

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

طراحی و تحلیل الگوریتم‌ها

مؤلف:

مهندس ابوالفضل اسفندی

عضو هیأت علمی دانشگاه آزاد اسلامی واحد بروجرد



سرشناسه	: اسفندی، ابوالفضل
عنوان و نام پدیدآور	: طراحی و تحلیل الگوریتم ها / مولف ابوالفضل اسفندی.
مشخصات نشر	: بروجرد: دانشگاه آزاد اسلامی، واحد بروجرد، ۱۳۹۲.
مشخصات ظاهری	: ۳۰۶ص.
شابک	: 978-964-10-2413-2 : ۷۰۰۰۰ ریال
وضعیت فهرست نویسی	: فیبا
موضوع	: الگوریتم‌های کامپیوتری
شناسه افزوده	: دانشگاه آزاد اسلامی، واحد بروجرد
رده بندی کنگره	: ۱۳۹۲ الف ۹ الف ۵ / ۹ / ۷۶ / ۹ QA
رده بندی دیویی	: ۰۰۵ / ۱۲۰۲۸
شماره کتابشناسی ملی	: ۳۲۵۷۲۶۷



دانشگاه آزاد اسلامی واحد بروجرد

طراحی و تحلیل الگوریتم‌ها

مؤلف:

مهندس ابوالفضل اسفندی

ناشر: دانشگاه آزاد اسلامی واحد بروجرد

مدیر مسؤول: دکتر علی آریاپور

ناظر چاپ: حسام‌الدین یاریار

چاپ و صحافی: نویسنده

نوبت چاپ: اول / ۱۳۹۲

شمارگان: ۲۰۰۰

بها: ۷۰۰۰۰ ریال

شابک: ۹۷۸-۹۶۴-۱۰-۲۴۱۳-۲

حق چاپ برای ناشر محفوظ است

بروجرد- میدان نواب - خیابان یادگار امام - مجتمع دانشگاهی امام خمینی(ره)
تلفکس ۰۶۶۲-۳۵۱۸۰۱۳

Email: Publisher@iaub.ac.ir

سخن ناشر

به نام خداوند لوح و قلم حقیقت نگار وجود و عدم
خدایی که داننده رازهاست نخستین سرآغاز آغازهاست

دانشگاه آزاد اسلامی، به عنوان یکی از ارکان نظام آموزش عالی کشور در چشم‌انداز بیست ساله خود، توسعه کیفی را به طور جدی مدنظر قرار داده است. دانشگاه آزاد اسلامی بروجرد به عنوان یکی از باسابقه‌ترین واحدهای دانشگاهی سطح کشور با عنایت به پتانسیل‌های موجود و در راستای این حرکت عظیم علمی و فرهنگی و به منظور ایجاد سرعت، دقت و هماهنگی بیشتر در اجرای کلیه امور مربوط به تأمین منابع آموزشی مورد نیاز و همچنین تشویق و ترغیب محققان، مؤلفان و مترجمان دانشگاهی به نگارش و ترجمه کتب معتبر علمی، شورای انتشارات علمی خود را تشکیل داد. بدیهی است که ورود به این عرصه خطیر جز با استمداد از خداوند متعال و چشم‌یاری داشتن از محققان و پژوهشگران و استادان ارجمند امکان‌پذیر نیست. لذا از همه صاحب‌نظران ارجمند دعوت می‌شود ما را از نقطه‌نظرها و راهنمایی‌های ارزشمند خود بهره‌مند سازند. در پایان بر خود لازم می‌دانم از جناب آقای دکتر فرهاد دانشجو ریاست دانشگاه آزاد اسلامی، معاون محترم پژوهشی دانشگاه به سبب برقراری شرایط و بستر مناسب انجام فعالیت‌های حوزه پژوهش واحد در بخش توسعه فعالیت‌های چاپ و نشر، تشکر و قدردانی نمایم. ضمناً شایسته است از فعالیت‌ها و تلاش‌های علمی آقای مهندس ابوالفضل اسفندی در تألیف کتاب طراحی و تحلیل الگوریتم‌ها مراتب سپاسگزاری خود را ابراز نمایم.

دکتر احمد سیف
رئیس دانشگاه آزاد اسلامی بروجرد

پیشگفتار

در دنیای امروز، نرم افزارهای رایانه‌ای بخش اعظمی از زندگی انسان‌ها را در زمینه‌های مختلف مثل صنعت، بهداشت و علوم پزشکی، امور نظامی، اتوماسیون‌های اداری و اجتماعی و در کل همه جنبه‌های زندگی آن‌ها را در بر گرفته است. به طوری که چه در سطح کلان و ملی و چه در حد زندگی روزمره افراد، نقش بسیار مهم و حساسی ایفا می‌کند و نادیده گرفتن آن تقریباً غیر ممکن شده است. حال با توجه به این اهمیت فراوان، انتظارات و توقعات افراد از سیستم‌ها و نرم افزارهای رایانه‌ای نیز روز به روز بیشتر شده است، به طوری که انتظار استفاده از سیستم‌هایی با قابلیت‌های بیشتر، کارآمدتر و امن‌تر را دارند و این افزایش تقاضاها باعث پیچیدگی بیشتر الگوریتم‌های مورد استفاده در این سیستم‌ها می‌شود، که شاید تناقضی با نیاز زندگی‌های امروزی بشر یعنی کارایی این سیستم‌ها در بحث سرعت پردازشی و پاسخگویی برای صرف زمان کمتر خواهد بود.

پس پاسخگویی به این مشکل و رفع این تناقض و تمرکز بر آن بسیار حیاتی به نظر می‌رسد، بنابراین طراحان و سازندگان آینده این نرم افزارها، یعنی دانشجویان فعلی، می‌بایست پس از آشنایی اولیه با برنامه‌سازی و ابزارهای مورد استفاده، قبل از هر چیز دیگر، با نحوه طراحی الگوریتم‌هایی موثر و کارا که تا حد امکان علاوه بر پاسخگویی به نیاز کاربران، کارایی لازم را نیز داشته باشند، آشنا شده و بتوانند الگوریتم‌های مختلف را تحلیل و با هم مقایسه کنند.

در زمینه درس طراحی الگوریتم‌ها، مراجع و کتاب‌های بسیار عالی و معتبری مانند دو کتاب بسیار ارزشمند: [Cormen, Leiserson, Revest and Stein "Introduction to Algorithms"] و [Neapolitan and Naimipour "Foundations Of Algorithms Using C++ Pseudo code"] وجود داشته به طوری که در اکثر دانشگاه‌ها تدریس آن‌ها متداول شده است و شاید این کتاب حاضر در مقایسه با آن‌ها جای نقد بسیار دارد. اما اگر به نحوه استفاده و تدریس این مراجع در دانشگاه‌های مختلف دقت شود، می‌بینید که غالباً به دلیل حجم زیاد مطالب، بیان مسائل و فصل‌هایی که خارج از سرفصل مصوب وزارت علوم، تحقیقات و فناوری در کشور ما بوده، بخش‌های اعظمی از آن‌ها هیچ‌گاه مورد استفاده و بحث قرار نمی‌گیرد.

به علاوه گاهی نبود یک روند منطقی در این کتاب‌ها و بریدگی مطالب و در برخی موارد با اضافه شدن ترجمه‌های نامناسب، بیشتر باعث سر در گمی دانشجویان و عدم بهره‌گیری مناسب و فهم درست مطالب از این کتاب‌ها شده است، به طوری که به دلیل مجموعه این عوامل گاهی دیده می‌شود که برخی دانشجویان ضعیف‌تر با وجود مطالب بسیار کامل، جامع و دقیق این کتاب‌ها از آن‌ها فراری هستند!!

بنابراین با توجه به اینکه درس طراحی الگوریتم، یکی از دروس بسیار مهم رشته کامپیوتر در مقاطع و آزمون‌های مختلف می‌باشد، بر آن شدیم که با استفاده از همین مراجع ارزشمند به خصوص با تمرکز بیشتر بر مرجع اول، یعنی کتاب "Introduction to Algorithms"، سعی در بیان مطالب آن‌ها به شیوه‌ای روان‌تر و قابل فهم‌تر کنیم، به طوری که با حفظ مطابقت با سرفصل مصوب، از بیان مطالب اضافه و شاید بی‌استفاده خودداری کرده و تمرکز خود را بجای افزایش تعداد صفحات کتاب، بر روی توضیح مطالب گاهاً پیچیده این درس به زبان ساده و کاربردی که حتی برای دانشجویان لذت بخش هم باشد، قرار دادیم. در ضمن تا حد امکان از ورود و پرداختن به مباحث پیچیده ریاضی و اثبات‌های غیر ضروری، اجتناب کرده‌ایم.

کتاب "طراحی و تحلیل الگوریتم‌ها" برای دانشجویان رشته کامپیوتر و کلیه افراد دخیل در بحث برنامه‌نویسی نرم افزار و الگوریتم نویسی، نوشته شده است. در این کتاب سعی شده در کمال سادگی و مختصر گویی، کلیه مطالب مورد نیاز دانش پژوهان به همراه مثال‌هایی کاربردی ارائه گردد و کم‌ترین ارجاع به مطالب درس‌های دیگر را داشته باشد و هر جا نیز به اطلاعاتی مقدماتی نیاز بوده است، سعی شده که یادآوری از آن مطالب ذکر گردد. همچنین این کتاب برای داوطلبان کنکورهای کارشناسی ارشد و دکتری و نیز دانش آموزانی که خود را برای دوره‌های المپیاد رایانه آماده می‌کنند، می‌تواند به صورت مفید و قابل استفاده باشد. علاوه بر این ویژگی‌ها، کتاب حاضر بیش از ۹۰ درصد از مطالب درس طراحی الگوریتم‌ها، مطابق با سرفصل مصوب وزارت علوم، تحقیقات و فناوری را در بر می‌گیرد.

امید است که این کتاب بتواند در فراگیری بهتر دانشجویان و بهره‌گیری از مفاهیم بسیار جالب و زیبای این درس، تأثیری کوچک داشته باشد. البته به خوانندگان پیشنهاد می‌شود که قبل از خواندن این کتاب، مروری بر مطالب کتاب‌های ساختمان داده‌ها و زبان‌های برنامه‌سازی مانند ++C داشته باشند.

ساختار کتاب

این کتاب برای حفظ یک روند منطقی همراه با پیوستگی مطالب و بریده نشدن آن‌ها و جلوگیری از سردرگمی دانشجو در انبوهی از مطالب، در ۱۰ فصل مجزا به صورت زیر سازماندهی شده است:

- فصل اول. الگوریتم‌ها و نحوه تحلیل آن: در این فصل ابتدا به تعریف الگوریتم و مفاهیم وابسته به آن پرداخته و سپس به نحوه تحلیل و مقایسه الگوریتم‌ها از جنبه‌های مختلف می‌پردازیم. در ضمن با توجه به آنکه مهم‌ترین مسئله در بحث تحلیل الگوریتم‌ها، مسئله زمان اجرا و پیچیدگی زمانی یک الگوریتم است، به طور کامل همراه با بررسی چند الگوریتم معروف، به این بحث پرداخته می‌شود.
- فصل دوم. روابط بازگشتی و حل آن‌ها: در فصل دوم، ابتدا با روابط بازگشتی جهت یادآوری از درس ساختمان داده‌ها، آشنا شده و سپس روش‌های مختلف حل این گونه روابط را با ذکر مثال‌هایی بررسی می‌کنیم.
- فصل سوم. روش تقسیم و حل: در این فصل با ساختار کلی روش تقسیم و حل آشنا شده و با بررسی الگوریتم‌های مختلفی، مانند چندین روش مرتب‌سازی که به این روش حل می‌شوند، دانشجو را با نحوه به کارگیری آن آشنا می‌کنیم.
- فصل چهارم. مرتب‌سازی هرمی و صف اولویت: هدف از این فصل آشنایی با داده ساختار Heap و دو کاربرد آن یعنی مرتب‌سازی هرمی و صف اولویت آشنا می‌شویم، که در آن مرتب‌سازی هرمی در ادامه الگوریتم‌های مرتب‌سازی فصل سوم و صف اولویت به دلیل کاربرد بسیار زیاد در الگوریتم‌های مختلف بررسی می‌شود.
- فصل پنجم. مرتب‌سازی‌های خطی: در فصل پنجم سه الگوریتم مرتب‌سازی خاص که برخلاف الگوریتم‌های مرتب‌سازی فصل‌های قبل، با ساختاری متفاوت در زمان خطی قادر به مرتب‌سازی عناصر هستند، آشنا می‌شویم.
- فصل ششم. روش‌های مبتنی بر استقرار و برنامه‌ریزی پویا: در این فصل دو روش متداول دیگر از روش‌های طراحی الگوریتم را به صورت جداگانه بررسی می‌کنیم. در هر یک از این روش‌ها، پس از بیان روش کار هر یک، مثال‌های متنوعی را به خصوص برای روش برنامه‌ریزی پویا که نوعی روش بهینه‌سازی است، بررسی می‌دهیم.

- فصل هفتم. روش حریرصانه: هدف از این فصل بررسی یکی از روش‌های ساده طراحی الگوریتم با نام روش حریرصانه می‌باشد که ایده ارائه آن‌ها بسیار ساده بوده و در زندگی روزمره نیز بکار گرفته می‌شوند، اما در عین سادگی نیاز به اثبات درستی روش پیشنهادی دارند.
 - فصل هشتم. الگوریتم‌های گراف: در این فصل ابتدا به بررسی مقدماتی گراف‌ها و روش‌های نمایش گراف‌ها آشنا شده و سپس به بررسی و تحلیل الگوریتم‌های مختلف آن‌ها مانند الگوریتم‌های درخت پوشای کمینه و یافتن کوتاه‌ترین مسیرها در یک گراف می‌پردازیم.
 - فصل نهم. جستجوی فضای حالت: در فصل نهم با مجموعه روش‌هایی برای طراحی الگوریتم که در آن‌ها نحوه کار بررسی فضای حالات ممکن برای جواب است، مانند روش‌های پس‌گرد و انشعاب و تحدید، آشنا می‌شویم.
 - فصل دهم. پیچیدگی محاسباتی و نظریه NP: هدف از این فصل بررسی بعضی از مفاهیم که در بحث پیچیدگی‌های الگوریتم‌ها و محاسبه زمان اجرای آن‌ها و طبقه‌بندی الگوریتم‌ها مطرح می‌شود و به خصوص نظریه NP و مفاهیم وابسته به آن می‌باشد.
- در اینجا لازم می‌دانم که از جناب آقای دکتر احمد سیف ریاست دانشگاه آزاد اسلامی بروجرد، دکتر مهدیان معاونت پژوهش و فناوری و مهندس یاریار مدیر انتشارات علمی بروجرد به خاطر همکاری‌های بی‌دریغشان قدردانی کرده و از آقای مهندس حسین عسگرپور که در نوشتن این کتاب کمک‌های شایان نمودند، تشکر ویژه‌ای داشته باشم. همچنین از داوران محترم که با نظرات اصلاحی و سازنده خود موجب ارتقای علمی این کتاب شده‌اند، سپاسگزارم و البته قدردانی و تشکری مخصوص و خاضعانه از استاد بزرگوارم، جناب آقای دکتر حمید ضرابی زاده دارم که همیشه خود را مدیون ایشان دانسته و آموخته‌های ناچیزم در این زمینه را از ایشان می‌دانم.
- کتاب حاضر دسترنج ۷ سال تدریس درس طراحی الگوریتم‌ها می‌باشد، لیکن نمی‌تواند خالی از ایراد و اشکال علمی و چاپی باشد. بدیهی است که فرستادن راهنمایی‌ها و انتقادات سازنده اساتید، دانشجویان و سایر دانش پژوهان به آدرس ناشر یا ایمیل اینجانب (AEsfandi@IAUB.AC.IR) می‌تواند موجب تصحیح کتاب و دلگرمی نویسنده گردد.

ابوالفضل اسفندی
۱۳۹۲

تقدیم به:

پدرو مادر عزیزتر از جانم که هرچه دارم از این دو فرشته است.

و امید و تنها یادگار برادرم...

محمد صالح

فهرست مطالب

فصل اول: الگوریتم‌ها و نحوه تحلیل آن‌ها

۱-۱	مقدمه	۲۴
۲-۱	آشنایی با الگوریتم	۲۴
۳-۱	تحلیل الگوریتم‌ها	۲۶
۴-۱	مرتبه الگوریتم و توابع رشد	۲۷
۱-۴-۱	توابع رشد	۲۷
۲-۴-۱	قضایا و خواص مربوط به نمادها	۳۰
۵-۱	محاسبه زمان اجرا	۳۱
۶-۱	جستجوی خطی	۳۵
۷-۱	مرتب‌سازی درجی	۳۶
۸-۱	تحلیل الگوریتم	۳۸
۹-۱	تمرینات فصل اول	۴۳
۱۰-۱	تست‌های فصل اول	۴۵

فصل دوم: روابط بازگشتی و حل آن‌ها

۱-۲	مقدمه	۵۰
۲-۲	روش جایگذاری (حدس و استقرا)	۵۱
۳-۲	روش درخت بازگشت	۵۳
۴-۲	قضیه اصلی	۵۴
۵-۲	روش‌های حل رابطه‌های بازگشتی همگن و غیر همگن	۵۸
۱-۵-۲	رابطه‌های بازگشتی همگن	۵۸
۲-۵-۲	رابطه‌های بازگشتی غیر همگن	۵۹

- ۶-۲ تکنیک تغییر متغیر ۶۰
- ۷-۲ برخی روابط ریاضی ۶۱
- ۸-۲ تمرینات فصل دوم ۶۳
- ۹-۲ تست‌های فصل دوم ۶۵

فصل سوم: روش تقسیم و حل

- ۱-۳ مقدمه ۶۸
- ۲-۳ ساختار روش تقسیم و حل ۶۸
- ۳-۳ جستجوی دودویی ۶۹
- ۴-۳ مرتب‌سازی ادغامی ۷۱
- ۵-۳ ضرب دو عدد n بیتی ۷۸
- ۶-۳ مرتب‌سازی سریع ۸۲
- ۱-۶-۳ نحوه کار مرتب‌سازی سریع ۸۲
- ۲-۶-۳ نسخه‌های دیگر مرتب‌سازی سریع (مرتب‌سازی سریع تصادفی) ۹۰
- ۳-۶-۳ چرا مرتب‌سازی سریع؟! ۹۱
- ۷-۳ یافتن مرتبه آماری k ام ۹۳
- ۱-۷-۳ یافتن بزرگ‌ترین عنصر ۹۳
- ۲-۷-۳ یافتن بزرگ‌ترین و کوچک‌ترین عنصر ۹۴
- ۳-۷-۳ یافتن دومین بزرگ‌ترین عنصر ۹۶
- ۴-۷-۳ یافتن k امین عنصر کوچک‌تر ۹۸
- ۸-۳ ضرب ماتریسی استراسن ۱۰۰
- ۹-۳ تمرینات فصل سوم ۱۰۶
- ۱۰-۳ تست‌های فصل سوم ۱۱۰
- مرتب‌سازی هرمی و صف اولویت ۱۱۳

فصل چهارم: مرتب‌سازی هرمی و صف اولویت

- ۱-۴ مقدمه ۱۱۴
- ۲-۴ درخت Heap ۱۱۴
- ۳-۴ مرتب‌سازی هرمی ۱۱۷
- ۱-۳-۴ بازسازی مجدد heap ۱۱۷

۱۱۹	heap ساخت ۲-۳-۴
۱۲۱	مرتب‌سازی ۳-۳-۴
۱۲۳	صف اولویت ۴-۴
۱۲۷	تمرینات فصل چهارم ۵-۴
۱۲۹	تست‌های فصل چهارم ۶-۴

فصل پنجم: مرتب‌سازی‌های خطی

۱۳۴	۱-۵ مقدمه
۱۳۵	۲-۵ مرتب‌سازی شمارشی
۱۳۷	۳-۵ مرتب‌سازی سطلی
۱۳۹	۴-۵ مرتب‌سازی مبنایی (Radix Sort)
۱۴۳	۵-۵ تمرینات فصل پنجم
۱۴۵	۶-۵ تست‌های فصل پنجم

فصل ششم: روش‌های مبتنی بر استقرار و برنامه‌ریزی پویا

۱۴۸	۱-۶ مقدمه
۱۴۸	۲-۶ روش مبتنی بر استقرار
۱۴۹	۱-۲-۶ مسئله ستاره مشهور
۱۵۴	۳-۶ روش برنامه‌ریزی پویا
۱۵۵	۱-۳-۶ یافتن n مین جمله فیوناچی
۱۵۷	۲-۳-۶ ترکیب k از n
۱۶۰	۳-۳-۶ ضرب زنجیره‌ای ماتریس‌ها
۱۶۹	۴-۳-۶ بزرگ‌ترین زیر دنباله مشترک
۱۷۴	۵-۳-۶ مثلث بندی بهینه یک چند ضلعی محدب
۱۷۷	۶-۳-۶ مسئله کوله‌پشتی
۱۸۳	۴-۶ تمرینات فصل ششم
۱۸۶	۵-۶ تست‌های فصل ششم

فصل هفتم: روش حریصانه

۱۹۰	۱-۷ مقدمه
-----	-------	-----------

۱۹۰	۲-۷ خرد کردن پول.....
۱۹۳	۳-۷ مسئله کوله پشتی.....
۱۹۴	۱-۳-۷ روش حریصانه برای حل کوله پشتی صفر و یک.....
۱۹۵	۲-۳-۷ روش حریصانه برای حل کوله پشتی کسری.....
۱۹۶	۴-۷ مسئله زمان بندی کلاس ها.....
۲۰۰	۵-۷ زمان بندی با مهلت معین.....
۲۰۵	۶-۷ کد هافمن.....
۲۱۴	۷-۷ تمرینات فصل هفتم.....
۲۱۶	۸-۷ تست های فصل هفتم.....
۲۱۷	الگوریتم های گراف.....

فصل هشتم: الگوریتم های گراف

۲۱۸	۱-۸ مقدمه.....
۲۱۸	۲-۸ روش های نمایش گراف.....
۲۲۱	۳-۸ جستجوی اول عمق.....
۲۲۴	۴-۸ جستجوی اول سطح.....
۲۲۷	۵-۸ مرتب سازی توپولوژیکی.....
۲۲۸	۶-۸ درخت پوشای کمینه.....
۲۳۱	۱-۶-۸ الگوریتم پریم.....
۲۳۶	۲-۶-۸ الگوریتم کروسکال.....
۲۳۹	۷-۸ یافتن کوتاه ترین مسیر.....
۲۴۱	۱-۷-۸ کوتاه ترین مسیرها از یک مبدأ واحد (یا مقصدی واحد).....
۲۴۳	۱-۱-۷-۸ الگوریتم بلمن-فورد.....
۲۴۶	۲-۱-۷-۸ الگوریتم دیکسترا.....
۲۵۰	۲-۷-۸ کوتاه ترین مسیر بین همه جفت راس ها.....
۲۵۱	۱-۲-۷-۸ الگوریتم فلویید-وارشال.....
۲۵۶	۲-۲-۷-۸ الگوریتم جانسون.....
۲۶۰	۸-۸ تمرینات فصل هشتم.....
۲۶۳	۹-۸ تست های فصل هشتم.....

فصل نهم: جستجوی فضای حالت

۲۶۶	۱-۹ مقدمه
۲۶۷	۲-۹ روش پس‌گرد
۲۶۸	۱-۲-۹ مولد ترکیبات
۲۶۹	۲-۲-۹ مسئله Π وزیر
۲۷۵	۳-۹ درخت بازی
۲۷۷	۴-۹ محدود کردن فضای حالت
۲۷۸	۱-۴-۹ هرس کردن
۲۷۹	۲-۴-۹ انشعاب و تحدید
۲۸۰	۱-۲-۴-۹ فروشنده دوره‌گرد
۲۸۴	۵-۹ تمرینات فصل نهم
۲۸۶	۶-۹ تست‌های فصل نهم

فصل دهم: پیچیدگی محاسباتی و نظریه NP

۲۹۰	۱-۱۰ مقدمه
۲۹۰	۲-۱۰ کنترل ناپذیری
۲۹۲	۳-۱۰ نظریه NP
۲۹۳	۱-۳-۱۰ مجموعه‌های P و NP
۲۹۵	۲-۳-۱۰ مسائل NP کامل
۲۹۸	۳-۳-۱۰ تعمیم مجموعه NP
۳۰۲	۴-۱۰ تمرینات فصل دهم
۳۰۳	۵-۱۰ تست‌های فصل دهم
۳۰۵	منابع و مراجع انگلیسی
۳۰۶	منابع و مراجع فارسی

فهرست اشکال

- شکل ۱-۱: نمادهای Θ , O , Ω بصورت نموداری ۲۹
- شکل ۱-۲: نمودار زمان اجرای سه الگوریتم مختلف یافتن دنباله فیبوناچی ۳۵
- شکل ۱-۳: مرتب کردن ورق‌ها با استفاده مرتب‌سازی درجی ۳۷
- شکل ۱-۴: عملکرد الگوریتم مرتب‌سازی درجی ۳۸
- شکل ۱-۵: نمودار زمان اجرای الگوریتم مرتب‌سازی درجی ۴۱
- شکل ۳-۱: عملکرد الگوریتم MERGE ۷۴
- شکل ۳-۲: عملکرد الگوریتم مرتب‌سازی ادغامی ۷۶
- شکل ۳-۳: عملکرد الگوریتم مرتب‌سازی سریع و زیرروال PARTITION ۸۵
- شکل ۳-۴: درخت بازگشت مربوط به مرتب‌سازی سریع در حالت تقسیم‌بندی متعادل ۸۹
- شکل ۳-۵: درخت تصمیم‌گیری برای مرتب‌سازی سه عدد ۹۲
- شکل ۳-۶: ساختار تورنمنتی برای یافتن بزرگ‌ترین و دومین بزرگ‌ترین ۹۷
- شکل ۴-۱: شکل یک درخت Max-Heap و آرایه متناظر آن ۱۱۵
- شکل ۴-۲: شش درخت اول فیبوناچی ۱۱۶
- شکل ۴-۳: نحوه عملکرد زیرروال MAX-HEAPIFY ۱۱۸
- شکل ۴-۴: نحوه عملکرد زیرروال Build-Max-Heap ۱۲۰
- شکل ۴-۵: عملکرد الگوریتم مرتب‌سازی هرمی ۱۲۲
- شکل ۴-۶: عملکرد الگوریتم HEAP-INCREASE-KEY ۱۲۵
- شکل ۵-۱: نحوه عملکرد الگوریتم مرتب‌سازی شمارشی ۱۳۶

- شکل ۵-۲: نحوه عملکرد الگوریتم مرتب‌سازی سطلی..... ۱۳۸
- شکل ۵-۳: نحوه عملکرد الگوریتم مرتب‌سازی مبنایی..... ۱۳۹
- شکل ۵-۴: جدول مقایسه الگوریتم‌های مرتب‌سازی مختلف..... ۱۴۱
- شکل ۶-۱: سوالات مورد نیاز برای تعیین ستاره مشهور نهایی..... ۱۵۰
- شکل ۶-۲: سوالات لازم برای تعیین ستاره مشهور از بین X و Y ۱۵۲
- شکل ۶-۳: درخت بازگشت حاصل از محاسبه جمله ششم دنباله فیوناچی..... ۱۵۶
- شکل ۶-۴: درخت بازگشت حاصل از محاسبه $\binom{6}{3}$ ۱۵۹
- شکل ۶-۵: تناظر مسئله‌های ضرب زنجیره ماتریس‌ها و مثلث بندی چندضلعی محدب..... ۱۷۶
- شکل ۷-۱: کلاس‌های پیشنهادی برای انتخاب..... ۱۹۶
- شکل ۷-۲: کلاس‌های پیشنهادی بعد از مرتب‌سازی بر اساس زمان پایان..... ۱۹۸
- شکل ۷-۳: تعداد تکرار حروف و کدهای آن‌ها در دو حالت با طول ثابت و متغیر..... ۲۰۶
- شکل ۷-۴: درخت‌های مربوط به هر دو روش کد با طول ثابت و متغیر..... ۲۰۷
- شکل ۷-۵: درخت حاصل از الگوریتم هافمن..... ۲۱۰
- شکل ۷-۶: تشریح مراحل اصلی در اثبات لم ۷-۳..... ۲۱۲
- شکل ۸-۱: دو نمایش استاندارد برای یک گراف بدون جهت..... ۲۱۹
- شکل ۸-۲: دو نمایش استاندارد برای یک گراف جهت‌دار..... ۲۱۹
- شکل ۸-۳: نحوه عملکرد مرتب‌سازی توپولوژیکی بر روی یک DAG..... ۲۲۸
- شکل ۸-۴: نحوه عملکرد الگوریتم پریم..... ۲۳۲
- شکل ۸-۵: نحوه عملکرد الگوریتم کروسکال..... ۲۳۷
- شکل ۸-۶: تأثیر وزن‌های منفی در گراف‌های جهت‌دار..... ۲۴۲
- شکل ۸-۷: یافتن مسیر کوتاه‌تر برای رأس u از طریق یال v به u ۲۴۳
- شکل ۸-۸: نحوه عملکرد الگوریتم بلمن - فورد..... ۲۴۵
- شکل ۸-۹: مقادیر آرایه‌های d و π در اجرای الگوریتم بلمن - فورد..... ۲۴۵
- شکل ۸-۱۰: نحوه عملکرد الگوریتم دیکسترا..... ۲۴۸
- شکل ۸-۱۱: مقادیر آرایه‌های d و π در اجرای الگوریتم دیکسترا..... ۲۴۸

- شکل ۸-۱۲: مقادیر آرایه‌های D و Π در اجرای الگوریتم فلوید-وارشال..... ۲۵۵
- شکل ۸-۱۳: نحوه وزن دهی مجدد گراف..... ۲۵۷
- شکل ۹-۱: درخت مربوط به مولد ترکیبات برای سه عدد..... ۲۶۹
- شکل ۹-۲: فراخوانی الگوریتم PERM برای مجموعه سه عنصری $A=\{a,b,c\}$ ۲۶۹
- شکل ۹-۳: نحوه عملکرد الگوریتم روش اول پس گرد برای چهار وزیر..... ۲۷۱
- شکل ۹-۴: درخت جستجوی مربوط به چهار وزیر..... ۲۷۳
- شکل ۹-۵: نحوه عملکرد الگوریتم روش سوم پس گرد برای چهار وزیر..... ۲۷۵
- شکل ۹-۶: ساختار کلی یک درخت بازی..... ۲۷۶
- شکل ۹-۷: بخشی از درخت بازی MIN-MAX مربوط به بازی X-O..... ۲۷۷
- شکل ۹-۸: نحوه هرس کردن شاخه‌های بیهوده..... ۲۷۸
- شکل ۹-۹: نمونه‌ای از یک هرس α - β ۲۷۹
- شکل ۹-۱۰: گراف متناظر با شهرها راه‌های مابین..... ۲۸۰
- شکل ۹-۱۱: درخت جستجوی مربوط به مسئله فروشنده دوره گرد..... ۲۸۲
- شکل ۹-۱۲: تور همیلتنی یافت شده با حداقل هزینه..... ۲۸۳
- شکل ۱۰-۱: طبقه بندی مجموعه مسائل..... ۲۹۵
- شکل ۱۰-۲: جایگاه مجموعه NP-کامل..... ۲۹۷

الگوریتم‌ها و نحوه تحلیل آن‌ها

در این فصل با مفاهیم زیر آشنا می‌شوید:

- آشنایی با مفهوم الگوریتم و روش‌های طراحی آن
- آشنایی با رفتار الگوریتم‌ها از نظر زمان اجرا و کارایی
- عوامل موثر بر زمان اجرا و نحوه محاسبه آن
- آشنایی با دو الگوریتم ساده جستجوی خطی و مرتب‌سازی درجی

۹-۱ تمرینات فصل اول

۱- نشان دهید که $\log n! \in \theta(n \log n)$ است؟

۲- ادعاهای زیر را اثبات یا رد کنید:

a) $f(n) = O(g(n)) \Rightarrow 2^{f(n)} = O(2^{g(n)})$

b) $f(n) = O(g(n)) \Rightarrow (f(n))^k = O(g(n)^k)$

۳- نشان دهید که اگر $f(n) = 5n-9$ باشد آنگاه $f(n) = O(n^2)$ خواهد بود؟

۴- فرض کنید که $f, g: \mathbb{N} \rightarrow \mathbb{R}^+$ و $\lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} = 1$ باشد. حال ثابت کنید که $f(n) \theta(g(n))$

۵- الگوریتم مرتب‌سازی درجی را برای مرتب‌سازی نزولی عناصر به جای صعودی بازنویسی کنید؟

۶- مرتب‌سازی درجی می‌تواند به شرح زیر به شکل بازگشتی بیان شود: برای مرتب‌سازی آرایه

$A[1 \dots n]$ به صورت بازگشتی $A[1 \dots n-1]$ را مرتب می‌کنیم سپس $A[n]$ را در آرایه مرتب

شده قبلی درج می‌کنیم. حال شبه کد معادل آن را نوشته و رابطه بازگشتی و زمان اجرای آن را

محاسبه کنید؟

۷- برای هر یک از جفت تابع داده شده تعیین کنید که کدام یک از پنج نماد مجانبی بین تابع اول

با دوم برقرار است؟

a) $\log^k n$, n^ϵ

b) n^k , c^k

c) \sqrt{n} , $n^{\sin \pi}$

d) 2^n , $2^{n/2}$

e) $n^{\lg c}$, $c^{\lg n}$

f) $\lg(n!)$, $\lg(n^n)$

۸- توابع داده شده زیر را به ترتیب افزایش رشد، کلاس بندی کنید، بدیهی است که دو تابع $f(n)$ و

$g(n)$ در کلاسی یکسان هستند اگر و فقط اگر $f(n) = \theta(g(n))$ باشد.

$\lg(\lg^* n)$	$2^{\lg^* n}$	$(\sqrt{2})^{\lg n}$	n^2	$n!$	$(\lg n)!$
$\left(\frac{3}{2}\right)^n$	n^3	$\lg^2 n$	$\lg(n!)$	2^{2^n}	$n^{1/\lg n}$
$\ln \ln n$	$\lg^* n$	n^{2^n}	$n^{\lg \lg n}$	$\ln n$	1
$2^{\lg n}$	$(\lg n)^{\lg n}$	e^n	$4^{\lg n}$	$(n+1)!$	$\sqrt{\lg n}$
$\lg^*(\lg n)$	$2^{\sqrt{2 \lg n}}$	N	2^n	$n \lg n$	$2^{2^{n+1}}$

۹- فرض کنید برای مرتب کردن یک آرایه n تایی به نام A بدین شکل عمل کنیم که ابتدا کوچک‌ترین عنصر را یافته و با عنصر $A[1]$ تعویض کنیم، سپس دومین عنصر کوچک‌تر را یافته و با $A[2]$ تعویض کنیم و این روند را برای $n-1$ عنصر اول آرایه ادامه دهیم. شبه کدی برای آن نوشته و در حالات مختلف زمان اجرای آن را با نماد θ بیان کنید؟ به نظر شما چرا لازم است این روند فقط برای $n-1$ عنصر اول آرایه و نه کل آرایه اجرا شود؟

۱۰- الگوریتم جستجوی خطی را طوری تغییر دهید که کارا کتری خاص را در یک رشته با n کارا کتر جستجو کند؟

۱۱- مسئله جمع دو عدد صحیح n بیتی را در نظر بگیرید که در دو آرایه n عنصری به نام‌های A و B ذخیره شده‌اند. جمع این دو عدد باید به شکل دودویی در آرایه ای $n+1$ عنصری به نام C ذخیره شود. شبه کدی برای این کار با زمان اجرای $\theta(n)$ ارائه دهید. درستی زمان اجرای الگوریتم خود را تحلیل کنید.

روابط بازگشتی و حل آنها

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با ساختار روابط بازگشتی
- آشنایی با هر یک از چهار روش اصلی حل روابط بازگشتی
- توانایی حل هر نوع رابطه بازگشتی با بکارگیری روش‌های این فصل
- آشنایی با برخی روابط ریاضی پر کاربرد در حل روابط بازگشتی

۲-۸ تمرینات فصل دوم

۱- روابط بازگشتی زیر را به روش حدس و استقرا حل کنید:

- a) $T(n) = T(n/2) + 1, T(1) = 1.$
- b) $T(n) = 7T(n/2), T(1) = 1.$
- c) $T(n) = 3T(n/2) + 5, T(1) = 2.$
- d) $T(n) = 9T(n/3) + n^2, T(1) = 2.$

۲- دنباله... ۴۷، ۲۹، ۱۸، ۱۱، ۷، ۴، ۳، ۱ به دنباله لوکا معروف است که در بین جملات آن رابطه

بازگشتی زیر برقرار است:

$$L_1 = 1, L_2 = 3, \forall n \geq 3: L_n = L_{n-1} + L_{n-2}$$

رابطه بازگشتی فوق را حل کنید.

۳- روابط بازگشتی زیر را به صورت دقیق حل کنید:

- a) $T(n) = 3T(n/3) + 1, T(1) = 2. (n = 3k)$
- b) $T(n) = 2T(n-2) + T(n-1), T(0) = 1, T(1) = 3.$
- c) $T(n) = 3T(n-3) + n, T(0) = 4. (n = 3k)$

۴- روابط بازگشتی زیر را با روشی دلخواه، حل کنید:

- a) $T(n) = T\left(\frac{9n}{10}\right) + n$
- b) $T(n) = 16T\left(\frac{n}{4}\right) + n^2$
- c) $T(n) = \sqrt{n}T(\sqrt{n}) + n$
- d) $T(n) = T(n-2) + T(n-3) + 2$
- e) $T(n) = T(\sqrt{n}) + 1$
- f) $T(n) = T(n-1) + T(n-3)$
- g) $T(n) = 3T\left(\frac{n}{3} + 5\right) + n/2$
- h) $T(n) = 5T\left(\frac{n}{5}\right) + n/\lg n$
- i) $T(n) = T\left(\frac{n}{2}\right) + T\left(\frac{n}{4}\right) + n$

۵- روابط بازگشتی زیر را با رسم درخت بازگشت، حل کنید:

- a) $T(n) = 3T\left(\frac{n}{2}\right) + n$
- b) $T(n) = T(n-1) + \lg n$
- c) $T(n) = T(n-2) + 2\lg n$
- d) $T(n) = T(n-1) + 1/n$
- e) $T(n) = 4T(n/5) + 1$

$$f) T(n) = 6T\left(\frac{n}{3}\right) + n^2$$

$$g) T(n) = 2T\left(\frac{n}{2}\right) + n^3$$

$$h) T(n) = 3T\left(\frac{n}{2}\right) + n \lg n$$

۶- رابطه بازگشتی $T(n) = T(n-1) + 2T(n-2) - 2T(n-3)$ را برای $n \geq 3$ را تا حد امکان به

صورت دقیق حل کنید اگر برای $0 \leq n \leq 2$ داشته باشیم که: $T(n) = 9n^2 - 15n + 106$.

۷- روابط بازگشتی زیر که به صورت نامساوی هستند، را حل کنید:

$$a) T(n) \geq 3T\left(\frac{n}{4}\right) + n$$

$$b) T(n) \leq T\left(\frac{n}{3}\right) + \sqrt{n}$$

$$c) T(n) \geq T(n-1) + n \lg n$$

$$d) T(n) \leq 2T\left(\frac{n}{4} + 3\right) + n^2$$

روش تقسیم و حل

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با ساختار طراحی الگوریتم به روش تقسیم و

حل

- آشنایی با نحوه تحلیل این نوع الگوریتم‌ها
- آشنایی و تحلیل الگوریتم‌های جستجوی دودویی، مرتب‌سازی ادغامی، مرتب‌سازی سریع، یافتن مرتبه آماری n ام، ضرب دو عدد n بیتی (ضرب اعداد بزرگ) و ضرب دو ماتریس به روش استراسن

۳-۹ تمرینات فصل سوم

- ۱- شبه کدی غیر بازگشتی برای الگوریتم جستجوی دودویی ارائه دهید. زمان اجرای آن را تحلیل کرده و با زمان اجرای شبه کد بازگشتی که در این بخش ۳-۱ آمده است مقایسه کنید.
- ۲- شبه کدی برای ضرب دو عدد n بیتی در حالت بهبود یافته، یعنی با زمان اجرایی کمتر از $\Theta(n^2)$ ارائه دهید.
- ۳- فرض کنید که در الگوریتم مرتب‌سازی سریع، آرایه با استفاده از زیر روال PARTITION به دو بخش c و $n-c$ (c عددی ثابت است) عنصری تقسیم شده باشد. ثابت کنید که در چنین حالتی، بدترین حالت اتفاق افتاده و زمان اجرای مرتب‌سازی سریع در این شرایط $\Theta(n^2)$ خواهد بود.
- ۴- فرض کنید آرایه‌ای با تعداد عناصر فرد n داشته باشیم. شبه کدی ارائه دهید که بزرگ‌ترین و کوچک‌ترین عنصر در این آرایه را با $(3n-3)/2$ مقایسه بدست آورد.
- ۵- نشان دهید که برای یافتن دومین بزرگ‌ترین عنصر در یک آرایه n عنصری ($n > 1$)، به $2 + \lg n + n$ مقایسه نیاز است.
- ۶- یک مسئله با استفاده از الگوریتم تقسیم و غلبه حل شده است. در این مسئله هر مسئله بزرگ به پنج مسئله کوچک‌تر شکسته شده و اندازه هر مسئله شکسته شده یک سوم مسئله بزرگ‌تر می‌باشد. الگوریتم شکستن و ترکیب دارای زمان $O(n)$ می‌باشند. رابطه بازگشتی این الگوریتم را نوشته و زمان اجرای آن را محاسبه کنید.
- ۷- آرایه n عنصری A را در نظر بگیرید فرض کنید $n = 2^k$ باشد. الگوریتم مرتب‌سازی درجی بر روی این آرایه در بدترین حالت چند مقایسه میان عناصر آرایه انجام می‌دهد؟ (ابتدا فرض کنید که $k=3$ باشد، سپس آن را تعمیم دهید)
- ۸- دو آرایه مرتب $x[1..n]$ و $y[1..n]$ را در نظر بگیرید. سریع‌ترین الگوریتم ممکن برای یافتن میانه بین $2n$ عضو آرایه‌های x, y را پیدا کنید. زمان اجرای الگوریتم خود را تحلیل کنید.
- ۹- در باشگاهی قرار است مسابقات والیبال برگزار شود به این ترتیب که بین $n=2^k$ تیم شرکت کننده اول $n/2$ مسابقه صورت می‌گیرد، سپس بین تیم‌های برنده و نیز بین تیم‌های بازنده دوباره به همین شکل مسابقاتی برگزار می‌شود (در هر مرحله هر تیم برنده با یک تیم برنده دیگر از همان

روش تقسیم و حل / ۱۰۷

مرحله مسابقه می‌دهد و همین‌طور برای تیم‌های بازنده آن مرحله). در هر مرحله مسابقات به همین منوال ادامه پیدا می‌کند تا زمانی که به گروه‌های یک تیمی برسیم. تعداد کل بازی‌ها برای $n = 128$ چند خواهد بود؟

۱۰- پنج فایل مرتب شده به اندازه‌های ۵، ۱۰، ۲۰، ۲۵، ۳۰ مفروضند. می‌خواهیم از ادغام دو به دوی آن‌ها یک فایل مرتب شده واحد شامل همه رکوردها به دست آوریم. در هر ادغام رکوردهای فایل‌های ورودی ممکن است چند بار از یک فایل خوانده و در یک فایل دیگر نوشته شوند. به هر کدام از این نوشتن و خواندن یک جابه‌جایی می‌گوییم. حداقل تعداد کل این جابه‌جایی‌ها برای ادغام همه فایل‌ها چقدر است؟

۱۱- در ضرب ماتریس‌ها به روش استراسن اگر مسئله کوچک ضرب ماتریس‌های 2×2 باشد برای ضرب دو ماتریس 8×8 چند ضرب عددی صورت می‌گیرد؟

۱۲- اگر در ماتریس 4×4 با روش ضرب استراسن در یکدیگر ضرب شوند برای ضرب این دو ماتریس چند ضرب عددی صورت می‌گیرد؟

۱۳- فرض کنید که ما فقط ضرب اعداد یک رقمی را می‌توانیم انجام دهیم و همچنین فرض کنید ضرب در اعداد 10^x با عمل شیفت انجام میشود. برای ضرب دو عدد 5618×1234 حداقل تعداد عمل ضرب چقدر خواهد بود؟

۱۴- در گونه جدیدی از مرتب‌سازی سریع، برای انتخاب محور از میان n عنصر، $2\sqrt{n} + 1$ عنصر اول آرایه را انتخاب می‌کنیم و با یک الگوریتم ساده مانند مرتب‌سازی درجی آن‌ها را مرتب می‌کنیم. عنصر میانه این تعداد عنصر مرتب را با عنوان محور الگوریتم انتخاب می‌کنیم. بقیه الگوریتم مانند قبل عمل میکند. رابطه بازگشتی و زمان اجرای این الگوریتم را در بدترین حالت بیابید؟

۱۵- آرایه $A[1..n]$ را آرایه‌ای از n عدد غیر تکراری در نظر بگیرید. اگر $i < j$ باشد و $A[i] > A[j]$ باشد آنگاه زوج (i, j) را یک وارونگی در آرایه A می‌نامیم. الگوریتمی ارائه دهید که تعداد این وارونگی‌ها را در یک آرایه n عنصری را در بدترین حالت با زمان اجرای $\Theta(n \lg n)$ بیابد.

۱۶- اگر در الگوریتم مرتب‌سازی سریع، همه عناصر آرایه مساوی باشند، زمان اجرا این الگوریتم چقدر است؟

۱۷- الگوریتمی با زمان اجرای $\Theta(n \lg n)$ ارائه دهید، به طوری که بتواند در یک مجموعه n عنصری S ، دو عددی را بیابد که مجموع آن‌ها برابر x باشد.

۱۸- فرض کنید که $T[1..n]$ آرایه‌ای مرتب و n تایی از اعداد صحیح متمایز باشد. الگوریتمی به روش تقسیم و حل ارائه دهید که اندیسی مانند i را در صورت وجود در این آرایه با زمان اجرای $\Theta(n)$ بیابید به طوری که $T[i]=i$ باشد؟

۱۹- یک ماتریس n در n را در نظر بگیرید که هر سطر آن از چپ به راست و هر ستون آن از بالا به پایین به صورت صعودی مرتب است. الگوریتمی به روش تقسیم و حل ارائه دهید که عددی مانند x را با زمان اجرای $\Theta(n)$ در این ماتریس بیابید.

۲۰- فرض کنید که $A[1..n]$ آرایه‌ای مرتب و n تایی از اعداد صحیح متمایز باشد. در صورتی که $n=32$ باشد، حداکثر تعداد مقایسه در الگوریتم جستجوی دودویی چه تعداد خواهد بود؟

۲۱- فرض کنید آرایه‌ای n تایی از اعداد مختلف داریم، الگوریتمی با زمان اجرای $O(n)$ ارائه دهید که در صورت وجود عددی که تعداد تکرارش از $\lfloor n/2 \rfloor$ بیشتر است را بیابد؟

۲۲- الگوریتمی غیر بازگشتی برای مرتب‌سازی ادغامی ارائه دهید. زمان اجرای آن را تحلیل کنید؟

۲۳- الگوریتمی بنویسید که یک آرایه n عنصری را با تقسیم به سه زیر آرایه تقریباً $n/3$ عنصری مرتب نماید. این الگوریتم هر یک از زیر آرایه‌ها را به صورت بازگشتی مرتب کرده و در نهایت با هم ادغام می‌کند. شبه کد چنین الگوریتمی را نوشته و زمان اجرای آن را تحلیل کنید؟

۲۴- با استفاده از مرتب‌سازی سریع، آرایه ۸ عنصری زیر را مرتب کنید:

3	4	12	1	9	11	5	8
---	---	----	---	---	----	---	---

۲۵- مسئله برج‌های هانوی^۱ را در نظر بگیرید. مسئله برج‌های هانوی شامل سه میله و n دیسک با اندازه‌های متفاوت است. هدف جا به جا کردن دیسک‌هایی از یک میله به میله‌ای دیگر با استفاده از میله کمکی سوم است به طوری که دیسک‌ها به ترتیب اندازه در میله قرار دارند. مسئله باید با توجه به قوانین زیر حل شود:

- وقتی دیسکی حرکت داده می‌شود، باید روی یکی از سه میله قرار گیرد.
- در هر مرتبه فقط یک دیسک و آن هم دیسک بالایی میله را می‌توان حرکت داد.
- هیچ‌گاه دیسک بزرگ‌تر روی دیسک کوچک‌تر قرار نمی‌گیرد.

روش تقسیم و حل / ۱۰۹

حال با این توضیحات:

- الف) یک الگوریتم تقسیم و حل برای این مسئله ارائه دهید.
- ب) نشان دهید که برای الگوریتم شما تعداد جابه‌جایی‌های مورد نیاز برای دیسک‌ها برابر $2^n - 1$ است.
- ج) ثابت کنید که هر الگوریتم دیگری نیز حداقل به $2^n - 1$ جابه‌جایی برای انجام این کار نیاز دارد.

مرتب سازی هرمی و صف اولویت

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با درخت Heap و انواع آن
- آشنایی با مرتب سازی هرمی و تحلیل آن
- آشنایی با صغف اولویت

۴-۵ تمرینات فصل چهارم

۱- با توجه به آنکه زمان مورد نیاز برای زیر روال MAX-HEAPIFY هنگامی که برای گره ای با ارتفاع h فراخوانی می‌شود، برابر $O(h)$ است، نشان دهید هزینه روال BUILD-MAX-HEAP برابر $O(n)$ خواهد بود؟

۲- فرض کنید که A یک آرایه دلخواه با ۱۲ عنصر باشد. چند مقایسه حداکثر نیاز می‌باشد تا این آرایه تبدیل به یک Min-heap شود؟

۳- با استفاده از Heap الگوریتمی با زمان اجرای $O(n \lg k)$ برای ادغام k لیست مرتب شده در یک لیست مرتب ارائه دهید که در آن n تعداد کل عناصر در همه لیست‌های ورودی است.

۴- فرض کنید که آرایه T با هفت عنصر به صورت زیر باشد. به صورت مرحله به مرحله با استفاده از مرتب‌سازی هرمی، این آرایه را مرتب کنید.

5	8	12	6	3	9	15
---	---	----	---	---	---	----

۵- یک Heap دتایی مانند یک Heap دودویی عادی است با این تفاوت که گره‌های غیر برگ به جای دو فرزند دارای d فرزند هستند.

الف) توضیح دهید که چگونه می‌توان یک Heap، دتایی را در یک آرایه پیاده‌سازی کرد.

ب) ارتفاع یک Heap، دتایی با n عنصر را بر حسب n و d بیابید.

ج) پیاده‌سازی مناسبی از EXTRACT-MAX برای Max-heap، دتایی ارائه دهید. زمان اجرای آن را بر حسب n و d بیان کنید.

د) پیاده‌سازی مناسبی از INSERT برای Max-heap، دتایی ارائه دهید. زمان اجرای آن را بر حسب n و d بیان کنید.

ه) پیاده‌سازی مناسبی از INCREASE-KEY(A, i, k) برای Max-heap، دتایی ارائه دهید، به طوری که در ابتدا مقدار $\max(A[i], k)$ در $A[i]$ قرار گرفته، سپس ساختار Max-heap، دتایی به شکل مناسبی بازسازی می‌شود. زمان اجرای آن را بر حسب n و d بیان کنید.

۶- فرض کنید که تابع $\text{Heap-Delete}(A,i)$ عنصر گره i نام را از Heap به نام A حذف می‌کند. شبه کدی برای این کار ارائه دهید به طوری که برای یک Max-heap با n عنصر، زمان اجرای آن $O(\lg n)$ باشد.

۷- نشان دهید که در هر Heap با n عنصر، حداکثر $\left\lceil \left(\frac{n}{2}\right)^{h+1} \right\rceil$ گره با ارتفاع h وجود دارد.

مرتب سازی های خطی

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با سه الگوریتم مرتب سازی خطی، مرتب سازی شمارشی، مرتب سازی سطلی، مرتب سازی مبنایی
- آشنایی با شرایط خاص مربوط به هر یک از الگوریتم ها مرتب سازی خطی
- آشنایی با تحلیل این الگوریتم ها و مقایسه کلی همه الگوریتم های مرتب سازی ذکر شده

۵-۵ تمرینات فصل پنجم

۱- نشان دهید که الگوریتم مرتب‌سازی شمارشی پایدار است؟
 ۲- مرتب‌سازی شمارشی را به صورت مرحله به مرحله بر روی آرایه $A = \{5, 3, 1, 2, 3, 4, 1, 5, 2, 6, 2\}$ اجرا کنید.

۳- در الگوریتم مرتب‌سازی شمارشی، اگر حلقه for آخر را بجای حالت نزولی $(for\ j \leftarrow n\ downto\ 1)$ به صورت صعودی $(for\ j \leftarrow 1\ to\ n)$ بنویسیم، آیا همچنان الگوریتم درست کار می‌کند؟ آیا الگوریتم تغییر یافته پایدار است؟

۴- نشان دهید که چطور می‌توان n عدد صحیح در بازه 0 تا $n^2 - 1$ را در زمان $O(n)$ مرتب کرد.
 ۵- مرتب‌سازی سطلی را به صورت مرحله به مرحله بر روی آرایه $A = \{15, 23, 83, 27, 13, 74, 19, 29, 72, 26, 81\}$ اجرا کنید.

۶- زمان اجرای الگوریتم مرتب‌سازی سطلی در بدترین حالت چقدر است؟ چه تغییر کوچکی در این الگوریتم علاوه بر حفظ زمان اجرای خطی مورد نظر، در بدترین حالت زمان اجرای آن برابر $O(n \lg n)$ می‌گردد؟

۷- فرض کنید که به جای مرتب‌سازی یک آرایه، فقط لازم است که عناصر به طور میانگین افزایش یابند. به طور دقیق‌تر آرایه A با n عنصر را k مرتب می‌نامیم اگر برای تمام $i = 1, 2, 3, \dots, n-k$ رابطه زیر برقرار باشد:

$$\frac{\sum_{j=i}^{i+k-1} A[j]}{k} \leq \frac{\sum_{j=i+1}^{i+k} A[j]}{k}$$

الف) اگر آرایه‌ای 1 مرتب باشد، به چه معناست؟

ب) یک جایگشت از اعداد یک تا ده ارائه دهید که 2 مرتب باشد، ولی مرتب شده نباشد؟

ج) ثابت کنید که یک آرایه n عنصری k مرتب است اگر و فقط اگر برای تمام

$$A[i] \leq A[i+k], \quad i = 1, 2, 3, \dots, n-k$$

د) الگوریتمی ارائه دهید که یک آرایه n عنصری را در زمان $O(n \lg(n/k))$ ، k مرتب کند؟

ه) نشان دهید که یک آرایه k مرتب با تعداد عناصر n را می‌تواند در زمان $O(n \lg k)$

مرتب نمود؟

و) نشان دهید هنگامی که k ثابت باشد، زمان لازم برای k مرتب کردن یک آرایه n

عنصری برابر $\Omega(\lg n)$ است؟

۸- چنان که گفته شد، در مرتب‌سازی مبنایی به صورت یک رقم، یک رقم، مرتب‌سازی بر روی اعداد انجام می‌شود. حال فرض کنید که نمی‌خواهیم به صورت رقم به رقم این کار را انجام دهیم.

فرض کنید که n عدد b بیتی داریم که بجای یک رقم یک رقم، می‌خواهیم r بیت، r بیت مرتب‌سازی را انجام دهیم. یعنی هر عدد به b/r دسته r بیتی تقسیم می‌شود. تحلیل کنید که مقدار

r را چگونه باید تعیین کنیم که خللی در خطی بودن زمان اجرا مرتب‌سازی مبنایی ایجاد نکند؟

۹- الگوریتمی ارائه دهید که با زمان اجرای $O(n)$ عنصر میانه در یک آرایه n عنصری را بیابد؟

روش‌های مبتنی بر استقرا و برنامه ریزی پویا

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با روش مبتنی بر استقرا و موارد کاربرد آن
- آشنایی با تفاوت ما بین روش تقسیم و حل و مبتنی بر استقرا
- آشنایی با روش برنامه‌ریزی پویا
- آشنایی با مزایای استفاده از روش برنامه‌ریزی پویا و موارد کاربرد آن
- آشنایی با مثال‌هایی از روش برنامه‌ریزی پویا مانند: یافتن n امین جمله فیبوناچی، ترکیب k از n ضرب زنجیره‌ای ماتریس‌ها، بزرگ‌ترین زیر دنباله مشترک، مثلث‌بندی چند ضلعی محدب، مساله کوله‌پشتی

۴-۶ تمرینات فصل ششم

۱- ثابت کنید که رابطه زیر به ازای هر $n, k \in \mathbb{N}$ ($n \geq k$) همیشه درست است؟

$$\binom{n}{k} = \binom{n-1}{k} + \binom{n-1}{k-1}$$

۲- ثابت کنید که تعداد فراخوانی‌های بازگشتی انجام شده در الگوریتم ترکیب k عنصر از n عنصر برابر $2 - 2\binom{n}{k}$ خواهد بود؟

۳- شبه کد الگوریتم یافتن n امین جمله فیبوناچی را چنان تغییر دهید که بدون ایجاد خللی در زمان اجرای آن، حافظه مصرفی به $\Theta(1)$ کاهش یابد؟

۴- شبه کد الگوریتم ترکیب k عنصر از n عنصر را چنان تغییر دهید که بدون ایجاد خللی در زمان اجرای آن، حافظه مصرفی این الگوریتم به $\Theta(k)$ کاهش یابد؟

۵- نشان دهید که زمان اجرای زیر روال PRINT-PARENS در بدترین حالت برابر $\Theta(n)$ است؟

۶- یک پراتز گذاری بهینه برای زنجیره شش ماتریس زیر بیابید:

$$A_{4 \times 2} B_{2 \times 3} C_{3 \times 1} D_{1 \times 5} E_{5 \times 3} F_{3 \times 7}$$

۷- بزرگ‌ترین زیر دنباله مشترک بین دو رشته زیر را بیابید:

$$X = AABAACAACBB$$

$$Y = ABBCACCABAA$$

۸- فرض کنید می‌خواهیم مقدار چند جمله‌ای $P_n(x)$ از درجه n را بر حسب x محاسبه کنیم. الگوریتمی به روش مبتنی بر استقرا ارائه دهید که این عمل را در زمان $\theta(n)$ انجام دهد. (این روش به نام الگوریتم هورنر^۱ معروف است)

۹- با توجه به شبه کد مربوط به ضرب زنجیره ماتریس‌ها، شبه کدی مناسب برای مسئله مثلث بندی بهینه یک چند ضلعی محدب به روش برنامه‌ریزی پویا ارائه داده و زمان اجرای آن را محاسبه کنید؟

۱۰- با توجه به شبه کد مربوط به مسئله کوله‌پشتی در حالتی که عناصر هم ارزش هستند، شبه کدی مناسب برای مسئله کوله‌پشتی در حالتی که عناصر دارای ارزش متفاوت هستند به روش برنامه‌ریزی پویا ارائه داده و زمان اجرای آن را محاسبه کنید؟

۱۱- فرض کنید اجناس زیر همراه با وزن و ارزش هایشان موجود باشند و وزن قابل تحمل کوله پشتی برابر

۳۵ کیلو باشد. مشخص کنید که کدام اجناس را باید برداریم تا بیشترین سود را بدست آوریم؟

$$W_1 = 5 \text{ kg}, W_2 = 10 \text{ kg}, W_3 = 12 \text{ kg}, W_4 = 13 \text{ kg}, W_5 = 15 \text{ kg}, W_6 = 20 \text{ kg}$$

$$V_1 = \$50, V_2 = \$60, V_3 = \$75, V_4 = \$85, V_5 = \$90, V_6 = \$100$$

۱۲- فرض کنید که مجموعه اجناس که همگی ارزشی یکسان دارند به شکل $A = \{8,5,7,6,9,16\}$

باشد. اگر وزن قابل تحمل کوله پشتی برابر $k=18$ کیلو باشد، مشخص کنید که:

الف) آیا می توان چنین کوله ای را پر کرد یا خیر؟

ب) اگر پاسخ قسمت الف مثبت است با کدام اجناس و در صورت منفی بودن حداکثر تا چه

میزان و با کدام اجناس می توان کوله را پر کرد؟

ج) حال فرض کنید که اگر از اجناس ۵ و ۹ کیلویی ۲ تا وجود داشته باشد. با این فرض

قسمت های الف و ب را حل کنید.

۱۳- مسئله چاپ منظم یک پاراگراف در یک چاپگر را در نظر بگیرید. متن ورودی، رشته ای از n

کلمه با طول های L_1, L_2, \dots, L_n بر حسب تعداد کاراکتر می باشد. می خواهیم طوری این پاراگراف

را به طور منظم در تعدادی خط چاپ کنیم که هر یک حداکثر M کاراکتر را در بر گیرد. معیار

منظم بودن چنین است که اگر خطی شامل کلمات i تا j ($i \leq j$) باشد و بین کلمات یک فاصله

داشته باشیم، تعداد کاراکترهای فاصله اضافی در آخر خط برابر:

$$M - j + i - \sum_{k=i}^j L_k$$

است که باید مقداری نامنفی باشد تا کلمات در خط جای گیرند. حال می خواهیم مجموع مربعات

تعداد کاراکترهای فاصله برای همه خطوط بجز خط آخر، مینیمم شود. الگوریتمی به روش

برنامه ریزی پویا ارائه دهید که پاراگرافی با n کلمه را به طور منظم چاپ کند. زمان اجرا و حافظه

مورد نیاز برای الگوریتم خود را تحلیل کنید.

۱۴- فرض کنید که n عنصر متمایز و یک پشته (stack) در اختیار داریم. حال بیان کنید که چند

روش متمایز برای وارد کردن این n عنصر به درون پشته و خارج کردن آن ها از پشته وجود دارد؟

۱۵- فرض کنید برای تبدیل رشته a به رشته b امکان انجام اعمال زیر را داریم:

- افزودن کاراکتری در محلی دلخواه از رشته
- حذف کاراکتری دلخواه از رشته
- تعویض کاراکتری از رشته با کاراکتری دیگر.

روش‌های مبتنی بر استقرا و برنامه‌ریزی پویا / ۱۸۵

حال الگوریتمی به روش تقسیم و حل ارائه دهید که با حداقل تعداد ممکن استفاده از این سه عمل، رشته ورودی a را به رشته دیگری مانند b که لزوماً دارای طول (تعداد کاراکتر) یکسانی هم نیستند، تبدیل کند. زمان اجرا و حافظه مورد نیاز برای الگوریتم خود را تحلیل کنید.

۱۶- مسئله زمان بندی کلاس‌ها (انتخاب فعالیت‌ها) را که در بخش ۷-۲ توضیح داده شده است را به روش برنامه‌ریزی پویا حل کنید. زمان اجرا و حافظه مورد نیاز برای الگوریتم خود را تحلیل کنید.

۱۷- تساوی زیر را اثبات کنید:

$$\sum_{diagonal=1}^{n-1} [(n - diagonal) \times diagonal] = \frac{n(n-1)(n+1)}{6}$$

روش حرिवانه

در این فصل با مفاهيم زير آشنا مي شويد:

- آشنائي با مفهوم و روش طراحي الگوريتم‌هاي حرिवانه
- آشنائي با ساختار كلي و بخش‌هاي مربوط به الگوريتم‌هاي حرिवانه
- آشنائي با برخي کاربردها و مثال‌ها از روش طراحي الگوريتم حرिवانه
- آشنائي با نحوه اثبات درستي روش ارائه شده به وسيله الگوريتم حرिवانه

۷-۷ تمرینات فصل هفتم

۱- ثابت کنید که الگوریتم حریصانه ارائه شده برای کوله‌پشتی کسری بهینه بوده و هیچ گونه ظرفیتی از کوله‌پشتی هدر نمی‌رود؟

۲- سکه‌های زیر موجودند، با استفاده از الگوریتم خرد کردن پول نشان دهید که چگونه با حداقل تعداد سکه‌ها بقیه پول مشتری را که ۴۷۰ ریال است بپردازیم؟

ارزش سکه	۵۰۰ریالی	۲۵۰ریالی	۱۰۰ریالی	۵۰ریالی	۱۰ریالی
تعداد سکه	۱	۱	۱	۱	۱۰

۳- اشیا زیر را در نظر بگیرید. ظرفیت کوله‌پشتی برابر با ۴۰ است، جواب بهینه‌ای برای این کوله‌پشتی کسری بیابید؟

X_i	p_i	w_i
X_1	8	16
X_2	5	15
X_3	15	25
X_4	10	8
X_5	20	15

۴- اگر تعداد تکرارهای حروف A تا H به صورت زیر باشد، با استفاده از الگوریتم هافمن، رشته

A: 5 B: 2 C: 4 D: 7 E: 6 F: 12 G: 8 H: 13

زیر را رمزگشایی کنید.

رشته: 10111111011011111111111111011110

۵- اگر تعداد تکرارهای هشت حرف اول الفبا بر مبنای هشت عدد اول دنباله فیبوناچی باشد، کد هافمن معادل این حروف را بدست آورید. آیا می‌توانید پاسخ خود را بجای هشت حرف به n حرف تعمیم دهید؟

۶- الگوریتم هافمن را برای کلمه کدهای سه سه‌ای (مبنای سه) تعمیم دهید (یعنی کلمه کدها فقط شامل 0,1,2 هستند). ثابت کنید که یک کد سه سه‌ای بهینه حاصل می‌شود؟

۷- یک الگوریتم بهینه و موثر ارائه دهید که از بین مجموعه نقاط داده شده $\{X_1, X_2, X_3, \dots, X_n\}$ که روی یک خط راست قرار دارند، کوچک‌ترین مجموعه بازه‌های بسته با طول واحد که شامل همه نقاط داده شده باشد، را تعیین کند؟ درستی الگوریتم خود را ثابت کنید.

روش حریصانه / ۲۱۵

۸- فرض کنید کارها، مهلت‌ها و سودهای زیر داده شده است، مجموعه بهینه کارها را پیدا کنید.

کار	مهلت	سود
۱	۳	۳۱
۲	۲	۴۳
۳	۱	۲۵
۴	۱	۲۹
۵	۳	۳۴
۶	۲	۳۷
۷	۱	۳۵

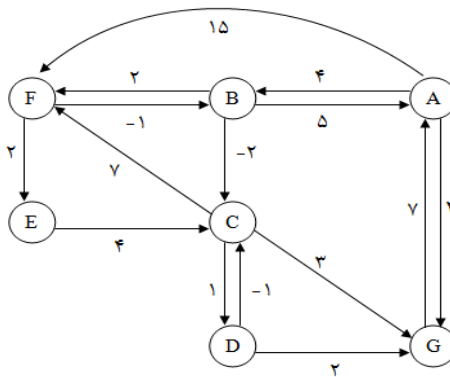
الگوریتم‌های گراف

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با دو روش نمایش و پیاده‌سازی گراف‌ها در حافظه
- آشنایی با دو روش جستجوی گراف‌ها یعنی جستجوی اول عمق و اول سطح
- آشنایی با نحوه مرتب‌سازی توپولوژیکی
- آشنایی با الگوریتم‌های مختلف یافتن درخت پوشای کمینه و تحلیل آن‌ها
- آشنایی با مساله یافتن کوتاه‌ترین مسیر در حالات مختلف

۸-۸ تمرینات فصل هشتم

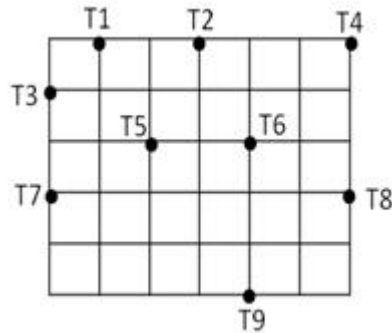
- ۱- مشابه اثباتی که برای الگوریتم پریم ارائه شد، درستی الگوریتم کروسکال را نیز اثبات کنید.
- ۲- فرض کنید که ماتریس مجاورت گراف غیر جهت دار G ماتریس b باشد، حال بررسی کنید که درایه‌های ماتریس $b \cdot b^t$ نشان دهنده چه چیزی هستند؟
- ۳- فرض کنید که T درخت پوشای کمینه برای یک گراف $G=(V,E)$ باشد. اگر وزن یک یال $e \in E$ را کاهش دهیم، با چه مرتبه اجرایی می‌توان درخت پوشای کمینه جدید را به دست آورد؟
- ۴- یک گراف متراکم شامل n گره مفروض است. هرگاه بخواهیم با استفاده از الگوریتم کروسکال درخت پوشای کمینه این گراف را به دست آوریم، در این صورت پیچیدگی زمانی این الگوریتم چقدر خواهد بود؟



- ۵- با استفاده از الگوریتم بلمن-فوردم کوتاه‌ترین مسیرها را از مبدأ A در گراف زیر را بدست آورید. آیا می‌توان این مسئله را با الگوریتم دیکسترا حل کرد؟
- ۶- در گراف تمرین قبل، وزن سه یالی که وزن منفی دارند را مثبت در نظر گرفته، سپس با استفاده از الگوریتم دیکسترا کوتاه‌ترین مسیرها را از مبدأ A را بدست آورید. با توجه به تعداد یال‌ها و رأس‌ها که در گراف می‌بینید، زمان اجرای الگوریتم‌های بلمن-فوردم و دیکسترا را درباره این گراف با هم مقایسه کنید.
- ۷- یک مهندس برق مداری را به شکل زیر طراحی نموده است که دارای ۹ ترمینال می‌باشد که باید ولتاژی معادل ۵ ولت به آن‌ها متصل گردد. فرض کنید برق ۵ ولت به یکی از ترمینال‌ها وصل

الگوریتم‌های گراف / ۲۶۱

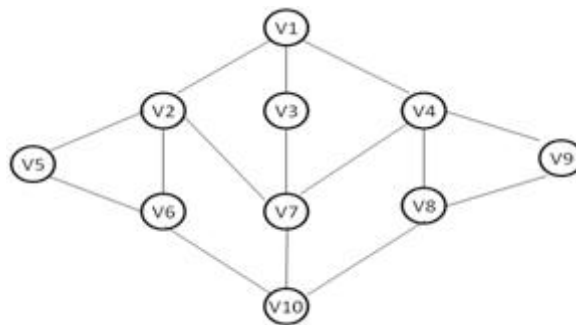
است برای اینکه کم‌ترین سیم بندی در مدار به کار رود، محاسبه کنید که حداقل طول سیم مورد نیاز چقدر است؟ (فاصله هر سطر و ستون را یک سانتی متر فرض کنید)



۸- فرض کنید که مسافت مابین ۵ شهر به شرح زیر است. اگر به ازای هر ۱۰۰ کیلومتر طی مسیر ۶ لیتر بنزین مصرف شود، با ذکر الگوریتم کم‌ترین میزان بنزین مورد نیاز برای رفتن از هر یک از شهرها به سایر شهرها را بدست آورد. توجه داشته باشید که جاده‌ها به صورت یک طرفه هستند

- شهر ۱ به شهر ۲: ۲۵۰ کیلومتر، شهر ۱ به شهر ۴: ۱۰۰ کیلومتر، شهر ۱ به شهر ۵: ۳۵۰ کیلومتر
- شهر ۲ به شهر ۳: ۲۰۰ کیلومتر، شهر ۲ به شهر ۴: ۵۰ کیلومتر، شهر ۲ به شهر ۵: ۵۰ کیلومتر
- شهر ۴ به شهر ۲: ۱۰۰ کیلومتر، شهر ۵ به شهر ۱: ۱۵۰ کیلومتر، شهر ۵ به شهر ۳: ۱۰۰ کیلومتر

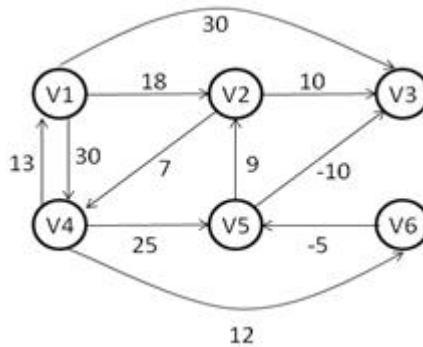
۹- حاصل اجرای الگوریتم‌های جستجوی اول عمق و جستجوی اول سطح را بر روی گراف زیر با ریشه V1 را بدست آورید؟



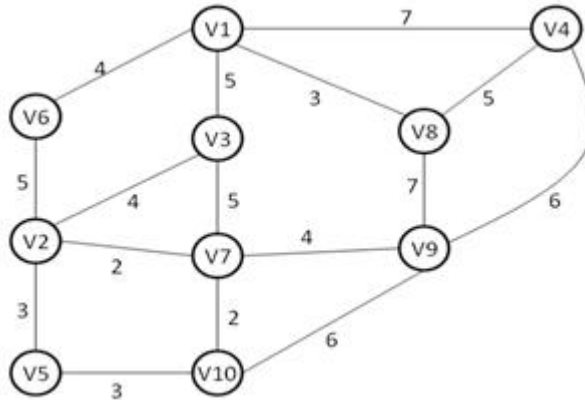
۱۰- برای گراف سوال قبل، لیست هم‌جواری و ماتریس مجاورت معادل آن را محاسبه کنید؟

۱۱- گراف جهت‌دار زیر را در نظر بگیرید:

- الف) با استفاده از الگوریتم جانسون، کوتاه‌ترین مسیر بین هر جفت رأس را بیابید؟
 ب) با استفاده از الگوریتم فلویید-وارشال، کوتاه‌ترین مسیر بین هر جفت رأس را بیابید؟
 ج) آیا جواب‌های بدست آمده در قسمت‌های الف و ب یکسان است؟ کدام یک از این الگوریتم‌ها برای این گراف مناسب‌تر است؟
 د) با در نظر نگرفتن یال‌های $\langle V4, V1 \rangle$ و $\langle V5, V2 \rangle$ و همچنین وزن یال‌ها، الگوریتم مرتب‌سازی توپولوژیکی را روش گراف حاصل انجام دهید.



- ۱۲- در گراف بدون جهت زیر، یک بار با الگوریتم کروسکال و یک بار با الگوریتم پریم، درخت پوشای کمینه آن را بیابید؟



جستجوی فضای حالت

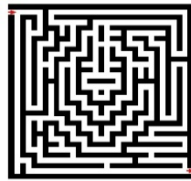
در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با انواع روش های جستجوی فضای حالات
- آشنایی با ساختار روش پس گرد و مثال هایی از آن و نحوه تحلیل هر یک
- آشنایی با نحوه استفاده از درخت بازی در حل مسائل
- آشنایی با روش محدود کردن فضای حالات و انواع آن
- بررسی مسئله فروشنده دوره گرد

۹-۵ تمرینات فصل نهم

۱- برای حل مسئله کوله پستی صفر و یک، که در آن n عنصر با وزن و ارزش‌های مثبت وجود داشته و وزن قابل تحمل کوله پستی به اندازه W است، راه حلی به روش پس‌گرد ارائه دهید؟

۲- یک جعبه بازی مشابه آنچه که در شکل زیر می‌بینید و شامل مسیرهایی پر پیچ و خم^۱ است را در نظر بگیرید، به طوری که یک ورودی وجود داشته که بازیکن باید از آن شروع کرده و با یافتن مسیر درست به خروجی برسد. دقت داشته باشید که همیشه فقط یک خروجی وجود دارد. حال الگوریتمی به روش پس‌گرد ارائه دهید که این مسئله را حل کند. این مسئله با نام مسئله مسیرهای پر پیچ و خم دوبعدی^۲ مشهور است.



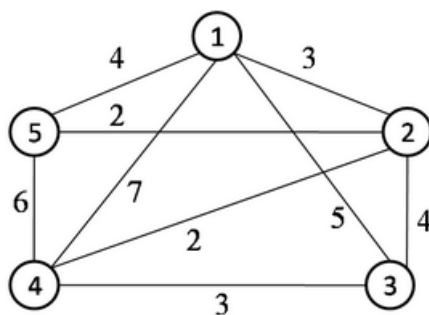
۳- مسئله m رنگ را در نظر بگیرید، در این مسئله می‌خواهیم تمام راه‌های رنگ آمیزی یک گراف بدون جهت را با استفاده از حداکثر m رنگ مختلف بیابیم، به طوری که هیچ دو گره مجاور دارای رنگی یکسان نباشند. الگوریتمی به روش پس‌گرد برای این کار ارائه دهید. این مسئله با نام مسئله رنگ آمیزی گراف^۳ مشهور است.

۴- الگوریتمی به روش انشعاب و تحدید برای مسئله رنگ آمیزی گراف با m رنگ ارائه دهید. سپس کارائی آن نسبت به الگوریتم ارائه داده شده برای این مسئله به روش پس‌گرد در تمرین قبل را بررسی کنید.

۵- الگوریتمی به روش پس‌گرد برای حل مسئله مجموع زیر مجموعه‌ها^۴ ارائه داده و آن را تحلیل کنید. منظور از مسئله مجموع زیر مجموعه‌ها آن است که n عدد صحیح مثبت وجود دارد که معمولاً به آن‌ها وزن می‌گویند و با w_i نشان داده می‌شوند، هدف آن است که تمام زیرمجموعه‌هایی از این اعداد که مجموع آن‌ها برابر W است را بیابیم.

1 - Maze
 2 - 2D Maze
 3 - Graph Coloring
 4 - Sum of Subset

۶- مسئله فروشنده دوره گرد را بر روی گراف زیر حل کنید؟



پیچیدگی محاسباتی و نظریه NP

در این فصل با مفاهیم زیر آشنا می شوید:

- آشنایی با دسته‌بندی‌های مختلف الگوریتم‌ها
- آشنایی با مفهوم الگوریتم‌های کنترل‌ناپذیر
- آشنایی با مفهوم مسائل P
- آشنایی با مفهوم نظریه NP
- آشنایی با انواع مختلف نظریه NP

۱۰-۴ تمرینات فصل دهم

- ۱- نشان دهید که یک مسئله، NP-آسان است اگر و فقط اگر به یک مسئله NP-کامل کاهش یابد؟
- ۲- اثبات کنید که اگر مسئله تصمیم گیری مانند B در مجموعه P بوده و $A \propto B$ باشد، آنگاه مسئله تصمیم گیری A نیز در P خواهد بود؟
- ۳- مسئله محاسبه nامین جمله فیبوناچی، جزو کدام دسته از سه نوع مسائل کنترل ناپذیر می باشد؟ پاسخ خود را توضیح دهید.
- ۴- فرض کنید که مسئله های A و B، دو مسئله مختلف از مسائل تصمیم گیری باشند. همچنین فرض کنید که مسئله A، کاهش پذیر چند به یک زمان چند جمله ای به مسئله B باشد. اگر مسئله A، NP-کامل باشد، آیا مسئله B نیز NP-کامل است؟ توضیح دهید.
- ۵- نشان دهید که اگر یک مسئله NP نباشد، NP-آسان هم نیست. بنابراین محاسبات Presburger و مسئله توقف نیز NP-آسان نمی باشند.
- ۶- رابطه بین مسائل از نوع P، NP-Hard و NP-Complete را مشخص کنید.

منابع و مراجع انگلیسی

- 1- Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein. "Introduction to Algorithms", Third Edition, MIT Press and McGraw-Hill Book Company, 2009.
- 2- Richard Neapolitan and Kumarss Naimipour. "Foundations Of Algorithms Using C++ Pseudo code", Third Edition, Jones and Bartlett Computer Science, 2003.
- 3- Gilles Brassard and Paul Bratley. "Fundamentals of Algorithmics", Prentice Hall, 1996
- 4- Michael R. Garey and David S. Johnson. "Computers and Intractability: A Guide to the Theory of NP-Completeness (Series of Books in the Mathematical Sciences)", New York: W.H. Freeman, 1979.
- 5- E. Horowitz, S. Sahni, S. Rajasekaran. "Fundamentals of Computer Algorithms", Computer Science Press, 1996.
- 6- Ian Parberry and William Gasarch. "Problems on Algorithms", Second Edition, Prentice Hall, 2002.
- 7- Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman. "data structures and algorithms", Addison-Wesley, 1985.

منابع و مراجع فارسی

- ۱- قدسی، محمد. "داده ساختارها و مبانی الگوریتمها"، چاپ دوم، انتشارات فاطمی، ۱۳۸۹
- ۲- فراهی، احمد و تنها، جعفر. "تحلیل و طراحی الگوریتم"، چاپ اول، دانشگاه پیام نور، ۱۳۹۰

Abolfazl esfandi