorithms on trees و MST

۲. Cache-oblivious Models (۲ جلسه)

- (a) معرفی مدل
- (b) مرتب‌سازی
- (c) داده‌ساختارهای جست‌وجو
- (d) ضرب ماتریس‌ها

۳. Streaming Models (۱۰ جلسه)

- (a) معرفی مدل
- (b) پیدا کردن اعداد با تکرار زیاد
- (c) پیدا کردن تعداد اعداد متمایز
- (d) معرفی تکنیک sketching
- (e) بررسی مسائل پایه‌ای گراف
- (f) بررسی مسائل هندسی و معرفی core-sets
- (g) خوشه‌بندی
- (h) نظریه ارتباطات و حد پایین

۴. Sublinear Algorithms (۲ جلسه)

- (a) معرفی مدل
- (b) محاسبه میانگین درجه‌ی گراف

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۷۰ درصد نمره)
- ارائه و گزارش پژوهشی (۱۵ درصد نمره)
- تمرین تئوری (۱۵ درصد نمره)

مراجع اصلی

- L. Arge, *External memory geometric data structures*. Lecture notes, 2005.
- A. Chakrabarti and D. College, *Data stream algorithms*. Lecture notes, 2011.
- A. Czumaj and C. Sohler, *Sublinear-time algorithms*. Lecture notes, 2010.
- E. Demaine, *Cache oblivious algorithms and data structures*. Lecture notes, 2002.
- U. Meyer, P. Sanders, and J. Sibeyn, *Algorithms for memory hierarchies*. Lecture notes, 2003.
- N. Zeh, *I/O efficient graph algorithms*. Lecture notes, 2002.



Computational Geometry

دانشکده مهندسی کامپیوتر

طراح درس: حمید ضرابی زاده

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش الگوریتم ها و محاسبات	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

هدف از این درس، آشنایی دانش جویان با داده ساختارها و الگوریتم های کارا برای حل مسائل هندسی است. موضوعات ارائه شده در این درس در سایر حوزه های علوم کامپیوتر از جمله گرافیک کامپیوتری، بینایی ماشین، روباتیک، پایگاه های داده ای، بهینه سازی، طراحی مدارهای مجتمع، مصورسازی داده ها و سیستم های اطلاعات جغرافیایی مورد استفاده قرار می گیرند.

ریز مواد

۱. مقدمه (۲ جلسه)
 - (a) معرفی درس، نمونه مسائل هندسی
۲. پوسته ی محدب (۴ جلسه)
 - (a) محاسبه ی پوسته ی محدب در صفحه، عملیات پایه ی هندسی
 - (b) چند روش برای محاسبه ی پوسته ی محدب در صفحه
 - (c) اثبات کران پایین، قضیه ی بن-آر
 - (d) الگوریتم های حساس به خروجی، دو الگوریتم بهینه از چن
۳. دوگان هندسی (۱ جلسه)
 - (a) دوگان نقاط، پوش های بالایی و پایینی، کاربردها، دوگان در فضای سه بعدی
۴. پوسته ی محدب در فضای سه بعدی (۲ جلسه)
 - (a) پیچیدگی ترکیباتی، نحوه ی نمایش، الگوریتم کادوپچی
 - (b) الگوریتم تصادفی کلارکسون-شور، پوسته ی محدب در فضاهای بالاتر
۵. تقاطع و چینش خطوط (۲ جلسه)
 - (a) ساخت چینش خطوط، الگوریتم افزایشی، قضیه ی قلمرو یک خط
 - (b) تقاطع پاره خط ها، الگوریتم جاروب صفحه، الگوریتم تقسیم و حل
۶. نمودار ورونوی و مثلث بندی دلونی (۴ جلسه)
 - (a) تعریف نمودار ورونوی، ویژگی ها و قضا یا
 - (b) مثلث بندی دلونی و خواص آن، الگوریتم فورچيون
 - (c) ارتباط با پوسته ی محدب، الگوریتم تصادفی ساخت
 - (d) کاربردهای نمودار ورونوی و مثلث بندی دلونی، نمودار ورونوی مرتبه ی بالاتر
۷. برنامه ریزی خطی (۴ جلسه)
 - (a) تعریف برنامه ریزی خطی، کاربردهای هندسی در فضای پایین
 - (b) الگوریتم هرس و جست و جوی مگیدو، الگوریتم تصادفی-افزایشی سایدل

- (c) الگوریتم نمونه‌برداری تصادفی کلارکسون
 - (d) مسائل شبیه به برنامه‌ریزی خطی، کوچک‌ترین دایره‌ی محیطی
۸. مکان‌یابی نقاط (۲ جلسه)

- (a) روش تقسیم و حل، نقشه‌ی دوزنقه‌ای، الگوریتم افزایشی تصادفی
 - (b) الگوریتم هرس و جست‌وجوی کرکپاتریک
۹. مثلث‌بندی چندضلعی (۲ جلسه)

- (a) روش دوزنقه‌بندی، الگوریتم تصادفی سایدل
 - (b) کاربردهای مثلث‌بندی، مسئله‌ی گالری هنر
۱۰. جست‌وجوی بازه‌ای (۲ جلسه)

- (a) بازه‌های متعامد: درخت کی‌دی، درخت بازه، آبشار کسری
- (b) پنجره‌بندی، درخت جست‌وجوی اولویت، درخت پاره‌خط

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۵۵ درصد نمره)
- تمرین: سه تمرین نظری (۲۵ درصد نمره)
- پروژه: پروژه‌ی پژوهشی (۲۰ درصد نمره)

مرجع اصلی

- M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars, *Computational Geometry: Algorithms and Applications*, 3rd ed. Springer-Verlag, 2008.

مرجع کمکی

- J. O'Rourke, *Computational Geometry in C*, 2nd ed. Cambridge University Press, 1998.



Advanced Algorithms

دانشکده مهندسی کامپیوتر

طراح درس: محمد قدسی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

این درس برای دانش‌جویان کارشناسی ارشد و دکتری ارایه می‌شود و هدف آن پرداختن به موضوع‌های مهمی از الگوریتم‌هاست که قاعداً یک دانشجوی کارشناسی ارشد کامپیوتر باید بر آن‌ها مسلط باشد.

ریز مواد

۱. مقدمات، تحلیل سرشکنی (Amortized) (۱ جلسه)
۲. تکمیل مباحث قبل: درهم سازی کامل و سراسری (۲ جلسه)
۳. داده ساختارهای پیشرفته (هرم فیبوناچی - FFT) (۳ جلسه)
۴. مسایل ان پی-کامل (NP-Complete) (۴ جلسه)
 - a. مقدمات
 - b. نظریه‌ی ان پی-کامل
 - c. رابطه با مسایل ان پی
 - d. قضیه‌ی کوک
 - e. مسایل اصلی (3-Sat, 3D-matching, Vertex-Cover, Clique, دور همیلتونی، افراز)
 - f. روش‌های اثبات ان پی-کامل بودن یک مسئله
 - g. استفاده از این موضوع برای تحلیل الگوریتم‌ها
۵. الگوریتم‌های شبکه و گراف (۴ جلسه)
 - a. شبکه‌ی شار (روش Ford-Fulkerson، الگوریتم‌های Preflow-Push و Relabel-to-front)
 - b. گونه‌های متفاوت مسئله
 - c. کاربردهای مختلف
 - d. مسئله‌های تطابق (Matching)، گمارش (Assignment)، حمل‌ونقل (Transportation) و ازدواج پایدار
۶. الگوریتم‌های برنامه‌ریزی خطی (Linear Programming) (۴ جلسه)
 - a. نمونه‌های مسئله‌ها
 - b. فرم رسمی
 - c. حالت‌های خاص (محدودیت تفاضل و ابعاد ثابت)
 - d. فرم استاندارد
 - e. فرم Slack
 - f. الگوریتم Simplex (اثبات درستی و تحلیل)
 - g. دوگانی در الگوریتم‌های برنامه‌ریزی خطی

۷. الگوریتم‌های موازی و چندریسه‌ای (Multithreaded Algorithms) (۳ جلسه)
۸. مطالب اضافی (در صورت وجود وقت): شبکه‌های مرتب‌ساز (Sorting Networks) و تطابق رشته‌ها، الگوریتم KMP (۲ جلسه)

ارزیابی^۱

- آزمون: آزمون‌های میان‌نیمسال (۲۰ تا ۲۵ درصد نمره)، و پایان‌نیمسال (۳۰ تا ۳۵ درصد نمره)
- تمرین: حدود ۵ تمرین نظری (۱۵ تا ۲۰ درصد نمره)
- ارائه: ارائه‌ی سر کلاس بخش تکمیلی از درس با موافقت استاد (۵ درصد نمره)
- مطالعه یا پژوهش: هر دانش‌جو باید (به‌انتخاب نهایی استاد) در انجام یکی از کارهای زیر مشارکت کند (۱۵ تا ۲۰ درصد نمره):
 ۱. مطالعه و فهم یک مقاله تعیین شده و آرایه‌ی آن در کلاس و تهیه‌ی گزارش فارسی از آن؛
 ۲. انجام پژوهش بر روی موضوعی خارج از درس که توسط مدرس تعیین می‌شود، و حاصل آن گونه‌ی اولیه‌ی یک مقاله است.

مراجع

بخش‌هایی از کتاب‌های زیر به عنوان مراجع درس استفاده می‌شوند:

- T. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*, 3rd ed. MIT Press, 2009.
- محمد قدسی، داده‌ساختارها و مبانی الگوریتم‌ها. انتشارات فاطمی، شهریور ۱۳۸۸، چاپ پنجم ۱۳۹۴.

^۱ درصدها بسته به کیفیت کار انجام شده تغییر می‌کنند.



نظریه پیچیدگی (۴۰۷۷۵)

Theory of Complexity

طراح درس: محمد ایزدی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش الگوریتم ها و محاسبات	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

هدف از این درس ارائه مدل های پایه برای پیچیدگی محاسبه، مروری بر دستاوردهای اصلی نظریه محاسبه، آشنایی با قضایای اساسی این حوزه، شناخت کلاس های پیچیدگی زمانی و فضایی اصلی و همچنین مروری بر به کارگیری این نظریه در شاخه های جدیدتر نظریه محاسبات مانند محاسبات موازی، محاسبات تصادفی، محاسبات کوانتومی و روش های رمزنگاری است.

ریز مواد

- مروری بر نظریه ماشین های تورینگ، ماشین های تورینگ چند نواری و غیرقطعی (Nondeterministic)، تز تورینگ-چرچ، مسایل و زبان های بازگشتی و به طور بازگشتی شمارا. تعریف مفاهیم زمان اجرا و فضای مصرفی یک الگوریتم. (۲ جلسه)
- مروری بر مسایل تصمیم ناپذیر، مساله توقف و انواع آن، قضیه رایس، روش قطری سازی (Diagonalization). (۲ جلسه)
- مروری بر منطق گزاره ها و منطق مرتبه اول، مدل های حساب، قضایای صحت و تمامیت نظام استنتاجی منطق مرتبه اول، قضیه تصمیم ناپذیری منطق مرتبه اول، قضایای ناتمامیت گدل. (۴ جلسه)
- تعریف کلاس های پیچیدگی زمانی و فضایی در حالت کلی و قضایای اساسی ارتباط آنها. قضیه سلسله مراتبی بودن (Hierarchy) و قضیه Gap در حوزه پیچیدگی زمانی و فضایی، مروری بر کلاس های زمانی P ، NP ، EXP و $NEXP$ و کلاس های مکمل آنها. مروری بر کلاس های فضایی L ، NL ، $PSPACE$ و $NPSpace$ و کلاس های مکمل آنها و ارتباط آنها با کلاس های زمانی. (۳ جلسه)
- تعریف تقلیل (Reduction) و مسایلی که برای یک کلاس C -تمام (C-Complete) هستند. بررسی کلاس های مسایل P -Complete و NP -Complete. قضیه کوک-لورین و مباحث مرتبط با رابطه کلاس P و NP . (۲ جلسه)
- مروری بر برخی مسایل معروف NP -Complete. (۲ جلسه)
- کلاس $coNP$ و مسایل توابع. کلاس $PSPACE$ -Complete و مسایل مهم در آن. (۱ جلسه)
- کلاس های پیچیدگی الگوریتم های تصادفی، کلاس های RP ، $CoRP$ ، ZPP و BPP ، پیچیدگی مداری (Circuit Complexity). (۳ جلسه)
- الگوریتم های موازی، ساختار درونی کلاس P . (۲ جلسه)
- کلاس های پیچیدگی الگوریتم های تقریبی، حدود تقریب پذیری و تقریب ناپذیری. (۲ جلسه)
- رابطه نظریه های پیچیدگی و رمزنگاری. اثبات تعاملی (Interactive Proof). (۲ جلسه)
- مباحث ویژه مانند نظریه پیچیدگی در حضور ماشین های تورینگ پیشگو (Oracle TM)، ماشین تورینگ تناوبی (Alternating TM)، نظریه پیچیدگی محاسبات کوانتومی. (۳ جلسه)

ارزیابی

- آزمون: آزمون های میان نیمسال (۲۵ درصد نمره) و پایان نیمسال (۴۰ درصد نمره)
- تمرین: چند سری تمرین بر اساس متون معرفی شده (۲۰ درصد نمره)

- گزارش پژوهشی: ارائه مقاله‌ای در موضوعات مرتبط که حداقل مستلزم مطالعه پنج مقاله به‌روز باشد (۱۵ درصد نمره).
- ارائه: ارائه یک موضوع منتخب در کلاس (± 10 درصد نمره)

مراجع اصلی

- C.H. Papadimitriou. *Computational Complexity*. Addison-Wesley, 1994.
- S. Arora and B. Barak, *Computational Complexity: A Modern Approach*. Cambridge University Press, 2009.
- D.Z. Du and K.I. Ko, *Theory of Computational Complexity*. Wiley, 2000.
- I. Wegener. *Complexity Theory: Exploring the Limits of Efficient Algorithms*. Springer-Verlag, 2005.



Combinatorial Optimization

طراح درس: محمد قدسی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف این درس آشنایی دانشجو با بهینه‌سازی ترکیبیاتی از طریق بررسی مسائل بهینه‌سازی و الگوریتم‌های خاص این حوزه از جمله الگوریتم‌های Simplex و Prime-Dual، دوگانی و برنامه‌ریزی خطی صحیح است.

ریز مواد

۱. مسئله‌های بهینه‌سازی
معرفی، همسایگی، بهینه‌سازی محلی و سراسری، مجموعه و توابع محدب، مسئله‌های برنامه‌ریزی محدب
۲. الگوریتم Simplex
فرم‌های مسئله‌ی برنامه‌ریزی خطی، راه‌حل امکان‌پذیر، هندسه‌ی برنامه‌ریزی خطی، الگوریتم و پیچیدگی سیمپلکس، جنبه‌های هندسی انتخاب محور
۳. دوگانی (Duality)
دوگان برنامه‌ریزی خطی در حالت کلی، Slackness، مسئله‌ی کوتاه‌ترین مسیر و دوگان آن، مسئله‌ی شبکه‌ی شاره، دوگان الگوریتم سیمپلکس، جنبه‌های محاسباتی الگوریتم سیمپلکس
۴. الگوریتم Primal-Dual
الگوریتم و کاربرد آن در شبکه‌ی شاره و مسئله‌های کوتاه‌ترین مسیر دایکسترا، الگوریتم‌های کارا برای مسئله‌های شبکه‌ی شاره، تطابق (عادی و وزن‌دار)، درخت‌های پوشای کمینه
۵. برنامه‌ریزی خطی صحیح
NP-Complete بودن مسئله و راه‌های تقریبی، استفاده از بهینه‌سازی در حل تقریبی مسئله‌های NP-Hard

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال
- تمرین
- گزارش پژوهشی

مرجع اصلی

- C.H. Papadimitriou and K. Steiglitz, *Combinatorial Optimization: Algorithms and Complexity*. Dover Publications, 1998.



Approximation Algorithms

طراح درس: حمید ضرابی زاده

گرایش: نرم افزار	مقطع: تحصیلات تکمیلی
واحد: ۳	نوع درس: زیرگرایش الگوریتم‌ها و محاسبات
پیش نیاز: -	هم نیاز: -

اهداف درس

بسیاری از مسائل بهینه‌سازی در ریاضیات، علوم کامپیوتر و مهندسی ان‌پی-سخت هستند و بنابراین به دست آوردن جواب‌های بهینه برای این دسته از مسائل در زمان چندجمله‌ای با فرض $P \neq NP$ امکان‌پذیر نیست. الگوریتم‌های تقریبی امکان دستیابی به جواب‌هایی نزدیک به جواب بهینه با ضریب تقریب قابل اثبات را برای این دسته از مسائل فراهم می‌آورند. هدف از این درس، آشنایی با مفاهیم و تکنیک‌های متداول در طراحی الگوریتم‌های تقریبی حول محور مسائل بنیادی در بهینه‌سازی ترکیبیاتی، و نیز آشنایی با روش‌های اثبات سختی تقریب برای برخی از این مسائل است.

ریز مواد

۱. مقدمات (۲ جلسه)
 - (a) مسائل ان‌پی-بهینه‌سازی
 - (b) الگوریتم‌های تقریبی
 - (c) رده‌های تقریب‌پذیری
۲. روش‌های ترکیبیاتی (۶ جلسه)
 - (a) الگوریتم‌های حریمانه
 - (b) جست‌وجوی محلی
 - (c) تکنیک لایه‌بندی
 - (d) هرس پارامتری
 - (e) برنامه‌ریزی پویا
 - (f) گرد کردن
 - (g) روش‌های توری
۳. روش‌های مبتنی بر برنامه‌ریزی خطی (۸ جلسه)
 - (a) گرد کردن قطعی
 - (b) گرد کردن تصادفی
 - (c) روش بیضوی
 - (d) روش اولیه-دوگان
 - (e) روش برازش دوگان
 - (f) برنامه‌ریزی نیمه‌معین
 - (g) برنامه‌ریزی برداری

۴. مسائل بهینه‌سازی (۸ جلسه)

- (a) مسائل پوششی: پوشش رأسی، پوشش مجموعه‌ای
- (b) مسائل شبکه‌ای: درخت‌های اشتاینر، درخت اشتاینر جمع‌کننده جایزه
- (c) مسائل گشت: فروشنده دوره‌گرد، فروشنده دوره‌گرد اقلیدسی
- (d) مسائل برش: برش بیشینه، k -برش، برش چندگانه
- (e) مسائل عددی: کوله‌پشتی، بسته‌بندی
- (f) مسائل هندسی: مجموعه‌ی مستقل از مربع‌های واحد
- (g) مسائل صدق‌پذیری: k -صدق‌پذیری بیشینه
- (h) مسائل خوشه‌بندی: k -مرکز، مکان‌یابی تسهیلات
- (i) مسائل زمان‌بندی: زمان‌بندی کارها با پردازنده‌های موازی

۵. سختی تقریب (۲ جلسه)

- (a) اثبات‌های اولیه
- (b) کاهش با ایجاد فاصله
- (c) کاهش با حفظ درجه‌ی تقریب
- (d) حدس بازی‌های یکتا
- (e) قضیه‌ی PCP (در صورت فرصت)

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۵۵ درصد نمره)
- تمرین: دو یا سه تمرین نظری (۲۵ درصد نمره)
- پروژه: پروژه‌ی پژوهشی (۲۰ درصد نمره)

مراجع اصلی

- V. Vazirani, *Approximation Algorithms*. Springer-Verlag, 2001.
- D. Williamson and D. Shmoys, *The Design of Approximation Algorithms*. Cambridge University Press, 2011.



Algorithmic Game Theory

طراحان درس: محمدامین فضلی، محمد ایزدی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش الگوریتم‌ها و محاسبات	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

نظریه بازی‌ها کاربردهای وسیعی در بسیاری از حوزه‌ها دارد که مهم‌ترین آن‌ها حوزه‌های اقتصادی، کسب‌وکار، و علوم اجتماعی است. به طور کلی در نظریه بازی‌ها با سیستم‌هایی شامل عامل‌های هوشمند و خودخواه سروکار داریم که هر کدام از آن‌ها بنا به مصالح خویش وضعیت سیستم را تغییر می‌دهند. نظریه بازی‌ها ابزار تحلیل این گونه از سیستم‌ها را در اختیار ما قرار می‌دهد و کمک می‌کند تا بتوانیم آن‌ها را به شیوه‌ای درست و منطقی کنترل کنیم. با توجه به رشد بازارهای دیجیتال و گسترده‌تر شدن و بزرگ‌شدن سیستم‌های کاربردی چندعامله با عامل‌های هوشمند، نیاز به توسعه روش‌ها و الگوهای محاسباتی و الگوریتمی بیشتر و بیشتر شده است. در بسیاری از موارد بدون این روش‌ها امکان تحلیل و طراحی استراتژی‌های کارا وجود ندارد.

هدف از این درس آشنایی با این روش‌ها و الگوریتم‌هاست که با توجه به ماهیت آن، در اصل کاربرد برخی تئوری‌های اصیل علوم کامپیوتر در این حوزه است.

ریز مواد

- ۱- آشنایی با مقدمات نظریه بازی‌ها - یادآوری (۸ جلسه)
- ۲- محاسبه نقاط تعادل و مسائل مربوطه (۶ جلسه):
 - (a) بازی‌های صفرجمع دونفره (Zero-sum Games) و قضیه MinMax
 - (b) بازی‌های صفرجمع چندنفره
 - (c) قضیه نش (Nash Theorem)، لم اسپرنر (Sperner's Lemma) و قضیه بروور (Brouwer's Theorem)
 - (d) الگوریتم لمکه هاوسون (Lemke Howson Algorithm)
 - (e) مسائل جستجوی تام (Total Search Problems) و کلاس‌های پیچیدگی PPA، PPP، PPAD و PLS
 - (f) کلاس پیچیدگی مسائل یافتن نقاط تعادل نش
- ۳- منطق، اتوماتا و بازی‌های بی‌نهایت (۷ جلسه):
 - (a) گراف بازی (Game Graph) و شرایط برد
 - (b) شرایط برد در حالت غیرقطعی (شرایط بوخی، مولر، رابین و ...) و تبدیلات آن‌ها
 - (c) بازی‌های بی‌نهایت و تشخیص (Determinacy) و تشخیص بی‌حافظه (Memoryless Determinacy)
 - (d) شرایط برد منطقی (Logical Winning Conditions)
 - (e) اتوماتای درختی
 - (f) بازی‌های زوجیت (Parity Games) و بازی‌های نیمه بازپرداخت (Mean Payoff) و حل آن‌ها
 - (g) بازی‌های قابلیت رسیدن (Reachability Games)
 - (h) بازی‌های تکرارشونده (Repeated Games)
 - (i) فرآیند تصمیم‌گیری مارکف و بازی‌های تصادفی

(j) شبیه‌سازی، دوتشابهی (Bisimulation) و بازی‌های ارنفوشت-فریز (Ehrenfeucht-Fraïssé)

-۴ طراحی مکانیزم الگوریتمی (۷ جلسه):

- (a) آشنایی با مقدمات طراحی مکانیزم و لم مایرسون (Myerson's Lemma)
- (b) مثال‌های مختلف از جمله مزایده‌های کوله پشتی (Knapsack Auctions)
- (c) مزایده‌های بیشینه‌کننده سود (Revenue Maximizing Auctions) و قیمت رزرو شده (Reserved Price)
- (d) مزایده‌های نزدیک بهینه ساده (Near Optimal Auctions)، نامساوی پیامبر (Prophet Inequality) و قضیه بولو کلمپر (Bulow Klemperer Theorem)
- (e) طراحی مکانیزم چند پارامتره و مکانیزم‌های VCG
- (f) مزایده‌های ترکیبیاتی (Combinatorial Auctions) و مزایده طیف‌های بی‌سیم (Wireless Spectrum)
- (g) طراحی مکانیزم برای حالت‌های غیرخطی - حالت‌های بودجه محدود و مزایده‌های پرچی (Clinching Auctions) و مکانیزم‌های بدون پول

(h) بازارهای تطابق و تطابق‌های پایدار، بازارهای تبادل کلیه (Kidney Exchange Markets)

-۵ دینامیک بازی‌ها و مسائل یادگیری (۶ جلسه):

- (a) هزینه آشوب (Price of Anarchy) و هزینه ثبات (Price of Stability)
- (b) بازی‌های پتانسیلی (Potential Games)
- (c) بازی‌های نرم (Smooth Games) و هزینه آشوب مستحکم
- (d) تعادل‌های نش قوی (Strong Nash Equilibria)
- (e) دینامیک‌های بهترین پاسخ (Best Response Dynamics) و نقاط تعادل تقریبی
- (f) یادگیری، دینامیک‌های بازی ساختگی (Fictitious Play) و دینامیک‌های بدون حسرت (No-Regret Dynamics) و یادگیری تقویتی

ارزیابی

- آزمون (۶۰ درصد نمره)
- تمرین (۲۰ درصد نمره)
- سمینار (۲۰ درصد نمره)

مراجع اصلی

- T. Roughgarden. *Twenty Lectures on Algorithmic Game Theory*. Cambridge University Press, 2016.
- K.R. Apt and E. Grädel (Eds.), *Lectures in game theory for computer scientists*. Cambridge University Press, 2011.
- Y. Shoham and K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.
- N. Nisan, T. Roughgarden, E. Tardos, and V.V. Vazirani. *Algorithmic game theory*. Cambridge University Press, 2007.

دروس زیرگرایش سیستم‌ها



مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف از این درس، آشنائی دانشجویان با مفاهیم سیستم‌های عامل توزیع شده و عناوین پژوهشی مطرح در ادبیات موضوع در زمینه ارتباطات بین پردازه‌ای، همگامی، تکرار، و تحمل خطا می باشد.

ریز مواد

۱. معرفی اجزای سیستم‌های عامل، سیستم‌های توزیع شده، سیستم‌های عامل توزیع شده و دیگر مفاهیم مقدماتی (۲ جلسه)
۲. ارتباطات در سیستم‌های عامل توزیع شده (۴ جلسه)
 - (a) پروتکل‌های لایه‌ای
 - (b) فراخوانی از راه دور (RPC)
 - (c) فراخوانی از راه متد (RMC)
 - (d) ارتباطات پیام محور (Message-Oriented)
 - (e) ارتباطات نهر محور (Stream-Oriented)
۳. پردازها در سیستم‌های عامل توزیع شده (۲ جلسه)
 - (a) ریشه‌ها (Threads)
 - (b) کارخواه و کارساز (Client & Server)
 - (c) مهاجرت کد (Code Migration)
۴. نام در سیستم‌های عامل توزیع شده (۲ جلسه)
 - (a) ضرورت نام و عناصر نام
 - (b) ساختارهای مدیریت نام
۵. حالت در سیستم‌های عامل توزیع شده (۳ جلسه)
۶. ساعت و همگامی (۴ جلسه)
 - (a) ساعت فیزیکی
 - (b) ساعت منطقی
 - (c) بردار ساعت
۷. انتخاب رهبر (۲ جلسه)
۸. توافق در محیط توزیع شده (۲ جلسه)
۹. ممانعت دوجانبه توزیع شده (۳ جلسه)
 - (a) روش‌های نامهره بنیاد
 - (b) روش‌های مهره بنیاد

(c) روش‌های حسی ذهنی

۱۰. ترمیم (۲ جلسه)

(a) نقطه مقابله‌گیری و ترمیم

(b) همگامی در ترمیم

(c) ناهمگامی در ترمیم

۱۱. تحمل خطا با تاکید بر **Commitment** (۲ جلسه)

(a) 2-Phase Commitment

(b) 3-Phase Commitment

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۵ درصد نمره)
- تمرین: تمرین و برنامه‌نویسی درون‌هسته‌ای سیستم عامل (۲۰ درصد نمره)
- گزارش: مقاله مروری درس و ارائه در روز سمینار (۱۵ درصد نمره)

مراجع اصلی

- A.S. Tanenbaum and M. van Steen, *Distributed Systems*. Prentice-Hall, 2002.
- M. Singhal and N. Shivarati, *Advanced Concepts in Operating Systems*. McGraw-Hill, 1994.
- S. Mullender (Ed.), *Distributed Systems*, 2nd ed. ACM Press, 1993.

مراجع فرعی

- N. Lynch, *Distributed Algorithms*. Morgan Kaufmann, 1997.
- A.S. Tanenbaum, *Distributed Operating Systems*. Prentice-Hall, 1994.



Decision Support Systems

دانشکده مهندسی کامپیوتر

طراح درس: جعفر حبیبی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

سیستم‌های پشتیبانی تصمیم‌گیری واحد درسی است که طی آن فرایندهای تصمیم‌گیری و استفاده از ابزارهای کامپیوتری برای حل بهتر مسائل و غلبه بر پیچیدگی‌های موجود در آن‌ها معرفی می‌شود. در این درس دانشجویان با سیستم‌های پشتیبانی مدیریت (MSS)، پشتیبانی تصمیم (DSS) و مکانیزم‌های مرتبط با هوشمندی تجاری (BI) و توسعه و ساخت این سیستم‌ها آشنا می‌شوند. در این درس چهارچوبی تدوین شده که دانشجویان در آن با ساختار سیستم‌های اطلاعاتی و ابزارها و تکنیک‌هایی نظیر انبار داده‌ها در فرایندهای سیستم پشتیبانی تصمیم آشنا شده و در طی آن مهارت‌های لازم برای توسعه یک سیستم پشتیبانی تصمیم را فرا می‌گیرند.

ریز مواد

۱. مفاهیم پایه فرایند تصمیم‌گیری (۲ جلسه)
۲. نیاز به سیستم‌های پشتیبانی تصمیم (۱ جلسه)
۳. مفاهیم مطرح در هوشمندی تجاری (Business Intelligence) و رابطه آن با سیستم‌های پشتیبانی تصمیم (۲ جلسه)
۴. مدیران و تصمیم‌گیری، دورنمایی از سیستم‌های پشتیبانی مدیریت (MSS) شامل سیستم پشتیبانی تصمیم (DSS) (۱ جلسه)
۵. سیستم پشتیبانی تصمیم گروهی (G-DSS) (۲ جلسه)
۶. سیستم پشتیبانی اجرایی (EIS) (۱ جلسه)
۷. سیستم خبره (ES) (۱ جلسه)
۸. سیستم مدیریت دانش (KMS) (۳ جلسه)
۹. مفهوم سیستم، تصمیم‌گیری، و فازهای آن (۲ جلسه)
۱۰. دورنمایی از سیستم‌های پشتیبانی تصمیم (DSS) شامل زیر سیستم داده، زیرسیستم مدل و زیرسیستم واسط کاربر (۲ جلسه)
۱۱. مدیریت داده‌ها شامل مخزن داده‌ها، بازیابی، پایگاه داده‌ها در DSS، پردازش تحلیلی برخط (OLAP)، داده کاوی و نمایش (۱ جلسه)
۱۲. انبار داده‌ها و مفاهیم مرتبط با آن (۲ جلسه)
۱۳. مدل سازی و تحلیل شامل انواع مدل‌های استاتیک و پویا، مدل سازی به وسیله صفحه گسترده، درخت تصمیم و جدول تصمیم، مدل‌های بهینه، و مدل‌های ابتکاری (Heuristic)، شبیه سازی، مدل‌های چند بعدی (OLAP)، مدل سازی و شبیه سازی بصری (۲ جلسه)
۱۴. توسعه DSS شامل چرخه سنتی عمر نرم‌افزار، و متدولوژی‌های مختلف برای توسعه DSS (۲ جلسه)
۱۵. سیستم‌های پشتیبانی تصمیم‌گیری گروهی و تکنولوژی‌های پشتیبانی گروهی (۱ جلسه)
۱۶. سیستم‌های پشتیبانی اجرایی EIS شامل سیستم‌های اطلاعاتی اجرایی نقش مدیران و اطلاعات، مشخصه‌های EIS، مقایسه EIS و DSS، زنجیره تامین و پشتیبانی تصمیم (۱ جلسه)
۱۷. سیستم‌های خبره و هوش مصنوعی شامل مفهوم هوش مصنوعی و اجزای آن، مفهوم سیستم‌های خبره، ساختار سیستم‌های خبره، زمینه‌های سیستم‌های خبره، و انواع سیستم‌های خبره (۱ جلسه)
۱۸. پشتیبانی تصمیم‌گیری شبکه‌ای مثل اینترنت، اینترنت و اکسترانت (۱ جلسه)
۱۹. پیاده سازی DSS و استراتژی‌های آن (۱ جلسه)

۲۰. DSS های هوشمند و آثار اجتماعی و سازمانی سیستم‌های پشتیبانی مدیریت (MSS) (۱ جلسه)

۲۱. ابزارهای داده و داده‌کاوی (۱ جلسه)

۲۲. مدیریت ارتباطی با مشتری (CRM) و برنامه‌ریزی منابع سازمان (ERP) (۱ جلسه)

ارزیابی

- **آزمون:** این درس شامل یک آزمون میان‌نیمسال است که در اواسط نیمسال برگزار شده و شامل مطالب شش فصل ابتدایی است؛ آزمون پایان‌نیمسال شامل کلیه مطالب تدریس شده است (جمعاً ۵۰ درصد نمره).
- **تمرین‌های پژوهشی:** (۱۰ درصد نمره)
(۱) تمرین‌هایی که باید با استفاده از اینترنت و متن کتاب تجزیه و تحلیل شده و پاسخ داده شوند.
(۲) تمرین‌هایی که در ارتباط با استفاده از مفاهیم و ابزارهای موجود در دنیای واقعی است که تحت عنوان مطالعه موردی به دانشجویان داده می‌شود.
- **تمرین‌های عملی:** در این بخش دانشجویان موظف به اجرا و پیاده‌سازی پروژه‌های مرتبط با این درس و استفاده از ابزارهای متنوع متن-باز موجود در این زمینه هستند (۱۰ درصد نمره).
- **سمینار و ارائه کارهای پژوهشی:** در این درس دانشجویان به صورت گروهی و یا انفرادی باید کارهای پژوهشی خود را ارائه دهند و از سمینارهای مذکور ارزش‌یابی به عمل می‌آید (۳۰ درصد نمره).

مراجع اصلی

- E. Turban, R. Sharda, and D. Delen, *Decision Support Systems and Business Intelligent Systems*, 9th ed. Prentice-Hall, 2010.
- I. Linden, S. Liu, and C. Colot, *ICDSST Decision Support Systems VII*, 1st ed. Springer-Verlag, 2017.
- G.M. Marakas, *Decision Support in the 21st Century*, 2nd ed. Prentice-Hall, 2003.
- C. Todman, *Designing a Data Warehouse: Supporting Customer Relationship Management*. Prentice-Hall Professional Technical Reference, 2000.
- V. Kumar and W. Reinartz, *Customer Relationship Management*, 3rd ed. Springer-Verlag, 2018.
- S.A. Brown, *Customer Relationship Management: A Strategic Imperative in the World of e-Business*. Wiley, 1999.
- E. Berner, *Clinical Decision Support Systems: Theory and Practice*, 3rd ed. Springer-Verlag, 2016.



IT Enterprise Architecture

طراح درس: جعفر حبیبی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف اصلی این درس آشنایی با مفاهیم، روش‌های اجرایی، چارچوب‌ها، متدولوژی‌ها و ابزارهای طراحی معماری سازمانی و معماری اطلاعات، به ویژه تکنیک‌های برنامه‌ریزی و طراحی توزیع‌شدگی داده‌ها و اطلاعات در سازمان‌ها می‌باشد.

ریز مواد

۱. مفاهیم معماری سازمانی (۱ جلسه)
۲. جایگاه کنونی معماری سازمانی (۲ جلسه)
۳. فرایند تدوین معماری سازمانی (۲ جلسه)
۴. نقش برنامه‌ریزی راهبردی فناوری اطلاعات در معماری سازمانی (۱ جلسه)
۵. چارچوب‌های معماری سازمانی (Zachman, FEAF, DoDAF, ToGAF) (۵ جلسه)
۶. متدولوژی‌های معماری سازمانی (IE, EAP, ADM) (۶ جلسه)
۷. متدولوژی معماری سازمانی سرویس‌گرا (۱ جلسه)
۸. تجربیات موفق پیاده‌سازی و نگهداری معماری سازمانی (Best Practices) (۲ جلسه)
۹. مدل‌های مرجع کسب و کار، کاربرد، اطلاعات، امنیت و زیرساخت (۵ جلسه)
۱۰. مدیریت و انتقال معماری سازمانی (۱ جلسه)
۱۱. پیاده‌سازی و نگهداری معماری سازمانی (۱ جلسه)
۱۲. ارزیابی معماری سازمانی (۱ جلسه)
۱۳. مدل حاکمیت سرویس‌های فناوری اطلاعات مبتنی بر COBIT (۲ جلسه)
۱۴. مدل مدیریت خدمات فناوری اطلاعات مبتنی بر ITIL (۲ جلسه)

ارزیابی

- آزمون پایان نیمسال (۵۰ درصد نمره)
- تمرین (۱۵ درصد نمره)
- پروژه (۲۵ درصد نمره)
- سمینار (۱۰ درصد نمره)

- D. Minoli, *Enterprise Architecture A to Z: Frameworks, Business Process Modeling, SOA, and Infrastructure Technology*. Taylor & Francis, 2008.
- M. Lankhorst, *Enterprise Architecture at Work*, 4th ed. Springer-Verlag, 2017.
- *Federal Enterprise Architecture Framework*, Version 2. 2013.
- C. Perks and T. Beveridge, *Guide to Enterprise IT Architecture*. Springer-Verlag, 2002.
- S. Bente, U. Bombosch, and S. Langade, *Collaborative enterprise architecture: Enriching EA with lean, agile, and enterprise 2.0 practices*. Morgan Kaufmann, 2012.
- S.H. Spewak, *Enterprise Architecture Planning: Developing a Blueprint for Data, Applications, and Technology*. Wiley, 1993.
- J. Kunigk, L. George, P. Wilkinson, and I. Buss, *Hadoop in the Enterprise. Architecture: A Guide to Successful Integration*, 1st ed. O'Reilly Media, 2017.
- J. Schekkerman, *How to Survive in the Jungle of Enterprise Architecture Frameworks: Creating or Choosing an Enterprise Architecture Framework*. Trafford, 2006.



Computer Networks Modeling and Analysis

دانشکده مهندسی کامپیوتر

طراح درس: علی موقر رحیم آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم ها	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

هدف این درس معرفی تاریخچه و دلایل بوجود آمدن شبکه های کامپیوتری، مروری بر کاربردهای گسترده آن ها چه در گذشته، چه در حال و چه در آینده، مروری دقیق بر مفاهیم اصلی استفاده شده در این سیستم ها، و نهایتاً استفاده از روش های مدل سازی ریاضی مطرح در مدل سازی و تحلیل شبکه های کامپیوتری است.

انواع شبکه های کامپیوتری مورد مطالعه شامل اینترنت و وب (The Internet and Web)، شبکه های اقتضایی سیار (Mobile Ad-hoc Networks)، شبکه های حسگر بی سیم (Wireless Sensor Networks)، شبکه های همتا - به - همتا (Peer-to-Peer Networks)، شبکه های فراگستر (Overlay Networks)، محاسبات توری (Grid Computing)، محاسبات ابری (Cloud Computing) و محاسبات فراگیر (Pervasive Computing) می باشد.

ریز مواد

۱. مفاهیم اولیه (۴ جلسه)

تعریف شبکه های کامپیوتری، شبکه های کامپیوتری و سیستم های توزیع شده، تاریخچه شبکه های کامپیوتری، چگونگی تکامل شبکه های کامپیوتری، شبکه های محلی و انواع آنها، شبکه های گسترده و انواع آنها، شبکه جهانی اینترنت، سخت افزار شبکه، نرم افزار شبکه، مفهوم لایه بندی در شبکه ها، سرویس های اتصال گرا، سرویس های بدون اتصال، مفهوم راهگزینی در شبکه ها، راهگزینی مداری، راهگزینی بسته ای، راهگزینی مدار مجازی، دیتاگرام، معماری شبکه، مدل مرجع OSI، مدل مرجع TCP/IP، استانداردهای مطرح شبکه.

۲. ارتباطات و پروتکل های نقطه به نقطه (۱۰ جلسه)

لایه فیزیکی، کانال ها و مودم ها، قضیه نمونه برداری، مفهوم پهنای باند و تعریف دقیق آن، قضیه Shannon، انواع کانال ها، اصول عملکرد مودم ها و انواع آنها، جفت سیم تابیده، کابل هم محور، فیبر نوری، انتقال بی سیم، ماهواره های ارتباطی و انواع آنها، تاریخچه تلفن های سیار و نحوه عملکرد آنها، نسل های مختلف تلفن های سیار، انواع ادغام (Multiplexing) و عدم ادغام (Demultiplexing)، پروتکل های ARQ، انواع پروتکل های ARQ، مفهوم قالب بندی (Framing)، انواع قالب بندی، روش های مطرح کنترل خطا، روش های مطرح کنترل جریان، کدهای چندجمله ای، انواع پروتکل های پنجره لغزان، درستی یابی پروتکل های ارتباطی.

۳. مدل های تأخیر در شبکه های داده (۱۰ جلسه)

ادغام ترافیک در خطوط ارتباطی، مدل های صف و قضیه Little، سیستم صف $M/M/1$ ، سیستم های صف $M/M/m$ ، $M/M/\infty$ و $M/M/m/m$ ، سیستم صف $M/G/1$ ، سیستم صف $M/G/1$ با تعطیلات، سیستم های سرکشی (Polling)، مدل سازی و تحلیل انواع سیستم های سرکشی، مدل سازی و تحلیل صف های با اولویت، شبکه های خطوط ارتباطی، برگشت پذیری زمانی، قضیه Burk، اصل استقلال Kleinrock، شبکه های صف، قضیه Jackson.

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۵ درصد نمره)
- تمرین: انجام حداقل ۲ یا ۳ تمرین جهت آشنایی با حداقل دو شبیه‌ساز مطرح مدل‌سازی و تحلیل شبکه‌ها (۲۰ درصد نمره)
- ارائه: گردآوری یک یا دو مقاله تحقیقی و ارائه شفاهی آن‌ها (۱۵ درصد نمره)

مرجع اصلی

- D. Bertsekas and R. Gallager, *Data Networks*, 2nd ed. Prentice-Hall, 1992.

مراجع فرعی

- A. Tanenbaum, *Computer Networks*, 4th ed. Prentice-Hall, 2003.
- J.F. Kurose and K.W. Ross, *Computer Networking: A Top-down Approach Featuring the Internet*. Addison-Wesley, 2000.
- A. Kumar, D. Manjunath, and J. Kuri, *Communication Networking: An Analytical Approach*. Morgan Kaufmann, 2004.
- Recent papers in computer networking which have appeared in renowned national and international conferences and journals.

**Theory of Distributed Systems**

طراح درس: محمد ایزدی

مقترح: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف از ارائه این درس آشنایی دانشجویان با اصول نظری و مفاهیم اساسی توزیع شده‌گی به ویژه توصیف، طراحی و تحلیل الگوریتم‌های مورد نیاز برای حل مسایل مطرح در مدل‌های مختلف سیستم‌های توزیع شده شامل مدل‌های هم‌گام یا ناهم‌گام و مدل‌های ارتباطی مبتنی بر حافظه مشترک یا مبتنی بر تبادل پیغام است. برخی از مسایل اصلی که به بررسی و تحلیل الگوریتم‌های آن پرداخته می‌شود عبارتند از: مساله انتخاب رهبر در انواع مدل‌های توزیع شده‌گی، حل توزیع شده برخی مسایل پایه گراف مانند مسایل کوتاه‌ترین مسیر، درخت پوشای کمینه، مجموعه مستقل بیشینه و جستجوی عرضی گراف، انواع مساله توافق (مانند Agreement و Commitment)، مساله اجماع (Consensus)، انواع روشهای برآوردن نیازمندی انحصار متقابل (Mutual Exclusion). همچنین به مسایل مرتبط با زمانمندی سیستم‌ها مانند هم‌گام کردن ساعت‌ها (Clock Synchronization) به ویژه با فرض وجود شکست (Failure) یا اشکال (Fault) در سیستم توزیع شده می‌پردازد. علاوه بر مباحث الگوریتمی فوق، این درس همچنین به موضوع اثبات صوری درستی عملکرد الگوریتم‌ها و پروتکل‌های توزیع شده و به خصوص روش‌های اثبات آن که ویژگی‌های لازم مانند ویژگی‌های سرزندگی (Liveness)، انصاف (Fairness) و فارغ از بن‌بست بودن (Deadlock Freeness) را ارضا می‌کنند می‌پردازد. بنابراین به عنوان ابزاری برای این گونه اثبات‌ها، این درس به روش‌های صوری مدل‌سازی سیستم‌های توزیع شده نیز می‌پردازد. در نهایت، به عنوان مطالب به روز و موضوعات پژوهشی، می‌توان به کاربرد تحلیلی الگوریتم‌ها و روش‌های فرمال فوق در برخی سیستم‌های توزیع شده بر بسترهای ارتباطی جدید مانند سیستم‌های بی‌سیم به خصوص شبکه‌های موبایل و ad hoc پرداخت.

ریز مواد

۱. معرفی نظریه سیستم‌های توزیع شده، مدل‌ها و روش‌های تحلیل آنها، دسته بندی انواع مدل‌های مختلف سیستم‌های توزیع شده شامل مدل‌های هم‌گام یا ناهم‌گام و مدل‌های ارتباطی مبتنی بر حافظه مشترک یا مبتنی بر تبادل پیغام، مروری بر روش‌های مدل‌سازی ریاضی انواع سیستم‌های بالا (۲ جلسه)
۲. مدل‌سازی شبکه‌های هم‌گام، روش‌های اثبات ویژگی‌های مدل، ملاک‌های پیچیدگی الگوریتم‌ها و سیستم‌های توزیع شده (۱ جلسه)
۳. مدل‌سازی شبکه‌های ناهم‌گام، اتوماتای ورودی-خروجی (Lynch's I/O Automata)، روش‌های اثبات درستی و ویژگی‌های لازم پروتکل‌های ارتباطی توزیع شده (۱ جلسه)
۴. مدل‌سازی شبکه‌های ناهم‌گام مبتنی بر حافظه مشترک (Shared Memory) یا مبتنی بر ارتباط با تبادل پیغام (Message Passing) (۱ جلسه)
۵. الگوریتم‌های انتخاب رهبر در: (۳ جلسه)
 - (a) شبکه Ring هم‌گام.
 - (b) شبکه‌های هم‌گام عمومی.
 - (c) شبکه‌های ناهم‌گام با تبادل پیغام.
 - (d) شبکه‌های ناهم‌گام با حافظه مشترک.

۶. الگوریتم‌های توزیع شده پایه مبتنی بر گراف مانند جستجوی عرضی، کوتاهترین مسیر و درخت پوشای کمینه در: (۳ جلسه)
 - (a) شبکه‌های هم‌گام.
 - (b) شبکه‌های ناهم‌گام با تبادل پیغام.
 - (c) شبکه‌های ناهم‌گام با حافظه مشترک.
۷. الگوریتم‌های اجماع (Consensus) در: (۳ جلسه)
 - (a) شبکه‌های هم‌گام با فرض امکان شکست ارتباط (Communication Failure).
 - (b) شبکه‌های هم‌گام با فرض امکان شکست پردازش (Process Failure).
 - (c) شبکه‌های ناهم‌گام با فرض امکان شکست ارتباط (Communication Failure).
 - (d) شبکه‌های ناهم‌گام با فرض امکان شکست پردازش (Process Failure).
۸. مسایل و الگوریتم‌های انحصار متقابل در شبکه‌های ناهم‌گام مبتنی بر حافظه مشترک (۲ جلسه)
۹. مسایل و الگوریتم‌های انحصار متقابل در شبکه‌های ناهم‌گام مبتنی بر تبادل پیغام (۲ جلسه)
۱۰. سایر الگوریتم‌ها برای شبکه‌های ناهم‌گام مبتنی بر حافظه مشترک شامل الگوریتم‌های تخصیص منابع (Recourses Allocation) و الگوریتم‌های اجماع؛ اثباتی برای عدم امکان اجماع در شبکه‌های ناهم‌گام مبتنی بر حافظه مشترک که دارای پردازش مشکل‌دار (شبکه‌های Fault Prone) هستند (۲ جلسه)
۱۱. مفاهیم اشیای اتمی و مسایل مرتبط با تصویرهای لحظه‌ای اتمی (Atomic Snapshots) (۳ جلسه)
۱۲. مروری بر مسایل مطرح شده بالا برای سیستم‌های توزیع شده مبتنی بر زمان‌بندی (Timing) به خصوص مسایل انحصار متقابل در این گونه سیستم‌ها، مسایل هم‌گام‌سازی ساعت‌ها (Clock Synchronization) و مساله اجماع (۲ جلسه)
۱۳. معرفی برخی زمینه‌های پژوهشی منتخب) کاربرد تحلیلی الگوریتم‌ها و روش‌های فرمال فوق در سیستم‌های توزیع شده بر بسترهای ارتباطی جدید مانند سیستم‌های بی‌سیم به خصوص شبکه‌های موبایل و ad hoc؛ مختصری پیرامون حساب پای (π-Calculus)، حساب عامل‌های متحرک (Ambient Calculus) و مدل‌های مبتنی بر اکتور (Actor)، زنجیره‌های بلوکی (Block-Chain) و ارزهای دیجیتال، مسایل خود پایدارسازی (Self-Stabilization)، رابطه نظریه بازی و نظریه سیستم‌های توزیع شده (۵ جلسه)

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال (۲۵ درصد نمره) و پایان‌نیمسال (۴۰ درصد نمره)
- تمرین: چند سری تمرین بر اساس متون معرفی شده (۱۵ درصد نمره)
- گزارش پژوهشی: ارائه مقاله‌ای در موضوعات مرتبط که حداقل مستلزم مطالعه پنج مقاله به‌روز باشد (۲۰ درصد نمره)
- ارائه: ارائه یک موضوع منتخب در کلاس (± 10 درصد نمره)

مراجع اصلی

- N. Lynch, *Distributed Algorithms*. Morgan Kaufmann, 1996.
- H. Attiya and J. Welch, *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*, 2nd ed. Wiley, 2004.
- G. Tel, *Introduction to Distributed Algorithms*, 2nd ed. Cambridge University Press, 2000.



Reactive Systems Verification

طراح درس: علی موقر رحیم‌آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

سیستم‌های واکنشی (reactive)، سیستم‌هایی هستند که رفتار آن‌ها به گونه‌ای پیوسته و تنگاتنگ در ارتباط با محیط (environment) تعریف می‌شود. اینگونه سیستم‌ها معمولاً در کاربردهای حساس (critical applications) به کار می‌آیند. نمونه‌هایی از این کاربردها عبارتند از: سیستم‌های عامل (operating systems)، شبکه‌های کامپیوتری سریع (high-speed computer networks)، سیستم‌های همراه بی سیم (wireless mobile systems)، سیستم‌های سایبری-فیزیکی (cyber-physical systems)، اینترنت اشیاء (Internet of things)، هوانوردی (avionics)، کنترل فرایندهای صنعتی (industrial process control)، نیروگاه‌های اتمی (nuclear plants) و غیره. از خواص مهم این سیستم‌ها، همروندی (concurrency)، جنبه‌های بی‌درنگ (real-time)، و کارایی (performance)، اتکاپذیری (dependability) و امنیت (security) بالا است. سه خاصیت اخیر معمولاً مستلزم درستی رفتار اینگونه سیستم‌ها در شرایط مختلف است. از سویی دیگر، تعیین چنین درستی رفتاری به دلیل جنبه‌های همروندی و بی‌درنگی معمولاً بسیار پیچیده است. این درس درباره مفاهیم و روش‌های رسمی (formal methods)ی است که برای توصیف مشخصات (specification) و اعتبارسنجی (validation) سیستم‌های واکنشی به کار می‌آیند.

ریز مواد

۱. درستی‌یابی سیستم (۱ جلسه)
وارسی الگو، مشخصات وارسی الگو، نقاط قوت و ضعف.
۲. مدل‌های همروندی (۳ جلسه)
سیستم‌های گذار، مدل‌های متنی مبتنی بر متغیرهای مشترک، مسائل معنایی، روابط هم‌ارزی بین دستورات، مدل‌های متنی مبتنی بر آر سال پیام، مدل CSP، مدل CCS، شبکه‌های پتری، هم‌ارزی رفتاری، توازی همگام، NanoPromela، توازی همگام، مساله انفجار فضای حالت، تبعات ملزومات عدالت، عدالت ضعیف، عدالت قوی، دستورات هماهنگ سازی، دستورات ارتباطی، جنبه‌های معنایی عدالت، عدالت در شبکه‌های پتری، عدالت در سیستم‌های گذار.
۳. خواص زمان خطی (۴ جلسه)
بن بست، رفتار زمان خطی، خواص و تغییر ناپذیرهای ایمنی، خواص زندگی، انصاف.
۴. خواص منظم (۴ جلسه)
خودکارها در روی کلمات متناهی، وارسی الگوی خواص ایمنی منظم، خودکارها در روی کلمات نامتناهی، وارسی الگوی خواص w منظم.
۵. منطق زمانی خطی (۶ جلسه)
نحو و معنا، هم‌ارزی فرمول‌های LTL، وارسی الگوی مبتنی بر خودکارهای LTL.
۶. منطق درخت محاسباتی (۶ جلسه)
نحو و معنا، قابلیت توصیف CTL نسبت به LTL، وارسی الگوی CTL، انصاف در CTL، وارسی الگوی نمادین CTL، CTL*.
۷. هم‌ارزی و تجرید (۳ جلسه)

شبیه‌سازی دوسویه، شبیه‌سازی دوسویه و هم‌ارزی CTL^* ، الگوریتم‌های پیمانه‌سازی شبیه‌سازی دو سویه، روابط شبیه‌سازی، شبیه‌سازی دوسویه و هم‌ارزی CTL^* $\forall$

۸. کاهش ترتیب جزئی (۱ جلسه)

استقلال اعمال، رهیافت مجموعه‌ی زمان خطی Ample، رهیافت مجموعه‌ی زمان شاخه‌ای Ample.

۹. خودکارهای زمانی (۱ جلسه)

معنا، منطق درخت محاسباتی زمانی، واریسی الگوی TCTL.

۱۰. سیستم‌های احتمالی (۱ جلسه)

زنجیره‌های مارکوف، منطق درخت محاسباتی احتمالی، خواص زمان خطی، $PCTL^*$ و شبیه‌سازی دو سویه احتمالی، زنجیره‌های مارکوف و هزینه‌ها، فرایند‌های تصمیم‌گیری مارکوف.

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۵ درصد نمره)
- ارائه: گردآوری یک یا دو مقاله تحقیقی و ارائه شفاهی آنها (۳۵ درصد نمره)

مرجع اصلی

- C. Baier and J.P. Katoen, *Principles of Model Checking*. MIT Press, 2008.

مراجع فرعی

- E. Clarke, O. Grumberg, and D.A. Peled, *Model Checking*. MIT Press, 1999.
- M. Huth and M. Ryan, *Logic in Computer Science: Modelling and Reasoning about Systems*. Cambridge University Press, 2000.
- R. Alur and T.A. Henzinger, *Computer-Aided Verification*. Draft, 1999.
- Z. Manna and A. Pnueli, *The Temporal Logic of Reactive and Concurrent Systems: Specification*. Springer-Verlag, 1992.
- C. Hoare, *Communicating Sequential Processes*. Prentice-Hall, 1985.
- R. Milner, *Communication and Concurrency*. Prentice-Hall, 1989.



Semantic Web

طراح درس: مرتضی امینی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف اصلی فناوری معنایی افزایش هوشمندی ماشینی و کاهش سربار پردازشی عامل‌های انسانی است که به طور ویژه‌ای در کاربرد وب نمود پیدا می‌کند. در این درس دانشجویان با مفاهیم مرتبط با فناوری معنایی و به طور خاص کاربرد آن در وب که وب معنایی خوانده می‌شود، آشنا می‌شوند. در این راستا سطوح و لایه‌های مختلف معنایی و امکانات زبانی و منطقی موجود در لایه‌های مختلف و مسائل کاربردی و مهندسی مرتبط با لایه هستان‌شناسی (به عنوان مهم‌ترین سطح معنایی در این ساختار) معرفی می‌گردند. نحوه ایجاد هستان‌شناسی، هم‌ترازی و نگاشت هستان‌شناسی‌ها، و استفاده از آنها در موتورهای جستجوی معنایی و بعنوان بستری برای ترکیب وب سرویس‌ها از جمله مواردی هستند که در این درس مورد بحث قرار می‌گیرند. در بخش‌های مختلف این درس، دانشجویان با ابزارهای مختلف موجود آشنا شده و در قالب تمرین‌های عملی نیز استفاده از این ابزارها را تجربه می‌نمایند.

ریز مواد

۱. معرفی وب معنایی و لایه‌های کیک معنایی (۱ جلسه)
۲. گونه‌های تصمیم‌پذیر منطق کلاسیک به عنوان زیرساخت منطقی وب معنایی (۳ جلسه)
۳. معرفی منطق توصیفی (Description Logic) و رویه استنتاج (۳ جلسه)
۴. لایه ساختار: معرفی XML و XML-Schema (XSD) (۲ جلسه)
۵. تبدیل XML با XSLT (۱ جلسه)
۶. لایه فراداده: معرفی RDF و RDF-Schema (RDFS) (۲ جلسه)
۷. زبان‌های پرس و جو بر روی وب: XQuery, RDQL و SparQL (۲ جلسه)
۸. لایه هستان‌شناسی: مفهوم هستان‌شناسی و زبان OWL (۱ تا ۲ جلسه)
۹. مهندسی و ارزیابی هستان‌شناسی (۱ جلسه)
۱۰. هم‌ترازی هستان‌شناسی (Ontology Alignment) (۱ تا ۲ جلسه)
۱۱. حاشیه‌نویسی و غنی‌سازی هستان‌شناسی (۱ جلسه)
۱۲. لایه قواعد و منطق: معرفی RuleML و SWRL (۱ تا ۲ جلسه)
۱۳. وب سرویس معنایی (۱ جلسه)
۱۴. جستجو در وب معنایی و جستجوی معنایی (۲ جلسه)
۱۵. امنیت در وب معنایی (۲ جلسه)

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال (۳۰ درصد نمره) و پایان‌نیمسال (۳۰ درصد نمره)
- تمرین: تمرین‌های نظری و عملی (۲۵ درصد نمره)

- پروژه: پروژه تحقیقاتی (۱۵ درصد نمره)

مراجع اصلی

- H.P. Alesso and C.F. Smith, *Thinking on the Web: Berners-Lee, Gödel and Turing*. Wiley, 2006.
- F. Baader, D. Calvanese, D.L. McGuinness, D. Nardi, and P.F. Patel-Schneider (Eds.), *The Description Logic Handbook: Theory, Implementation and Applications*. Cambridge University Press, 2003.
- Related recent papers on Semantic Web and its applications
- Semantic web languages specifications reference website: <http://www.w3.org>



Data Mining

دانشکده مهندسی کامپیوتر

طراح درس: محمد امین فضلی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

هدف این درس، آشنا کردن دانشجویان کارشناسی ارشد و دکترا با مفاهیم و مسائل مطرح در کاوش حجم و سیعی از داده‌ها و کشف الگوهای موجود در آنهاست. در ابتدا مفاهیم پیش‌پردازش داده‌ها و انبار داده ارائه شده و سپس به مباحث مربوط به حوزه‌های اصلی در داده کاوی پرداخته می‌شود.

ریز مواد

۱. مقدمه - معرفی داده کاوی و مدل‌های اصلی در آن (۱ جلسه)
۲. روش‌های مربوط به پیش‌پردازش داده‌ها (۲ جلسه)
۳. معرفی انبار داده، مکعب داده‌ای و گزارش‌گیری تحلیلی (۲ جلسه)
۴. مدل نگاشت-کاهش (۳ جلسه)
 - (a) الگوریتم‌های نگاشت-کاهش
 - (b) نظریه پیچیدگی مدل نگاشت-کاهش
 - (c) آشنایی با پشته‌های فناوری مربوطه مانند اسپارک، هادوپ و ...
۵. کاوش مجموعه‌های عناصر در حجم بزرگ (۳ جلسه)
 - (a) پیدا کردن مجموعه عناصر مشابه و توابع درهم‌سازی حساس به محل
 - (b) پیدا کردن مجموعه‌های با تکرر بالا، الگوریتم A-Priori، الگوریتم PCY، الگوریتم SON و تویوونن
۶. کاوش جریان داده‌ها (۳ جلسه)
 - (a) نمونه‌برداری جریان داده‌ها
 - (b) پالایش جریان داده‌ها و فیلترهای بلوم
 - (c) الگوریتم‌های جریان داده
۷. تحلیل شبکه‌های داده‌ای (۴ جلسه)
 - (a) ویژگی‌های ساختاری شبکه‌ها
 - (b) رتبه صفحه و قدم‌زنی تصادفی
 - (c) الگوریتم‌های کاوش انجمن‌ها
 - (d) الگوریتم‌های شمارش موتیف‌ها در گراف‌ها
 - (e) الگوریتم‌های نمونه‌برداری در گراف‌ها
۸. الگوریتم‌های خوشه‌بندی داده (۴ جلسه)
 - (a) الگوریتم‌های سلسله‌مراتبی
 - (b) الگوریتم K-Means

- (c) الگوریتم CURE
- (d) خوشه‌بندی جریان‌های داده
- ۹. الگوریتم‌های کاهش ابعاد (۲ جلسه)

- (a) الگوریتم SVD و PCA
- (b) الگوریتم CUR

۱۰. الگوریتم‌های یادگیری ماشین در داده‌های بزرگ (۴ جلسه)

- (a) مفهوم یادگیری ماشین و الگوریتم‌های رگرسیون
- (b) شبکه‌های عصبی، معرفی شبکه‌های عصبی عمیق
- (c) الگوریتم SVM

۱۱. داده‌کاوی در صنعت (۲ جلسه)

- (a) سیستم‌های توصیه‌گر
- (b) تبلیغات در وب و مسئله Adwords
- (c) بازارهای تطابق‌یابی
- (d) مسئله قیمت‌گذاری
- (e) داده‌کاوی سلامت
- (f) و ...

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۰ درصد نمره)
- تمرین: سه تمرین پژوهشی، و یک تمرین عملی که در طول نیمسال تحویل داده می‌شوند (۲۵ درصد نمره).
- گزارش پژوهشی: موضوع پژوهش قبل از آزمون پایان‌نیمسال تعیین می‌شود. دانشجو کار پژوهش را با کمک استاد درس آغاز کرده و پس از انجام کار، نتیجه پژوهش را در قالب گزارش ارائه می‌دهد (۱۵ درصد نمره).

مراجع اصلی

- J. Leskovec, A. Rajaraman, and J.D. Ullman, *Mining of massive datasets*. Cambridge University Press, 2014.
- J. Han, J. Pei, and M. Kamber, *Data mining: Concepts and techniques*. Elsevier, 2011.

۶. مدل‌سازی کارایی مبتنی بر شبکه‌های پتری (Petri nets) (۱ جلسه)
شبکه‌های پتری کلاسیک، شبکه‌های پتری زمانی، شبکه‌های پتری تصادفی (stochastic Petri nets)، شبکه‌های پتری تصادفی تعمیم‌یافته، شبکه‌های فعالیت تصادفی (stochastic activity networks).
۷. مدل‌های شبکه‌های صف ضرب‌پذیر (product-form queueing networks) (۴ جلسه)
تبیین راه حل ضرب‌پذیر، توازن محلی و راه حل ضرب‌پذیر، فرایندهای مارکوف برگشت‌پذیر (reversible)، سیستم‌های صف شبه برگشت‌پذیر (quasi-reversible)، مدل‌های شبکه صف باز، شرایط پایداری، شبکه‌های ضرب‌پذیر تک زنجیره‌ای، شبکه‌های ضرب‌پذیر چند زنجیره‌ای، مدل‌های شبکه صف بسته، شبکه‌های تک زنجیره‌ای، الگوریتم‌های brute-force، شبکه‌های چند زنجیره‌ای، شبکه‌های صف چند رده‌ای (multiclass)، الگوریتم‌های مختلف برای شبکه‌های بسته، الگوریتم‌های دقیق، الگوریتم‌های تقریبی، کران‌های کارایی، شرایط حل ضرب‌پذیر.

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۵ درصد نمره)
- تمرین: انجام دو یا سه برنامه شبیه‌سازی کامپیوتری (۲۰ درصد نمره)
- ارائه: گردآوری یک یا دو مقاله تحقیقی و ارائه شفاهی آن‌ها (۱۵ درصد نمره)

مراجع اصلی

- K. Kant, *Introduction to Computer System Performance Evaluation*. McGraw-Hill, 1992.
- M. Harchol-Batler, *Performance Modeling and Design of Computer Systems*. Cambridge University Press, 2013.

مراجع فرعی

- B.R. Haverkort, *Performance of Computer Communication Systems*. Wiley, 1998.
- E.D. Lazowska, J. Zahorjan, G.S. Graham, and K.S. Sevcik, *Quantitative System Performance*. Prentice-Hall, 1984.
- F. Nain, *Basic Elements of Queueing Theory*. Lecture Notes, 1998.
- R. Jain, *The Art of Computer System Performance Analysis*. Wiley, 1991.
- J. Banks, J.S. Carson, B.L. Nelson, and D.M. Nicol, *Discrete-Event Simulation*, 4th ed. Prentice-Hall, 2004.
- G. Bolch, S. Greiner, H. de Meer, and K. Trivedi, *Queueing Networks and Markov Chains*, 2nd ed. Wiley, 2006.
- D.W. Stroock, *An Introduction to Markov Processes*. Springer-Verlag, 2005.



Database Design 2

طراح درس: سید محمد تقی روحانی رانکوهی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش سیستم‌ها	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

در این درس با مفاهیم پیشرفته طراحی پایگاه داده‌ها آشنا می‌شویم و در انتهای نیمسال انتظار می‌رود که دانشجو بر مفاهیمی چون تراکنش، همروندی و پروتکل‌های کنترل آن، ترمیم و ایمنی پایگاه داده‌ها، بهینه‌سازی و دیگر مواردی که در بخش ریزمواد به تفصیل بیان شده‌اند مسلط شده باشد.

ریز مواد

۱. تراکنش

تعریف، خواص، حالات، تکنیک نقطه نگهداشت، زیرسیستم مدیریت تراکنش‌ها

۲. مفاهیم تئوری توالی پذیری

طرح اجرای متوالی، طرح اجرای همروند، مشکلات توارد کنترل‌نشده، طرح توالی‌پذیر، طرح‌های معادل (نتیجه‌ای-تعارضی-نمایی)، انواع طرح‌های توالی‌پذیر (نتیجه‌ای-تعارضی-نمایی)، آزمون توالی‌پذیری، قضیه بنیادی در تئوری توالی‌پذیری، کاربرد تئوری توالی‌پذیر

۳. پروتکل‌های کنترل همروندی

پروتکل‌های قفل‌گذاری: دوگانی، چند اسلوبی، قفل‌گذاری دو مرحله‌ای (مبنایی-محافظه‌کار-شدید-جسور-دقیق)، پروتکل قفل‌گذاری روی چند واحد قفل‌شدنی، قفل‌گذاری قصدی، قفل‌گذاری درختی، قفل‌گذاری جنگلی، قفل‌گذاری چند نسخه‌ای، راه‌حل‌های مشکل بن‌بست (پیش‌گیری-اجتناب-کشف)، پروتکل‌های مبتنی بر زمان مهر، پروتکل‌های چند نسخه‌سازی، پروتکل تایید، پدیده شبح داده، قفل‌گذاری مسندی، کنترل همروندی در شاخص درختی، پارامترهای ارزیابی تکنیک‌های کنترل همروندی.

۴. ترمیم پایگاه داده

تعریف، انواع خرابی، مدیر ترمیم، روش‌های تخلیه حافظه نهان، امکانات ترمیم، ثبت با نوشتن پیش‌رس، انواع فایل ثبت، زدایش فایل ثبت، ایجاد نقطه واریسی، روش‌های ترمیم خرابی سیستمی (UNDO-NO REDO, REDO-NO UNDO, REDO-UNDO, NO UNDO-NO REDO)، ترمیم خرابی رسانه‌ای.

۵. ترمیم پایگاه داده‌ها

تعریف، عوامل نقض جامعیت، انواع محدودیت جامعیتی، روش‌های توصیف محدودیت جامعیتی، سیستم فعال، قاعده فعال، معماری سیستم فعال، مدل اجرا، مراحل اعمال محدودیت‌ها توسط سیستم فعال، مزایا و کاربردهای سیستم فعال، رهانا (تعریف-کاربرد-مزایا-مشکلات-کم‌داشت‌ها)

۶. ایمنی پایگاه داده‌ها

تعریف، خطرات، تحلیل خطرات، شیء ایمنی، تدابیر ایمنی غیر کامپیوتری، تدابیر کامپیوتری: شناسایی کاربر، احراز هویت، مجازشماری، روش‌های کنترل دستیابی، روش‌های طراحی سیستم اطلاعاتی ایمن.

۷. بهینه‌سازی پرسش

تعریف- مراحل کلی پردازش پرسش- تجزیه پرسش- پیاده‌سازی عملگرهای جبر رابطه‌ای، بهینه‌سازی پرسش: بازنویسی، ارزیابی عبارات، انتخاب طرح اجرا: روش مبتنی بر هزینه، روش یافتاری، روش‌های دیگر.

۸. پارامترهای شناخت سیستم مدیریت پایگاه داده‌ها

رده‌بندی سیستم، اجزای سیستم، کاتالوگ سیستم، پارامترهای مربوط به کارایی، پارامترهای مربوط به معماری، پارامترهای مربوط به زبان داده‌ای فرعی، محورهای اصلی مقایسه سیستم‌ها، محک‌زنی سیستم.

ارزیابی

- آزمون کتبی
- تمرین
- مطالعه بیشتر
- تحقیق و ارائه شفاهی آن
- پروژه

مراجع اصلی

- R. Elmasri and S.B. Navathe, *Fundamentals of database systems*, 4th ed. Addison-Wesley, 2003.
- A. Silberschatz, H.F. Korth, and S. Sudarshan, *Database Systems Concepts*. 3rd ed. McGraw-Hill, 1997.
- C.J. Date, *An Introduction to Database Systems*, Volume 2. Addison-Wesley, 1983.
- H. Garcia-Molina, J.D. Ullman, and J. Widom, *Database System Implementation*. Prentice-Hall, 2000.
- G. Weikum and G. Vossen, *Transactional Information Systems*. Morgan Kaufmann Publishers, 2002.
- P.A. Bernstein, V. Hadzilacos, and N. Goodman, *Concurrency Control and Recovery in Database Systems*. Addison-Wesley, 1987.
- T.M. Connolly and C.E. Begg, *Database Solutions: A step by step guide to building databases*. Addison-Wesley, 2003.
- D.A. Simovici and R.L. Tenney, *Relational Database Systems*. Academic Press, 1995.
- J. Gray and A. Reuter, *Transaction Processing: Concepts and Techniques*. Morgan Kaufmann, 1993.
- روحانی رانکوهی: سیستم مدیریت پایگاه داده‌ها: مفاهیم و تکنیک‌ها...
- بازخوانی درس "طراحی و پیاده‌سازی پایگاه داده‌ها (کارشناسی):"
 - منابع ۱ تا ۳ و ۷ تا ۹
 - روحانی رانکوهی، مفاهیم بنیادی پایگاه داده‌ها، از چاپ سوم به بعد.

دروس زیرگرایش مهندسی نرم افزار



مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

هدف اصلی از این درس آشنایی با مفاهیم معماری نرم افزار، جایگاه آن، فرایند تدوین معماری، مستندسازی و ارزیابی معماری است که شامل اهداف جزئی زیر می باشد:

- فهم تاثیر پیشران های معماری بر ساختار سیستم های نرم افزاری
- درک نقش فنی، سازمانی و تجاری معماری نرم افزار
- شناخت ساختارهای کلیدی معماری (سبک ها، تاکتیک ها و ...)
- فهم اصول صحیح مستندسازی و ارائه معماری
- درک تاثیر COTS در طراحی معماری
- آشنایی با مشخصه های کیفی و روش های ارزیابی معماری
- آگاهی از آینده معماری نرم افزار

ریز مواد

۱. تعاریف معماری نرم افزار، پیشران های معماری (۲ جلسه)
۲. چرخه حیات معماری (۱ جلسه)
۳. نیازمندی های وظیفه مندی و غیر وظیفه مندی (۳ جلسه)
۴. نقش معماری نرم افزار در دستیابی به ویژگی های کیفی نرم افزار (۱ جلسه)
۵. جایگاه معماری نرم افزار در فرایند توسعه محصولات نرم افزاری (۱ جلسه)
۶. ساختارها و منظرهای معماری (۲ جلسه)
۷. تکنیک ها و متدهای طراحی معماری (۱ جلسه)
۸. تاکتیک ها، الگوها و سبک های معماری (۲ جلسه)
۹. طراحی معماری و روش های دستیابی به خصوصیات کیفی (۵ جلسه)
۱۰. مستندسازی معماری و زبان های توصیف معماری (۱ جلسه)
۱۱. روش های ارزیابی معماری نرم افزار (۳ جلسه)
۱۲. ارزیابی معماری نرم افزار (۱ جلسه)
۱۳. معماری نرم افزار خاص-دامنه (۱ جلسه)
۱۴. معماری خط تولید نرم افزار (۲ جلسه)
۱۵. معماری سرویس گرا (۱ جلسه)
۱۶. توسعه بر پایه معماری مولفه محور (۱ جلسه)
۱۷. معماری محاسبات ابری (۱ جلسه)

۱۸. مباحث نوین در معماری نرم افزار (۳ جلسه)

۱۹. آینده معماری نرم افزار (۱ جلسه)

ارزیابی

- آزمون پایان نیمسال (۵۰ درصد نمره)
- تمرین (۱۵ درصد نمره)
- پروژه (۲۵ درصد نمره)
- سمینار (۱۰ درصد نمره)

مراجع اصلی

- L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 3rd ed. Addison-Wesley, 2013.
- L. Bass, P. Clements, and R. Kazman, *Software Architecture in Practice*, 2nd ed. Addison-Wesley, 2003.
- P. Clements, et al. *Documenting Software Architectures: Views and Beyond*. Addison-Wesley, 2003.
- A.J. Lattanze, *Architecting Software Intensive Systems: A Practitioner's Guide*. Taylor and Francis/Auerbach, 2008.
- M. Shaw and D. Garlan, *Software Architecture: Perspectives on an Emerging Discipline*. Prentice Hall, 1996.



Software Evolution

طراح درس: عباس حیدرنوری

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

تولیدکنندگان نرم افزار اغلب به جای تولید سیستم های نرم افزاری جدید، حجم زیادی از منابع خود را صرف مراقبت و نگهداری از سیستم های موجود می کنند. این مراقبت و نگهداری اغلب شامل بهبود طراحی و پیاده سازی سیستم، پیدا کردن و رفع خطاها، و افزودن امکانات جدید به نرم افزار می باشد. هدف از فرآیند "تکامل نرم افزار"، حصول اطمینان از ادامه کار موفقیت آمیز یک سیستم نرم افزاری بعد از مرحله تولید آن می باشد. ولیکن، این فرآیند، به خودی خود، یک فرآیند پیچیده و زمان بر می باشد به گونه ای که روشها و ابزارهای گوناگونی برای این منظور طراحی و پیاده سازی گردیده است. هدف این درس، آشنا کردن دانشجویان کارشناسی ارشد و دکترا با مفاهیم "تکامل نرم افزار"، و آخرین پژوهشها و ابزارهای انجام شده در این زمینه می باشد.

ریز مواد

۱. بخش اول: مفاهیم مقدماتی تکامل نرم افزار
 - (a) آشنایی با تکامل نرم افزار و مفاهیم اولیه (۱ جلسه)
۲. بخش دوم: مهندسی معکوس سیستم های نرم افزاری
 - (a) مصورسازی سیستم های نرم افزاری (Software Visualization) (۱ جلسه)
 - (b) تحلیل و شناسایی کلونی های نرم افزاری (Clone Detection and Analysis) (۲ جلسه)
 - (c) جستجو در کد برنامه به منظور شناسایی نحوه پیاده سازی یک وظیفه مندی خاص (Feature Location) (۱ جلسه)
 - (d) برنامه نویسی جنبه گرا و جستجو در سیستم های نرم افزاری به منظور شناسایی جنبه ها (Aspect-Oriented Programming & Aspect Mining) (۲ جلسه)
 - (e) داده کاوی در مخزن های نرم افزاری (Mining Software Repositories) (۱ جلسه)
 - (f) سیستم های توصیه گر در مهندسی نرم افزار (Recommendation Systems in Software Engineering) (۱ جلسه)
 - (g) مکان یابی خطاها (Fault Localization) (۲ جلسه)
 - (h) پیش بینی خطاها (Bug Prediction) (۲ جلسه)
 - (i) درک چارچوب های نرم افزاری (Software Frameworks Comprehension) (۲ جلسه)
۳. بخش سوم: باز مهندسی سیستم های قدیمی
 - (a) باز مهندسی شیء گرا (Object-Oriented Reengineering) (۲ جلسه)
 - (b) مهاجرت سیستم های قدیمی (Migration of Legacy Systems) (۱ جلسه)
 - (c) بهبود طراحی و بازآرایی سیستم های نرم افزاری (Software Refactoring) (۱ جلسه)
۴. بخش چهارم: مفاهیم جدید در تکامل نرم افزار
 - (a) تکامل واسطه های برنامه نویسی کاربرد (API Evolution) (۱ جلسه)
 - (b) داده کاوی و تحلیل فروشگاه های برنامه های همراه (App Stores Mining and Analysis) (۲ جلسه)

- (c) بازبینی کد (Code Review) (۲ جلسه)
- (d) جمع‌سپاری در مهندسی نرم‌افزار (Crowdsourcing for Software Engineering) (۲ جلسه)
- (e) علم داده در مهندسی نرم‌افزار (Data Science for Software Engineering) (۱ جلسه)

ارزیابی

- **آزمون:** آزمون‌های میان‌نیمسال (۳۰ درصد نمره) و پایان‌نیمسال (۳۰ درصد نمره)
- **ارائه مقاله علمی:** هر دانشجو موظف به ارائه حداقل یک مقاله علمی می‌باشد که در یکی از بهترین کنفرانس‌ها یا مجلات مرتبط به درس در سال‌های اخیر به چاپ رسیده باشد (۱۰ درصد نمره).
- **پروژه و گزارش پژوهشی:** موضوع پروژه پژوهشی قبل از آزمون پایان‌نیمسال تعیین می‌شود. دانشجو کار پژوهش را با کمک استاد درس آغاز نموده و پس از انجام پروژه، نتیجه پژوهش را در قالب یک گزارش ارائه می‌دهد (۳۰ درصد نمره).

مراجع اصلی

- آخرین مقالات مروری (Survey) ارائه شده در زمینه سرفصل‌های مطالب در برترین مجلات مهندسی نرم‌افزار.
- T. Mens and S. Demeyer (Eds.), *Software Evolution*. Springer-Verlag, 2008.

مراجع کمکی

- C. Bird, T. Menzies, and T. Zimmermann, *The Art and Science of Analyzing Software Data*. Morgan Kaufmann, 2015.
- M. P. Robillard, W. Maalej, R. J. Walker, and T. Zimmermann, *Recommendation Systems in Software Engineering*. Springer-Verlag, 2014.
- P. Tripathy and K. Naik, *Software Evolution and Maintenance: A Practitioner's Approach*. Wiley, 2014.



Formal Program Development

طراح درس: سید حسن میریان حسین آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

این درس برای دانش جویان کارشناسی ارشد و دکتری ارایه می شود و هدف از آن پرداختن به روش های تولید برنامه از توصیف رسمی سیستم ها به طور سیستماتیک است. در این درس تولید برنامه از توصیف صوری و به خصوص تقلید مورد بحث قرار می گیرد. روش های دیگر از جمله جبر پالایش مورگان، متدولوژی B، تئوری انواع (Type Theory) و نسخه ساختی Z معرفی می گردند. ضمناً در حد امکان ابزارهای لازم برای به کارگیری این روش ها معرفی می گردد.

ریز مواد

۱. مقدمه ای بر تولید برنامه (۱ جلسه)
 - (a) چرا تولید برنامه از توصیف رسمی؟
 - (b) تولید برنامه (تبدیل، پالایش، تقلید و تئوری انواع)
۲. چرا لزوماً توصیف صوری مستقیماً قابل اجرا نیست؟ (۲ جلسه)
۳. تقلید (Animation) توصیف صوری (۳ جلسه)
 - (a) تقلید با استفاده از زبان های تابعی
 - (b) تقلید با استفاده از زبان های منطقی
 - (c) معرفی ابزارهای تقلید
۴. جبر پالایش مورگان (۸ جلسه)
 - (a) برنامه ها و پالایش
 - (b) انواع و اعلان ها
 - (c) جایگزینی و ترکیب ترتیبی
 - (d) جملات انتخابی
 - (e) ثابت های منطقی
 - (f) حلقه های تکرار
 - (g) مثال
 - (h) معنای قوانین پالایش
۵. متدولوژی B (۸ جلسه)
 - (a) ماشین های انتزاعی
 - (b) ساختن توصیف
 - (c) طراحی و پالایش
 - (d) اثبات و پیاده سازی

(e) مثال

۶. تئوری انواع (۸ جلسه)

(a) مقدمه‌ای بر ریاضیات ساختی (Constructive Mathematics)

(b) زبان برنامه نویسی در تئوری انواع Martin LÖf

(c) قواعد استنتاج در تئوری انواع Martin LÖf

(d) تئوری مجموعه‌های ساختی CZ

(e) اثبات درستی توصیف در CZ

(f) ترجمه درخت اثبات در CZ به درخت اثبات در تئوری انواع Martin LÖf

(g) استخراج برنامه از درخت اثبات در تئوری انواع Martin LÖf

(h) مثال

ارزیابی

- آزمون: حدود ۷ آزمون کوچک به عنوان آزمون میان‌نیمسال (۱۵ درصد نمره)، و آزمون پایان‌نیمسال (۵۵ درصد نمره)
- تمرین (۱۵ درصد نمره) و پروژه (۱۵ درصد نمره):
- دانشجویان به گروه‌های ۲ یا ۳ نفره تقسیم می‌شوند و هر گروه سه صورت برنامه در اندازه‌های کوچک، متوسط و بزرگ را پیشنهاد می‌نماید. پس از تصویب برنامه‌ها، هر گروه تمرینات (حداقل ۳ تمرین) را در طول نیمسال براساس مسایل پیشنهادی خود پاسخ خواهد داد.
- برنامه بزرگ صورت پروژه هر گروه را مشخص می‌کند که یک ماه پس از پایان امتحانات فرصت دارند تا پروژه خود را تحویل نمایند.
- دانشجویان باید حتی‌الامکان با استفاده از نرم‌افزارهای موجود در این زمینه کار کنند.
- سمینار: دانش‌جویان به طور اختیاری سمیناری را در ارتباط مطالب درس پس از گرفتن تایید ارایه می‌نمایند (۱۰ درصد نمره اضافه).

مراجع اصلی

- I.J. Hayes and C.B. Jones, *Specification are not (necessarily) executable*. Technical report UMCS-89-12-1, Computer Science Department, University of Manchester, December, 1989.
- N.E. Fuchs, *Specifications are (Preferably) Executable*. Software Engineering Journal, Volume 7, No. 5, pp. 323-334, September 1992.
- A. Diller, *Z: An Introduction to Formal Methods*. Wiley, 1992.
- X. Jia, *A Tutorial of ZANS - A Z Animation System*, Version 0.21. Electronically available from: <ftp://ise.cs.depaul.edu>, May 1995.
- S.H. Valentine, *Z⁻, an Executable Subset of Z*. In John E. Nicholls, editor, *Z User Workshop - York 1991 - Workshops in Computing*, Springer-Verlag, pp. 157-187, 1992.
- C. Morgan, *Programming from Specifications*. Prentice-Hall, 1990.
- K. Lano and H. Haughton, *Specification in B: An Introduction Using the B Toolkit*. Imperial College Press, 1996.
- B. Nordstrom, K. Petersson, and J.M. Smith, *Programming in Martin LÖf's Type Theory: An Introduction*. Oxford University Press, 1990.

- J. Woodcock and J. Davies, *Using Z Specifications, Refinement, and Proof*. Prentice-Hall, 1996.
- D. Gries and F.B. Schneider, *A Logical Approach to Discrete Math*. Springer-Verlag, 1993.



Software Requirements Engineering

طراح درس: عباس حیدرنوری

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش مهندسی نرم‌افزار	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

پیاده‌سازی یک سیستم کامپیوتری با شکست مواجه خواهد شد اگر به نیازمندی‌های آن سیستم به درستی پاسخ داده نشود. همراه با گسترش و افزایش پیچیدگی سیستم‌های کامپیوتری، شناسایی دقیق نیازمندی‌های آنها نیز به فرآیندی پیچیده تبدیل گردیده است. بنابراین، لازم است تا از روش‌ها و تکنیک‌هایی برای شناسایی، توصیف و مستندسازی، مدل‌سازی، و اعتبارسنجی نیازمندی‌های سیستم‌های کامپیوتری استفاده شود. هدف از این درس، آشناسازی دانشجویان مقاطع تحصیلات تکمیلی با این روش‌ها و تکنیک‌هاست. به طور خاص، مباحثی که در این درس مطرح خواهند شد عبارتند از: (۱) استخراج و شناسایی نیازمندی‌های وظیفه‌مندی و غیر وظیفه‌مندی، (۲) سازمان‌دهی و اولویت‌بندی نیازمندی‌ها، (۳) تکنیک‌های توصیف و مستندسازی نیازمندی‌ها، (۴) تکنیک‌های مدل‌سازی نیازمندی‌ها، (۵) تکنیک‌های تحلیل، واریسی، و اعتبارسنجی نیازمندی‌ها.

ریز مواد

۱. مبانی مهندسی نیازمندی‌ها (Basics of Requirements Engineering) (۲ جلسه)
۲. درک دامنه مسئله و استخراج نیازمندی‌ها (Domain Understanding & Requirements Elicitation) (۲ جلسه)
۳. ارزیابی نیازمندی‌ها (Requirements Evaluation) (۲ جلسه)
۴. توصیف و مستندسازی نیازمندی‌ها (Requirements Specification and Documentation) (۲ جلسه)
۵. تضمین کیفی نیازمندی‌ها (Requirements Quality Assurance) (۱ جلسه)
۶. تکامل نیازمندی‌ها و ردیابی (Requirements Evolution and Traceability) (۱ جلسه)
۷. مقصودگرایی در مهندسی نیازمندی‌ها (Goal Orientation in Requirements Engineering) (۲ جلسه)
۸. مدل‌سازی اهداف سیستم با استفاده از مدل‌های مقصود (Modeling System Objectives with Goal Diagrams) (۳ جلسه)
۹. تحلیل خطر با استفاده از مدل‌های مقصود (Risk Analysis on Goal Models) (۲ جلسه)
۱۰. مدل‌سازی نیازمندی‌ها با استفاده از نمودارهای سناریوگرا (Modeling Requirements with Scenario-Oriented Modeling) (۲ جلسه)
۱۱. مدل‌سازی نیازمندی‌ها با استفاده از نمودارهای UML (Modeling System Requirements with UML Diagrams) (۲ جلسه)
۱۲. مدل‌سازی عملیات سیستم (Modeling System Operations) (۱ جلسه)
۱۳. مدل‌سازی رفتار سیستم (Modeling System Behaviors) (۲ جلسه)
۱۴. واریسی و اعتبارسنجی نیازمندی‌ها (Requirements Verification and Validation) (۲ جلسه)
۱۵. مدیریت نیازمندی‌ها (Requirements Management) (۱ جلسه)

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال (۲۵ درصد نمره) و پایان‌نیمسال (۳۵ درصد نمره)
- ارائه مقاله علمی: هر دانشجو موظف به ارائه یک مقاله علمی می‌باشد که در یکی از بهترین کنفرانس‌ها یا مجلات مرتبط به درس در سال‌های اخیر به چاپ رسیده باشد (۱۰ درصد نمره).
- پروژه و گزارش پژوهشی: موضوع پروژه پژوهشی قبل از آزمون پایان‌نیمسال تعیین می‌شود. دانشجو کار پژوهش را با کمک استاد درس آغاز نموده و پس از انجام پروژه، نتیجه پژوهش را در قالب گزارش ارائه می‌دهد (۳۰ درصد نمره).

مراجع اصلی

- A. van Lamsweerde, *Requirements Engineering: From System Goals to UML Models to Software Specifications*. Wiley, 2009.
- K. Pohl, *Requirements Engineering: Fundamentals, Principles, and Techniques*. Springer-Verlag, 2010.

مراجع اضافی

- G. Kotonya and I. Sommerville, *Requirements Engineering: Processes and Techniques*. Wiley, 1998.
- S. Robertson and J.C. Robertson, *Mastering the Requirements Process*, 2nd ed. Addison-Wesley, 2006.
- B. Berenbach, D. Paulish, J. Kazmeier, and A. Rudorfer, *Software & Systems Requirements Engineering: In Practice*. McGraw-Hill, 2009.
- E. Hull, K. Jackson, and J. Dick, *Requirements Engineering*, 3rd ed. Springer-Verlag, 2010.
- J. Holt, S. Perry, and M. Brownsword, *Model-Based Requirements Engineering*. Institution of Engineering and Technology (IET), 2011.
- D. Leffingwell, *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley, 2011.
- M. Chemuturi, *Requirements Engineering and Management for Software Development Projects*. Springer-Verlag, 2012.



Software Development Methodologies

طراح درس: رامان رامسین

گرایش: نرم‌افزار	مقطع: تحصیلات تکمیلی
واحد: ۳	نوع درس: زیرگرایش مهندسی نرم‌افزار
پیش‌نیاز: -	هم‌نیاز: -

اهداف درس

هدف این درس، آشنا کردن دانشجویان کارشناسی ارشد و دکترا با متدولوژی‌های ایجاد نرم‌افزار و مفاهیم و اصول مرتبط است. دانشجویان ضمن آشنایی با متدولوژی‌های مطرح، با روش‌های تحلیل و ارزیابی متدولوژی‌ها، الگوها/پادالگوها و متامدل‌های فرایند ایجاد نرم‌افزار، و روش‌های مهندسی متدولوژی آشنا می‌شوند. این درس از نظر ساختار و محتوا متناظر با درس متدها می‌باشد که از طرف انستیتو مهندسی نرم‌افزار (SEI) پیشنهاد شده و از دروس اصلی کارشناسی ارشد مهندسی نرم‌افزار (MSE) در دانشگاه Carnegie Mellon است. با توجه به اینکه در حال حاضر مشی شی‌گرا در بین متدولوژی‌ها مبنای غالب است، ساختار و محتوای فعلی درس عمدتاً بر متدولوژی‌های شی‌گرا تمرکز دارد.

ریز مواد

۱. مقدمه - معرفی تاریخچه تکاملی متدولوژی‌های شی‌گرا و معیارهای ارزیابی مربوطه (۲ جلسه - مدت هر جلسه، نود دقیقه است)
۲. معرفی تحلیلی متدولوژی Fusion - بررسی نمودهای بارز مشی شی‌گرا (۲ جلسه)
۳. معرفی اجمالی متدولوژی‌های شاخص نسل‌های اول و دوم - متدولوژی‌های RDD، Booch، OMT، و OOSE (۳ جلسه)
۴. معرفی تحلیلی متدولوژی‌های نسل سوم - متدولوژی‌های RUP، USDP، EUP، FOOM، و TSP-PSP (۷ جلسه)
۵. معرفی تحلیلی متدولوژی‌ها و چارچوب‌های چابک - متدولوژی‌های XP، AUP، Crystal، FDD، DSDM، Scrum، و DAD (۹ جلسه)
۶. الگوها و پادالگوهای فرایند ایجاد نرم‌افزار (۳ جلسه)
۷. متامدل‌های فرایند ایجاد نرم‌افزار (۱ جلسه)
۸. روش‌های مهندسی متدولوژی (۳ جلسه)

ارزیابی

- آزمون: آزمون‌های میان‌نیمسال و پایان‌نیمسال (۶۰ درصد نمره)
- تمرین پژوهشی: سه تمرین که در طول نیمسال تحویل داده می‌شوند (۲۵ درصد نمره).
- تمرین عملی: یک تمرین مهندسی متدولوژی با ابزار EPFC (۱۵ درصد نمره)

مراجع اصلی

- S.W. Ambler, J. Nalbone, and M.J. Vizdos, *The Enterprise Unified Process: Extending the Rational Unified Process*. Prentice-Hall, 2005.
- A. Cockburn, *Agile Software Development: The Cooperative Game*, 2nd ed. Addison-Wesley, 2006.

- P. Shoval, *Functional and Object Oriented Analysis and Design: An Integrated Methodology*. Idea Group Publishing, 2007.
- OMG, *Software and Systems Process Engineering Metamodel Specification (v2.0)*. OMG, 2008.
- R. Ramsin and R.F. Paige, "Process-centered review of object-oriented software development methodologies," *ACM Computing Surveys* 40, 1, Article 3, pp. 1-89, 2008.
- K.S. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*. Addison-Wesley, 2012.
- S.W. Ambler and M. Lines, *Disciplined Agile Delivery: A Practitioner's Guide to Agile Software Delivery in the Enterprise*. IBM Press, 2012.
- B. Henderson-Sellers, J. Ralyté, P.J. Agerfalk, and M. Rossi, *Situational Method Engineering*. Springer-Verlag, 2014.
- Agile Business Consortium, *The DSDM Agile Project Framework Handbook*. DSDM Consortium, 2014.



Program Specification and Verification

دانشکده مهندسی کامپیوتر

طراح درس: سید حسن میریان حسین آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم‌افزار
نوع درس: زیرگرایش مهندسی نرم‌افزار	واحد: ۳
هم‌نیاز: -	پیش‌نیاز: -

اهداف درس

این درس برای دانش‌جویان کارشناسی ارشد و دکتری ارایه می‌شود و هدف از آن پرداختن به روش‌های صوری برای توصیف و واریسی سیستم‌ها است. در این درس ابزارهای لازم برای به کارگیری این روش‌ها معرفی و در مورد رابطه بین توصیف صوری و پیاده‌سازی به طور اختصار بحث می‌گردد.

ریز مواد

۱. مقدمه‌ای بر توصیف سیستم‌ها (۲ جلسه)

- (a) چرا توصیف صوری؟
- (b) توصیف صوری و مهندسی نرم‌افزار
- (c) تولید برنامه از توصیف (پالایش)
- ۲. جبر گزاره‌ها، جبر مسندات (۲ جلسه)
- ۳. تئوری مجموعه‌ها و زبان Z (۴ جلسه)
 - (a) تساوی
 - (b) انواع، مجموعه‌ها و عملیات روی آنها
 - (c) تعاریف
 - (d) روابط و عملیات روی آنها
 - (e) توابع و عملیات روی آنها
 - (f) مثال

۴. واحدهای ساختاری توصیف (۴ جلسه)

- (a) شِما (Schema) و نحوه مدل کردن سیستم
- (b) استفاده از شِما به عنوان اعلان، نوع و مسند
- (c) شِمای ژنریک
- (d) نحوه بیان اصول (Axiomatic Description)
- (e) مثال

۵. جبر شِماها (Schema Calculus) (۴ جلسه)

- (a) تغییر متغیر (Renaming and Decoration)
- (b) ترکیب شِماها با استفاده از $\wedge, \vee, \neg, \Rightarrow, \exists, \forall, \setminus$ و Inclusion
- (c) مثال

۶. ابزارگان ریاضی Z (۲ جلسه)

(a) ردیف‌ها و Bagها و عملیات روی آنها

(b) نوع آزاد (Free Type)

(c) مثال

۷. توصیف با استفاده از ارتقا (Promotion) (۲ جلسه)

۸. امکان پذیری توصیف و محاسبه پیش شرط‌ها (Precondition) (۲ جلسه)

۹. واریسی (Verification) (۶ جلسه)

(a) اصول تئوری مجموعه‌ها

(b) قوانین استنتاج

(c) قضیه حالت اولیه سیستم

(d) ساده‌سازی پیش شرط‌ها

(e) اثبات خصوصیات توصیف

(f) مثال

۱۰. تولید برنامه از توصیف صوری Z با استفاده از پالایش (Refinement) (۲ جلسه)

(a) پالایش ساختارهای داده‌ای

(b) پالایش عملیات

(c) مثال

ارزیابی

- آزمون: حدود ۷ آزمون کوچک به جای آزمون میان‌نیمسال (۱۵ درصد نمره)، و آزمون پایان‌نیمسال (۵۵ درصد نمره)
- تمرین (۱۵ درصد نمره) و پروژه (۱۵ درصد نمره):
 - دانشجویان به گروه‌های ۲ یا ۳ نفره تقسیم می‌شوند و هر گروه سه صورت برنامه در اندازه‌های کوچک، متوسط و بزرگ را پیشنهاد می‌نماید. پس از تصویب برنامه‌ها، هر گروه تمرینات (حداقل ۳ تمرین) را در طول نیمسال براساس مسایل پیشنهادی خود پاسخ خواهد داد.
 - برنامه بزرگ صورت پروژه هر گروه را مشخص می‌کند که یک ماه پس از پایان امتحانات فرصت دارند تا توصیف صوری کامل پروژه را تحویل نمایند.
 - دانشجویان باید با استفاده از نرم‌افزارهای کنترل کننده جامعیت و عدم تناقض و اثبات قضیه خصوصیات توصیف صوری خود را مورد ارزیابی قرار دهند.
- سمینار: دانش‌جویان به طور اختیاری سمیناری را در ارتباط با مطالب درس پس از گرفتن تایید ارایه می‌نمایند (۱۰ درصد نمره اضافه).

مرجع اصلی

- J. Woodcock and J. Davies, *Using Z Specifications, Refinement, and Proof*. Prentice-Hall, 1996.

مراجع کمکی

- D. Gries and F.B. Schneider, *A Logical Approach to Discrete Math*. Springer-Verlag, 1993.
- C. Morgan, *Programming from Specifications*. Prentice-Hall, 1990.



Patterns in Software Engineering

دانشکده مهندسی کامپیوتر

طراح درس: رامان رامسین

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

هدف این درس، آشنا کردن دانشجویان کارشناسی ارشد و دکترا با الگوها و کاربرد آنها در مهندسی نرم افزار است. دانشجویان ضمن آشنایی با الگوهای رایج تحلیل، طراحی، معماری، مهندسی مجدد و مهندسی فرایند، با الگوهای بازآرایی کد و پادالگوها نیز آشنا می شوند. به دلیل تعدد الگوها، سعی می شود ضمن تأکید بر معرفی تفصیلی الگوهای پرکاربرد، آشنایی کافی با ساختارها و اصول مبنایی و روش های مدیریت پیچیدگی و تحلیل الگوها نیز حاصل شود.

ریز مواد

۱. مقدمه: مبانی و تاریخچه (۱ جلسه - مدت هر جلسه، نود دقیقه است)
۲. الگوهای پایه Coad (۱ جلسه)
۳. الگوهای GoF (۵ جلسه)
۴. اصول و قواعد شیء گرایی در قالب الگوها - الگوهای GRASP (۲ جلسه)
۵. الگوهای معماری GoV (۴ جلسه)
۶. الگوهای طراحی GoV (۲ جلسه)
۷. الگوهای بازآرایی کد (۴ جلسه)
۸. الگوهای مهندسی مجدد (۴ جلسه)
۹. الگوهای فرایند ایجاد نرم افزار (۱ جلسه)
۱۰. پادالگوها (۳ جلسه)
۱۱. الگوهای تحلیل Fowler (۲ جلسه)
۱۲. روش های طبقه بندی، مدیریت پیچیدگی و تحلیل الگوها (۱ جلسه)

ارزیابی

- آزمون: آزمون های میان نیمسال و پایان نیمسال (۶۰ درصد نمره)
- تمرین: سه یا چهار تمرین پژوهشی که در طول نیمسال تحویل داده می شوند (۲۰ درصد نمره).
- گزارش پژوهشی: فهرستی از موضوعات پژوهشی توسط استاد درس به دانشجویان پیشنهاد می شود تا دانشجویان موضوع مورد علاقه خود را از آن فهرست انتخاب کنند. دانشجویان کار پژوهش را با کمک استاد درس آغاز می کنند و نتایج را نهایتاً در قالب یک گزارش پژوهشی تحویل می دهند (۲۰ درصد نمره).

- E. Gamma, R. Helm, R. Johnson, and J. Vlissides, *Design Patterns: Elements of Reusable Object-Oriented Software*. Addison-Wesley, 1995.
- F. Buschmann, R. Meunier, H. Rohnert, P. Sommerlad, and M. Stal, *Pattern-Oriented Software Architecture: A System of Patterns*, Vol. 1. Wiley, 1996.
- M. Fowler, *Analysis Patterns: Reusable Object Models*. Addison-Wesley, 1996.
- J. Kerievsky, *Refactoring to Patterns*. Addison-Wesley, 2004.
- A. Shalloway and J. Trott, *Design Patterns Explained: A New Perspective on Object-Oriented Design*, 2nd ed. Addison-Wesley, 2005.
- D. Manolescu, M. Voelter, and J. Noble, *Pattern Languages of Program Design*, Vol. 5. Addison-Wesley, 2006.
- F. Buschmann, K. Henney, and D.C. Schmidt, *Pattern-Oriented Software Architecture: On Patterns and Pattern Languages*, Vol. 5. Wiley, 2007.

آزمون نرم افزار (۴۰۸۲۸)



دانشکده مهندسی کامپیوتر

Software Testing

طراح درس: سید حسن میریان حسین آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

این درس برای دانش جویان کارشناسی ارشد و دکتری ارایه می شود و هدف از آن پرداختن به روش های مدل-رانه تولید داده آزمون از روی ساختارهای منتزع شده از مدل های مختلف نرم افزار از جمله کد و ورودی به طور سیستماتیک است. همچنین در طی این درس در حد امکان ابزارهای لازم برای خودکار سازی مراحل آزمون نرم افزار معرفی می گردد.

ریز مواد

۱. مقدمه (۲ جلسه)
۲. آزمون مدل رانه (۲ جلسه)
۳. آزمون خودکار (۲ جلسه)
۴. آزمون چابک (۲ جلسه)
۵. معیارهای پوشش (۲ جلسه)
۶. افراز فضای ورودی (۴ جلسه)
۷. پوشش گراف (۴ جلسه)
۸. پوشش منطق (۴ جلسه)
۹. آزمون مبتنی بر نحو (۴ جلسه)
۱۰. ملاحظات عملی (۲ جلسه)
۱۱. ساخت ابزار آزمون (۲ جلسه)

ارزیابی

- آزمون: حدود ۷ آزمون کوچک به عنوان آزمون میان نیمسال (۱۵ درصد نمره)، و آزمون پایان نیمسال (۵۵ درصد نمره)
- تمرین (۱۵ درصد نمره) و پروژه (۱۵ درصد نمره):
- دانشجویان به گروه های ۲ یا ۳ نفره تقسیم می شوند و هر گروه سه صورت برنامه در اندازه های کوچک، متوسط و بزرگ را پیشنهاد می نماید. پس از تصویب برنامه ها، هر گروه تمرینات (حداقل ۳ تمرین) را در طول نیمسال براساس مسایل پیشنهادی خود پاسخ خواهد داد.
- برنامه بزرگ صورت پروژه هر گروه را مشخص می کند که یک ماه پس از پایان امتحانات فرصت دارند تا پروژه خود را تحویل نمایند.
- دانشجویان باید حتی الامکان با استفاده از نرم افزارهای موجود در این زمینه کار کنند.
- سمینار: دانش جویان به طور اختیاری سمیناری را در ارتباط با مطالب درس پس از گرفتن تایید ارایه می نمایند (۱۰ درصد نمره اضافه).

- P. Ammann and J. Offutt, *Introduction to Software Testing*. Cambridge University Press, 2017.



Software Engineering 2

طراح درس: سید حسن میریان حسین آبادی

مقطع: تحصیلات تکمیلی	گرایش: نرم افزار
نوع درس: زیرگرایش مهندسی نرم افزار	واحد: ۳
هم نیاز: -	پیش نیاز: -

اهداف درس

این درس برای دانشجویان کارشناسی ارشد و دکتری ارائه می شود و هدف از آن پرداختن به مباحث پیشرفته در ارتباط با مهندسی نرم افزار است. در این درس روش های جدید در مورد هر یک از مراحل چرخه حیات نرم افزار مورد بحث قرار می گیرند که از آن جمله می توان به استفاده از روش های صوری در ثبت نیازها، روش مبتنی بر خانواده در تجزیه و تحلیل و طراحی و پیاده سازی، مهندسی امنیت، روش مبتنی بر جنبه و روش مبتنی بر خدمت اشاره نمود. ابزارهای لازم برای به کارگیری این روش ها به اختصار و در حد امکان معرفی می گردند.

ریز مواد

۱. یادآوری: مروری بر متدولوژی های چرخه حیات، مدیریت پروژه، برآورد، مدیریت ریسک، اندازه گیری و آزمون (۴ جلسه)
۲. روش های صوری ثبت نیازها و زبان های بر مبنای مدل و زبان های جبری (۶ جلسه)
۳. متدولوژی خط تولید و روش مبتنی بر خانواده (۱۲ جلسه)
۴. مهندسی امنیت (۴ جلسه)
۵. روش مبتنی بر جنبه (۲ جلسه)
۶. روش مبتنی بر خدمت (۲ جلسه)

ارزیابی

- آزمون: آزمون های کلاسی (۱۵ درصد نمره)، آزمون پایان نیمسال (۵۰ درصد نمره)
- تمرین: حدود ۵ تمرین نظری (۱۵ درصد نمره)
- مقاله: تعیین موضوعی مرتبط با موضوعات درس، تصویب موضوع، تهیه مقاله، ارائه سمینار، تحویل نسخه نهایی مقاله (۲۰ درصد نمره)

مراجع اصلی

- R.S. Pressman, *Software Engineering: A Practitioner's Approach*, 8th ed. McGraw-Hill, 2015.
- I. Sommerville, *Software Engineering*, 9th ed. Addison-Wesley, 2011.
- D. Weiss and C.T.R. Lai, *Software Product-Line Engineering: A Family-Based Software Development Process*. Addison-Wesley, 1999.