

سرشناسه: مدیری، ناصر
عنوان: مهندسی اصول پیچیدگی نرم افزار
تالیف و تدوین: ناصر مدیری، مسعود رفیعی

مشخصات نشر: تهران
مشخصات ظاهری: ۳۶۲ ص، مصور
شابک: 978-600-6236-32-2

وضعیت فهرست نویسی: قیفا
یادداشت: واژه نامه
یادداشت: کتاب نامه

موضوع: نرم افزار - ارزشیابی
شناسه افزوده: رفیعی، مسعود

رده بندی کنگره: ۱۳۹۱ ۴م ۴۴ الف ۷۶/۷۶ Q

رده بندی دیویی: ۰۰۵/۳۰۲۸۷۲۲

شماره کتابشناسی ملی: ۲۹۴۱۱۰۶



www.mehreganeghalam.com

انتشارات مهرگان قلم

مؤلف: ناصر مدیری - مسعود رفیعی

ناشر: مهرگان قلم

نوبت چاپ اول: ۱۳۹۱

چاپ و صحافی: سپهرنوین

تیراژ: ۵۰۰ نسخه

قیمت: ۱۵۰۰۰۰ ریال

شابک: ۹۷۸-۶۰۰-۶۲۳۶-۳۲-۲

کلیه حقوق این اثر متعلق به انتشارات مهرگان قلم می باشد.

تهران، خیابان انقلاب، خیابان دانشگاه، بین خیابان روانمهر و خیابان وحید نظری، کوچه قدیری، پلاک ۲۲ واحد ۳

تلفن: ۶۶۹۶۰۷۶۳ - ۶۶۹۵۸۵۵۸

پیشگفتار

لحظات، بسیار مهیج و به یادماندنی است زیرا در عصر انفجار اطلاعات زندگی می‌کنیم، طبق تحقیقات انجام گرفته توسط تیمی در دانشگاه برکلی دنیا ما بین یک و دو اگزا بایت اطلاعات بی نظیر در هر سال تولید می‌کند. که تقریباً ۲۵۰ مگا بایت برای هر مرد، زن و بچه در روی زمین است. هر اگزا بایت یک بیلیون گیگا بایت است. در علم کامپیوتر اندازه پیچیدگی نرم افزار به طور قابل توجه در حال افزایش است.

امروزه با هزینه‌های رو به رشد نرم افزار سازمانهای نرم افزاری را بر آن داشته تا دنبال راه حلی برای کاهش این هزینه‌ها باشند. برای نیل به این مقصود محققان دنبال رابطه‌ای بین ویژگی‌های نرم افزار و مشکلات و دشواریهای کار توسعه نرم افزار بودند. دشواری کار زمان بیشتری برای انجام آن طلب می‌کند در این زمان منابع بیشتری بکار گرفته می‌شوند. منابع بیشتر یعنی تحمیل هزینه‌های بیشتر. یکی از دلایل پرداختن به موضوع پیچیدگی نرم افزار و اندازه‌گیری آن برای کنترل هزینه‌ها در چرخه حیات نرم افزار است زیرا پیچیدگی نرم افزار از عوامل اصلی در افزایش سریع هزینه‌های توسعه و نگهداری عنوان می‌شود. پیچیدگی نرم افزار فاکتوری است که در فعالیت برنامه‌ریزی کمتر شناخته شده و به راحتی قابل اندازه‌گیری یا توصیف نیست و غالباً در طول فرایند برنامه‌ریزی پروژه نادیده گرفته می‌شود. بنابراین به دنبال روشی کمی برای پیش‌بینی میزان دشواری نگهداری، تغییر و فهم نرم افزار هستیم. تا با اندازه‌گیری و کنترل آن هزینه‌ها را در چرخه حیات نرم افزار کاهش دهیم.

در حالت کلی پیچیدگی نرم افزار، میزان کار فکری مورد نیاز برای درک نرم افزار را مشخص می‌کند. پیچیدگی در فاز توسعه نرم افزار، تلاش مورد نیاز برای آزمون و اشکال‌زدایی برنامه، پیمانه‌ها و زیر سیستم‌ها را به شدت تحت تأثیر قرار می‌دهد. در فاز نگهداری نرم افزار، پیچیدگی دشواری مکان‌یابی و تصحیح خطاهای کشف نشده پیاده‌سازی و همچنین میزان تلاش مورد نیاز برای تفسیر پیمانه‌هایی از برنامه که باید تصحیح شوند را مشخص می‌کند.

کار زیادی در مورد منشأ و ماهیت پیچیدگی نرم افزار انجام نشده است. بیشتر کارهای انجام شده در این زمینه به تأثیر پیچیدگی بر پروژه‌های نرم افزاری پرداخته‌اند که در رأس آنها کیفیت و هزینه‌های محصول قرار می‌گیرد. توجه مدیران و مهندسان پروژه‌های نرم افزاری به پیچیدگی نرم افزار برای کنترل و پیش‌بینی کیفیت و بهره‌وری محصول است.

گرچه در مورد علل و ماهیت پیچیدگی نرم افزار مطالب زیادی موجود نیست. اما چندین تعریف برای پیچیدگی ارائه شده است. بیشتر آنها بر اساس این مفهوم هستند که پیچیدگی نرم افزار میزان دشواری تحلیل، طراحی، کد، تست، نگهداری، تغییر و اصلاح نرم افزار است.

اما با توجه به اهمیت موضوع امنیت و ارتباط آن با پیچیدگی، لازم بود تا در چاپ دوم این کتاب فصل یازدهم با عنوان امنیت نرم افزار تدوین شود تا به رویکردی جهت ارتباط امنیت نرم افزار و پیچیدگی نرم افزار بپردازیم و در فصل دوازدهم نیز نرم افزاری کاربردی جهت آنالیز پیچیدگی نرم افزار معرفی می کنیم تا مطالب این کتاب به صورت عملی نیز مورد مطالعه قرار گیرد.

سرفصل های این کتاب به گونه ای است که افراد بتوانند مطالب مربوط به مفایم و ماهیت پیچیدگی های نرم افزار و دلیل آن را فرا گرفته و از مقالات متعددی در این زمینه بهره برده و تحقیقات خود را در صورت تمایل آغاز نمایند.

با توجه به گستردگی این مباحث، در پایان هر فصل منابعی جهت مطالعه ی بیشتر آورده شده است. خصوصیت بارز این کتاب، سادگی بیان به همراه استفاده از اشکال، جداول و طرح سوالاتی برای سنجش میزان یادگیری خود در جهت درک بهتر مفاهیم در هر بحث است. مطالب مطرح شده در هر فصل عبارتند از:

فصل اول، در این فصل مفاهیم پیچیدگی نرم افزار، ماهیت و تعریف پیچیدگی بیان شده و دسته بندی پیچیدگی های نرم افزار را بیان می کنیم.

فصل دوم، در این فصل به تعامل بین نگهداری و تولید پرداخته سپس لزوم تولید ورژن های مختلف بیان می شود. با مفهوم علمی تغییر و لزوم تغییر در برنامه پس از تحویل آشنا می شویم و چرخه تغییر مستندات را بررسی خواهیم کرد و در نهایت به بررسی هزینه های نگهداری نرم افزار و روشهای آن می پردازیم.

فصل سوم، در این فصل با تعریف علمی و ماهیت اندازه گیری را بیان می کنیم سپس روش های اندازه گیری را توضیح می دهیم و در انتها پارامترهای مهم در اندازه گیری را بیان می کنیم.

فصل چهارم، در این فصل ضمن آشنایی با متریک های نرم افزاری و انواع آن، پیچیدگی جریان اطلاعاتی و ساختمان داده را بیان می کنیم.

فصل پنجم، در این فصل مدل McCabe را بیان می کنیم. مزایا و مشکلات آن را بررسی می کنیم.

فصل ششم، در این فصل مدل Halsted را بیان می کنیم. مزایا و مشکلات آن را بررسی می کنیم.

فصل هفتم، در این فصل به شرح مهندسی نیازمندیا می‌پردازیم اجزای آن را بررسی می‌کنیم و در انتها تفاوتها و استانداردهای آنها را تشریح می‌نماییم.

فصل هشتم، در این فصل به معنا و مفهوم معماری، انواع معماری و سبکهای آن می‌پردازیم. امنیت نرم افزار را بیان کرده و در نهایت مدل های امنیتی و معماری سیستم معرفی می‌کنیم.

فصل نهم، در این فصل ماهیت پیچیدگی های طراحی نرم افزار و انواع آن را بیان می‌کنیم و نقش پیچیدگی های طراحی در افزایش پیچیدگی ها در فاز کدها را تشریح می‌کنیم و در نهایت متدهایی را برای کاهش پیچیدگی ها در فاز طراحی معرفی می‌کنیم.

فصل دهم، در این فصل ماهیت پیچیدگی های کد و انواع پیچیدگی های کد را شرح می‌دهیم و روش هایی را برای کاهش پیچیدگی در فاز کد بیان می‌کنیم و راه حل هایی را برای کاهش پیچیدگی پیشنهاد می‌کنیم و در نهایت با بررسی نقش کاهش پیچیدگی ها در فاز کد، به کم کردن هزینه های نگهداری یک نرم افزار می‌پردازیم.

فصل یازدهم، در این فصل امنیت نرم افزار، نیازمندی ها و مؤلفه های امنیت، سیاست امنیتی، راهکارهای امنیتی و تضمین های امنیتی را بررسی خواهیم کرد. با مدل های امنیتی مانند مدل های کنترل دسترسی و مدل های جریان اطلاعات آشنا خواهیم شد. راهبردهای معماری برای امنیت نرم افزاری را مرور و مقایسه خواهیم کرد، سپس رابطه امنیت با پیچیدگی نرم افزار را بیان می‌کنیم.

فصل دوازدهم، در این فصل به بررسی نرم افزار هایی برای محاسبه پیچیدگی و آنالیز نرم افزار می‌پردازیم. نرم افزار Understand نسخه 2.6 که یک نرم افزار آنالیز (برای بررسی و ویرایش) سورس کدها می باشد را مورد مطالعه قرار می‌دهیم و در حقیقت کدهای برنامه ی شما را تجزیه و تحلیل می‌کند.

امید است این کتاب بتواند در خدمت تمامی افرادی که به نحوی در ارتباط با نرم افزار هستند و بخصوص دانشجویان تحصیلات تکمیلی رشته های مختلف علوم کامپیوتر قرار گیرد. در خاتمه بر خود لازم می‌دانیم از کسانی که به نوعی در گردآوری مطالب این کتاب ما را یاری رساندند، کمال تشکر، قدردانی را به عمل آوریم.

nassermodiri@yahoo.com , masoud_r62@yahoo.com

پست الکترونیکی انعکاس نظرات:

ناصر مدیری - مسعود رفیعی

۱۳۹۱

فهرست مطالب

فصل ۱

۱۸	۱-۱: خلاصه بخش
۱۸	۲-۱: پیچیدگی نرم افزار
۱۹	۳-۱: تعریف و ماهیت پیچیدگی نرم افزار
۲۰	۴-۱: علل پیچیدگی نرم افزار
۲۲	۵-۱: دسته بندی پیچیدگی نرم افزار
۲۵	۶-۱: پیچیدگی در چرخه حیات نرم افزار
۲۶	۷-۱: فازهای پیچیدگی نرم افزار
۲۶	۸-۱: دلایل بروز پیچیدگی نرم افزار
۲۷	۹-۱: تاثیر و نتایج پیچیدگی نرم افزار
۲۷	۱-۹-۱: کیفیت نرم افزار
۲۷	۲-۹-۱: نگهداری نرم افزار
۲۸	۱۰-۱: بهره وری
۲۸	۱۱-۱: پیچیدگی و مهندسی نیازمندیا
۳۰	۱۲-۱: مدل عوامل پیچیدگی نرم افزار
۳۳	۱۳-۱: عوامل پیچیدگی ذاتی نرم افزار
۳۳	۱-۱۳-۱: نیازمندیهای کارکردی
۳۴	۲-۱۳-۱: نیازمندیهای غیر کارکردی
۳۶	۱۴-۱: پیچیدگی عارضی نرم افزار
۳۷	۱۵-۱: نتیجه گیری
۳۷	۱۶-۱: سوالات متداول
۳۸	۱۷-۱: منابعی برای مطالعه بیشتر

فصل ۲

۴۰	۱-۲: خلاصه بخش
۴۱	۲-۲: ماهیت نگهداری
۴۱	۳-۲: تعریف نگهداری
۴۲	۴-۲: نرم افزار تعمیر و نگهداری چیست؟

۴۳	۱-۴-۲: توانایی‌ها و ویژگی‌های یک نرم افزار نگهداری و تعمیرات (CMMS)
۴۴	۲-۴-۲: ویژگی‌های نرم افزار CMMS INTERNET BASE
۴۵	۵-۲: دسته‌بندی‌های تعمیر و نگهداری نرم افزار
۴۶	۶-۲: دسته بندی ISO در تعمیر نرم افزاری
۴۷	۷-۲: هزینه و رقابت‌ها
۴۸	۱-۷-۲: نسبت هزینه های تعمیر و نگهداری نرم افزار
۴۸	۲-۷-۲: هزینه های تعمیر مطلق نرم افزار
۴۸	۳-۷-۲: انواع وظایف تعمیر
۴۸	۴-۷-۲: مقادیر کدهای موروثی
۴۹	۸-۲: مدل‌ها، وظایف، استفاده مجدد و ابزارها
۵۰	۱-۸-۲: استفاده مجدد
۵۱	۲-۸-۲: وظایف
۵۲	۳-۸-۲: ابزار و محصولات در دسترس از لحاظ تجاری
۵۳	۹-۲: فرایند
۵۴	۱۰-۲: فرایند پیاده سازی
۵۵	۱۱-۲: اصلاحات پیاده سازی
۵۶	۱۲-۲: مدیریت تعمیر و نگهداری
۵۶	۱-۱۲-۲: طرح و برنامه ریزی
۵۷	۱۳-۲: ساختار سازمانی متفاوت در سازمان و تعمیر نرم افزار
۵۸	۱۴-۲: مهندسی معکوس
۵۹	۱۵-۲: مهندسی مجدد
۶۰	۱۶-۲: ارتباط پیچیدگی با نگهداری نرم افزار
۶۱	۱۷-۲: فرایند نگهداری نرم افزار
۶۱	۱۸-۲: انواع نگهداری
۶۳	۱۹-۲: هزینه های نگهداری و برآورد آن
۶۴	۲۰-۲: نتیجه‌گیری
۶۵	۲۱-۲: سوالات متداول
۶۶	۲۲-۲: منابعی برای مطالعه بیشتر

۶۸	۲-۳: اندازه‌گیری و اهداف اندازه‌گیری
۷۰	۳-۳ کاربردهای اندازه‌گیری
۷۰	۴-۳ تعریف متریک نرم‌افزاری
۷۱	۵-۳ طبقه‌بندی متریک‌های نرم‌افزاری
۷۴	۶-۳ متریک‌های پیچیدگی نرم‌افزار
۷۵	۱-۶-۳ خطوط متن
۷۵	۲-۶-۳ متریک فرایست نیازمندی
۷۶	۳-۶-۳ تعداد خطوط کد
۷۷	۴-۶-۳ متریک نقاط کارکرد
۸۰	۷-۳ متریک‌های جریان اطلاعات
۸۰	۱-۷-۳ متریک چسبندگی
۸۰	۲-۷-۳ متریک‌های پیوستگی
۸۰	۳-۷-۳ متریک جریان اطلاعات HENRY و KAFURA
۸۱	۸-۳ متریک‌های شیء‌گرا
۸۲	۱-۸-۳ سلسله متریک‌های C&K
۸۴	متریک پاسخ‌های کلاس (RFC)
۸۴	متریک فقدان انسجام بین متدها (LCOM)
۸۵	متریک پیوستگی بین کلاس‌های شیء (CBO)
۸۵	۲-۸-۳ سلسله متریک‌های MOOD
۸۷	۹-۳ نتیجه‌گیری
۸۸	۱۰-۳ سوالات متداول
۸۸	۱۰-۳ منابعی برای مطالعه بیشتر

فصل ۴

۹۰	۱-۴ خلاصه بخش
۹۰	۲-۴ تعریف متریک‌های نرم‌افزاری
۹۳	۳-۴ اندازه و میزانهای پیچیدگی
۹۵	۴-۴ متریک‌های دیگر پیچیدگی
۹۵	۱-۴-۴ متریک جریان اطلاعات HENRY و KAFURA
۹۶	۲-۴-۴ استاندارد IEEE 982.2
۱۰۰	۵-۴ مدل پیچیدگی شیء‌گرا

۱-۲	۶-۴ دسته بندی معیارهای شی گرا (OO).....
۱-۳	۷-۴ سوالات متداول.....
۱-۴	۸-۴ منابعی برای مطالعه بیشتر.....

فصل ۵

۱-۶	۱-۵ خلاصه بخش.....
۱-۶	۲-۵ عدد دورانی MCCABE.....
۱-۷	۳-۵ محاسبه پیچیدگی سیکلوماتیک CC.....
۱-۷	۴-۵ مثالی از روش مک کیب.....
۱-۸	۵-۵ کاربرد های پیچیدگی دورانی مک کیب.....
۱-۹	۶-۵ سوالات متداول.....
۱-۱۰	۷-۵ منابعی برای مطالعه بیشتر.....

فصل ۶

۱۱۲	۱-۶: خلاصه بخش.....
۱۱۲	۲-۶ روش HALSTEAD.....
۱۱۳	۳-۶ مشکلات متریکهای هالستد.....
۱۱۳	۴-۶: میزان های HALSTEAD.....
۱۱۴	۵-۶: مثالی از روش HALSTEAD.....
۱۱۶	۶-۶: سوالات متداول.....
۱۱۶	۷-۶: منابعی برای مطالعه بیشتر.....
۱۱۸	۱-۷ خلاصه بخش.....
۱۱۸	۲-۷ مقدمه.....
۱۱۹	۳-۷ تاریخچه مهندسی نیازمندیها.....
۱۲۰	۴-۷ تعریف نیازمندی.....
۱۲۱	۵-۷ طبقه بندی نیازمندیها.....
۱۲۴	۶-۷ ذینفعان.....
۱۲۶	۷-۷ مهندسی نیازمندیها.....
۱۲۷	۸-۷ دلایل اهمیت مهندسی نیازمندیها.....
۱۳۰	۹-۷ مراحل مهندسی نیازمندیها.....

۱۳۱	۷-۹-۱ استخراج نیازمندیها
-----	--------------------------

فصل ۷

۱۳۴	۷-۱۰ تحلیل و مذاکرات نیازمندیها
۱۳۶	۷-۱۱ مستندسازی نیازمندیها
۱۴۱	۷-۱۲ تایید اعتبار نیازمندیها
۱۴۱	۷-۱۳ مدیریت نیازمندیها
۱۴۲	۷-۱۴ محصول مهندسی نیازمندیها
۱۴۶	۷-۱۵ چگونگی انجام مهندسی نیازمندیها
۱۴۶	۷-۱۶ نتیجه گیری
۱۴۷	۷-۱۷ سوالات متداول
۱۴۷	۷-۱۸ منابعی برای مطالعه بیشتر

فصل ۸

۱۵۰	۸-۱ خلاصه بخش
۱۵۰	۸-۲ مقدمه
۱۵۲	۸-۳ معماری ها از کجا می آیند؟
۱۵۳	۸-۴ انواع معماری
۱۵۶	۸-۵ معماری نرم افزار
۱۵۷	۸-۵-۱ تعریف معماری نرم افزار
۱۵۷	۸-۵-۲ اجزای معماری نرم افزار
۱۵۹	۸-۶ سایر دیدگاه ها
۱۶۰	۸-۷ حلقه کاری معماری
۱۶۱	۸-۸ مراحل فرآیند معماری نرم افزار
۱۶۳	۸-۹ چرا معماری نرم افزار مهم است؟
۱۷۱	۸-۱۰ درباره معماری نرم افزار
۱۷۳	۸-۱۱ چه چیزی یک معماری خوب را می سازد؟
۱۷۷	۸-۱۲ چه زمانی می توان طراحی را شروع کرد؟
۱۷۸	۸-۱۳ معماری در چرخه حیات
۱۷۸	۸-۱۴ الگوهای معماری، مدل های مرجع و معماری های مرجع

۱۸۱	۱۵-۸ دیدها و ساختارهای معماری
۱۸۲	۱۶-۸ ساختارهای نرم افزار
۱۸۷	۱۷-۸ ساختارهای معماری یک سیستم
۱۸۸	۱-۱۷-۸ ساختارهای مرتبط با هم
۱۸۹	۲-۱۷-۸ کدام ساختارها انتخاب می شوند؟
۱۹۰	۱۸-۸ الگوها و سبک های معماری
۱۹۱	۱-۱۸-۸ سبک های معماری نرم افزار
۱۹۲	۲-۱۸-۸ تعریف سبک معماری
۱۹۲	۳-۱۸-۸ انواع سبک های معماری
۲۰۲	۱۹-۸ مدل های معماری نرم افزار
۲۰۲	۱-۱۹-۸ معماری MAIN FRAME
۲۰۴	۲-۱۹-۸ معماری FILE SERVER
۲۰۵	۳-۱۹-۸ معماری CLIENT/SERVER
۲۱۱	۴-۱۹-۸ معماری سرویس گرا
۲۱۸	۵-۱۹-۸ معماری مبتنی بر ERP
۲۲۰	۳-۱۹-۸ معماری نرم افزار شی گرا
۲۲۸	۲۰-۸ خصوصیات معماری
۲۲۹	۲۱-۸ وظیفه مندی و معماری
۲۲۹	۲۲-۸ معماری و خصوصیات کیفی
۲۳۰	۱-۲۲-۸ مشخصه های کیفیتی نرم افزار
۲۳۲	۲-۲۲-۸ مدل های کیفیتی در نرم افزار
۲۳۶	۳-۲۲-۸ صفت کیفیتی امنیت در معماری نرم افزار
۲۳۸	۲۳-۸ معماری نرم افزار در آینده
۲۴۰	۲۴-۸ نتیجه گیری
۲۴۰	۲۵-۸ سؤالات متداول
۲۴۱	۲۶-۸ منابعی برای مطالعه بیشتر

فصل ۹

۲۴۴	۱-۹ خلاصه بخش
۲۴۵	۲-۹ قوائد طراحی
۲۴۵	۳-۹ انتزاع/تجرد/چکیدگی (ABSTRACTION)

۲۴۵	 ۴-۹ وابستگی (COUPLING)
۲۴۵	 ۵-۹ پیوستگی (COHESION)
۲۴۶	 ۶-۹ تفکیک یا ماژول بندی (DECOMPOSITION/MODULARIZATION)
۲۴۶	 ۱-۶-۹ کپسوله سازی/پنهان سازی اطلاعات (HIDING INFORMATION/ENCAPSULATION)
۲۴۶	 ۷-۹ فعالیتهای طراحی
۲۴۷	 ۸-۹ روشهای کلی طراحی
۲۴۸	 ۹-۹ خصوصیات کیفی طراحی (QUALITY ATTRIBUTES)
۲۴۸	 ۱۰-۹ مراحل طراحی
۲۴۹	 ۱-۱۰-۹ فعالیت های طراحی سیستم های نرم افزاری بزرگ
۲۴۹	 ۱۱-۹ متدهای طراحی DESIGN METHODS
۲۵۰	 ۱۲-۹ توصیف طراحی (DESCRIBING DESIGN)
۲۵۰	 ۱۳-۹ استراتژی های طراحی (DESIGN STRATEGIES)
۲۵۱	 ۱۴-۹ کیفیت طراحی (DESIGN QUALITY)
۲۵۲	 ۱۵-۹ سطح ذینفعان
۲۵۴	 ۱۶-۹ سطح تیم پروژه
۲۵۶	 ۱۷-۹ مدل نتایج پیچیدگی نرم افزار
۲۵۷	 ۱۸-۹ خطا پذیری
۲۵۸	 ۱۹-۹ متریک پیچیدگی مبتنی بر نیازمندیها
۲۵۹	 ۱-۱۹-۹ پیچیدگی ورودی خروجی (I/O COMPLEXITY)
۲۵۹	 ۲-۱۹-۹ پیچیدگی نیازمندیهای کارکردی (FUNCTIONAL REQUIREMENT COMPLEXITY)
۲۶۱	 ۳-۱۹-۹ پیچیدگی نیازمندیهای غیر کارکردی (NON-FUNCTIONAL REQUIREMENT COMPLEXITY)
۲۶۶	 ۲۰-۹ سوالات متداول
۲۶۶	 ۲۱-۹ منابعی برای مطالعه بیشتر

فصل ۱۰

۲۶۸	 ۱-۱۰ خلاصه بخش
۲۶۸	 ۲-۱۰ الگوریتم
۲۶۸	 ۳-۱۰ خصوصیات یک الگوریتم
۲۶۹	 ۱-۳-۱۰ منشاء واژهی الگوریتم
۲۶۹	 ۲-۳-۱۰ نقش الگوریتمها در علوم رایانه

۲۶۹	۱۰-۴ تحلیل الگوریتم
۲۷۰	۱۰-۴-۱ جنبه حقوقی
۲۷۰	۱۰-۵ ارزیابی کارایی الگوریتم‌ها
۲۷۰	۱۰-۵-۱ پیچیدگی زمانی
۲۷۱	۱۰-۵-۲ پیچیدگی حافظه
۲۷۳	۱۰-۶ نمایش پیچیدگی الگوریتم
۲۷۴	۱۰-۷ بهبود پیچیدگی یک برنامه
۲۷۴	۱۰-۸ سوالات متداول
۲۷۴	۱۰-۹ منابعی برای مطالعه بیشتر

فصل ۱۱

۲۷۶	۱۱-۱ خلاصه بخش
۲۷۶	۱۱-۲ مقدمه
۲۷۷	۱۱-۳ نیازمندی‌های امنیت
۲۷۸	۱۱-۴ سیاست امنیتی
۲۷۸	۱۱-۵ راهکارهای امنیتی
۲۷۹	۱۱-۶ تضمین‌های امنیتی
۲۷۹	۱۱-۷ مؤلفه‌های امنیت
۲۸۱	۱۱-۸ مدل‌های امنیتی
۲۸۱	۱۱-۸-۱ مدل‌های کنترل دسترسی
۲۸۵	۱۱-۸-۲ مدل‌های جریان اطلاعات
۲۸۵	۱۱-۹ معیارهای رسمی برای ترکیب مؤلفه‌ها
۲۸۶	۱۱-۹-۱ ترکیب آبادی-آپرسورت در چارچوب آپرن-اِشناپدر
۲۸۷	۱۱-۱۰ جامعیت
۲۸۸	۱۱-۱۱ رازداری (امنیت اطلاعات)
۲۸۸	۱۱-۱۱-۱ ارائه چارچوب
۲۸۹	۱۱-۱۱-۲ ترکیب
۲۸۹	۱۱-۱۲ راهبردهای معماری برای امنیت نرم افزاری
۲۹۰	۱۱-۱۲-۱ تکنیک برجسب گذاری مبتنی بر شیء
۲۹۲	۱۱-۱۲-۲ مدل سازی ایمنی مبتنی بر UML
۲۹۲	۱۱-۱۲-۳ ASTER

۲۹۳	۱۱-۱۲-۴ مدل معماری سیستم
۲۹۶	۱۱-۱۲-۵ مقایسه رویکردهای معماری برای امنیت نرم افزار
۲۹۷	۱۱-۱۳ رابطه پیچیدگی و امنیت
۲۹۷	۱۱-۱۴ نتیجه گیری
۲۹۸	۱۱-۱۵ سوالات متداول
۲۹۸	۱۱-۱۶ منابعی برای مطالعه بیشتر

فصل ۱۲

۳۰۰	۱۲-۱ خلاصه بخش
۳۰۰	۱۲-۲ آشنایی با نرم افزار UNDERSTAND
۳۰۱	۱۲-۳ ایجاد پروژه جدید
۳۰۵	۱۲-۴ پنجره ی ENTITY FILTER
۳۰۶	۱۲-۵ پنجره ی INFORMATION BROWSER
۳۰۹	۱۲-۶ جست و جو در فایل ها
۳۱۰	۱۲-۷ نمای گرافیکی
۳۱۲	۱۲-۸ بروز رسانی تغییرات در فایل
۳۱۴	۱۲-۹ پنجره ی ARCHITECTURE BROWSER (مرورگر معماری)
۳۱۸	۱۲-۱۰ گراف های وابستگی معماری (DEPENDENCY GRAPHS)
۳۲۲	۱۲-۱۱ گرفتن اطلاعات در INFORMATION BROWSER
۳۲۴	۱۲-۱۲ مقایسه ی فایل ها و پوشه ها
۳۲۵	۱۲-۱۳ متریک ها (METRICS)
۳۲۹	۱۲-۱۴ گزارش ها (REPORTS)
۳۳۳	۱۲-۱۵ نمایش متریک ها به صورت یک نقشه درختی
۳۳۶	۱۲-۱۶ چک کردن کد ها (CHECK CODE)
۳۴۷	واژه نامه