

۱۳۵۰۸۰۸

اصول طراحی و توسعه نرم افزارهای مطمئن

۱۳۵۰۸۰۸

مؤلفین:

رضا ابراهیمی آقائی

مهران علیدهستانی



**NAGHOOS
PUBLICATION**

سرشناسه : ابراهیمی آتانی، رضا، ۱۳۵۹
 عنوان و نام پدیدآور : اصول طراحی و توسعه نرم افزارهای مطمئن
 رضا ابراهیمی آتانی، مهران علی دوست نیا.
 مشخصات نشر : تهران: انتشارات ناقوس، ۱۳۹۴.
 مشخصات ظاهری : ۳۰۲ ص: مصور، جدول.
 شابک : ۹-۷۸۶-۳۷۷-۹۶۴-۹۷۸
 بهای : ۱۹۵,۰۰۰ ریال
 وضعیت فهرست نویسی : فیپا
 موضوع : نرم افزار - اطمینان پذیری
 موضوع : نرم افزار - تولید
 موضوع : معماری نرم افزار
 شناسه افزوده : علی دوست نیا، مهران، ۱۳۶۷
 رده بندی سکره : ۱۳۹۴/۲ الف ۶/۷۶/۷۶ QAV۶
 رده بندی دیمی : ۰۰۵
 شماره کتابشناس ملی : ۳۸۹۱۰۰۰



برای دیدن آدرس به Online زیر مراجعه کنید:

www.naghoospress.com

انتشارات ناقوس

چاپ اول

نام کتاب : اصول طراحی و توسعه نرم افزارهای مطمئن

ناشر : انتشارات ناقوس

تألیف : رضا ابراهیمی آتانی، مهران علی دوست نیا

S.M.S : ۳۰۰۰۴۵۲۷۰۳

چاپ اول : ۱۳۹۴

تیراژ : ۲۰۰ جلد

چاپ و صحافی : نارنجستان

سرپرست صفحه آرا : صدف پورعبدی

قیمت : ۱۹۵,۰۰۰ ریال

شابک : ۹-۷۸۶-۳۷۷-۹۶۴-۹۷۸

ISBN : 978-964-377-786-9

حق چاپ برای، رمایه گذار محفوظ است. هر تمامی یا قسمتی از این اثر به صورت حروفچینی یا چاپ مجدد، چاپ افست، پلرکیپی، فتوکپی و انواع دیگر چاپ ممنوع است و پیگرد قانونی دارد.

مرکز پخش: انتشارات ناقوس

۱- میدان انقلاب، خیابان کارگر جنوبی، خیابان ۱۲ فروردین- بین وحید نظری و روانمهر- بن بست حقیقت- پلاک ۴

تلفن و فاکس : ۶۳۳۲۵ (۲۰ خط)

۲- کتابفروشی الیاس: خیابان انقلاب، نبش ۱۲ فروردین تلفن: ۶۶۴۰۵۰۸۴

۳- کتابفروشی گلریزان: ضلع جنوب شرقی میدان انقلاب، پلاک ۱۱ تلفن: ۶۶۹۷۶۲۳۰

فهرست مطالب

۳۸	۱-۳-۲ انتخاب تکنیک	۱۱	پیشگفتار
۳۸	۲-۳-۲ تعداد نسخه‌های برنامه		فصل اول: مقدمه‌ای بر طراحی نرم‌افزارهای
۳۸	۳-۳-۲ تست پذیرش و مکانیزم تصمیم‌گیری	۱۵	مطمئن
۴۱	۴-۲ تکنیک‌های ترمیم خطا و نقاط بازیابی	۱۷	۱-۱ تعاریف اولیه
۴۲	۱-۴-۲ راهبردهای ترمیم خطا	۱۸	۲-۲ تحمل‌پذیری خطا
۴۲	۲-۴-۲ مزایای روش برگشت به عقب	۱۸	۱-۲-۱ کاربرد تحمل‌پذیری خطا
۴۴	۳-۴-۲ معایب روش برگشت به عقب	۱۹	۲-۲-۱ تحمل‌پذیری خطا افزونگی
۴۴	۴-۴-۲ دسته‌بندی تکنیک‌ها	۱۹	۳-۲-۱ اهداف تحمل‌پذیری خطا
۴۴	۵-۴-۲ روش‌های ایجاد نقطه بازیابی		۴-۲-۱ بررسی قابلیت اطمینان در یک سیستم
۴۵	۶-۴-۲ روش‌های تولید نقاط بازیابی		نرم‌افزاری
۴۶	۶-۲ اجرای برنامه بدون نقاط بازیابی	۲۱	۵-۲-۱ میانگین زمان تا خرابی MTTF
۴۸	۷-۲ نقاط بازیابی با اندازه‌های مساوی	۲۲	۶-۲-۱ میانگین زمان تا تعمیر MTTR
۵۰	۸-۲ نقاط بازیابی در برنامه‌های ماژولار	۲۲	۷-۲-۱ میانگین زمان بین خرابی MTBF
۵۰	۹-۲ زمان اجرای یک برنامه ماژولار	۲۳	۳-۱ تحمل‌پذیری خطا در سخت‌افزار
	۱۰-۲ زمان اجرای یک نیازمندی پردازشی	۲۴	۱-۲-۱ تکنیک افزونگی غیرفعال
۵۱	مفروض	۲۸	۲-۲-۱ افزونگی فعال
۵۲	۹-۲ نقطه بازیابی تصادفی	۳۰	۴-۱ تحمل‌پذیری خطا در مهندسی نرم‌افزار
	۱۰-۲ زمان اجرای یک نیازمندی پردازشی	۳۲	سوالات نمونه فصل اول
۵۲	مفروض		فصل دوم: تنوع طراحی در طراحی نرم‌افزار
۵۵	سوالات نمونه فصل دوم	۳۵	مطمئن
	فصل سوم: تکنیک‌های تنوع طراحی در	۳۵	۱-۲ اهداف تنوع طراحی
۵۹	تحمل‌پذیری خطا	۳۶	۲-۲ سطوح تنوع طراحی
	۱-۲ تکنیک‌های کلاسیک تحمل‌پذیری خطای	۳۶	۱-۲-۲ تنوع سخت‌افزار
۵۹	نرم‌افزاری	۳۶	۲-۲-۲ تنوع نرم‌افزار کاربردی
۶۰	۲-۲ انواع تکنیک‌های تنوع طراحی	۳۷	۳-۲-۲ تنوع در نرم‌افزار سیستم
	۱-۲-۲ بلاک‌های بازیابی (Recovery	۳۷	۴-۲-۲ تنوع عملگرها
۶۰	Blocks)	۳۷	۵-۲-۲ واسطه بین مولفه‌ها
۶۱	۱-۲-۲ پیاده‌سازی بلاک‌های بازیابی	۳۷	۳-۲ تکنیک‌های تنوع طراحی

فصل چهارم: برنامه نویسی چندمنسجه ای و

۸۹ خرجه مهندسی نرم افزار

۱-۴ تحمل پذیری خطا در مدل برنامه نویسی

چندمنسجه ای ۹۰

۲-۴ ایجاد نرم افزار چندمنسجه ای ۹۴

۳-۴ مشخصه های نسخه های عضو نرم افزارهای

چندمنسجه ای ۹۴

۴-۴ فرایند برنامه نویسی چندمنسجه ای ۹۵

۵-۴ توابع محیط اجرایی چندمنسجه ای ۹۶

۵-۴ یک الگوی طراحی برای نرم افزار چندمنسجه ای

۹۷

۶-۴ فاز نیازمندی سیستم: تعیین روش نظارت

نرم افزار چندمنسجه ای ۹۹

۶-۴ انتخاب ابعاد تنوع نرم افزار ۹۹

۶-۴ اجرای پروتکل توسعه نرم افزار

چندمنسجه ای در فاز تست ۱۰۱

۶-۴ بهره برداری از برنامه نویسی

چندمنسجه ای در فاز تست ۱۰۲

۶-۱ ارزیابی قابلیت اطمینان نرم افزار

چندمنسجه ای در فاز ارزیابی و پذیرش ۱۰۳

۶-۴ پایداری رویه نگهداری نرم افزار

چندمنسجه ای در فاز عملیاتی ۱۰۳

۷-۴ مدل سازی نرم افزار: پذیر خطا ۱۰۴

۸-۴ میزبانی و حمایت از محیط اجرایی ۱۰۵

۹-۴ تنوع چند لایه ۱۰۵

۹-۴ پشتیبانی نرم افزار چندمنسجه ای برای

سیستم های امنیتی ۱۰۶

۹-۴ اصلاح نرم افزار چندمنسجه ای عملیاتی ۱۰۷

۱۰-۴ برنامه نویسی چندمنسجه ای سلسله مراتبی ۱۰۷

۱-۱۰-۴ استفاده کامل از امکانات Ada ۱۰۹

۲-۱۰-۴ برنامه نویسی چندمنسجه ای در سطح

زیرروال ۱۱۰

۲-۱-۲-۲ جزئیات پیاده سازی ۶۲

۳-۱-۲-۲ مباحث تکمیلی در بلاک های

بازیابی ۶۳

۴-۱-۲-۲ مزایا و معایب بلاک های بازیابی ۶۴

۲-۲-۲ برنامه نویسی چندمنسجه ای ۶۵

۳-۲-۲ برنامه نویسی N بار خودبررسی ۶۶

۴-۲-۲ بلاک های بازیابی توزیع شده ۶۸

۴-۲-۲ روند کلی DRB ۶۹

۴-۲-۲ ساختار و عملکرد DRB ۶۹

۴-۲-۲ حالات اجرایی در DRB ۷۰

۴-۴-۲-۲ برخیز از DRB ۷۳

۵-۴-۲-۲ مسائل و مشکلات DRB ۷۳

۶-۴-۲-۲ معماری LDRB ۷۴

۵-۲-۲ بلاک بازیابی اجماعی ۷۴

۱-۵-۲-۲ ساختار و عملکرد CRB ۷۵

۲-۵-۲-۲ حالات اجرایی در CRB ۷۵

۳-۵-۲-۲ مسائل و مشکلات تکنیک CRB ۷۷

۴-۵-۲-۲ معماری CRB ۷۹

۶-۲-۲ پذیرش رأی گیری (AV) ۷۹

۱-۶-۲-۲ روند کلی AV ۷۹

۲-۶-۲-۲ ساختار و عملکرد AV ۸۰

۳-۶-۲-۲ حالات اجرایی در AV ۸۰

۴-۶-۲-۲ مسائل و مشکلات تکنیک AV ۸۲

۵-۶-۲-۲ معماری AV ۸۳

۳-۲-۲ مقایسه تکنیک ها ۸۳

۱-۳-۲ ویژگی های اصلی تکنیک های مبتنی بر

تنوع طراحی ۸۳

۲-۳-۲ سربار وارده به تکنیک های تنوع طراحی

۸۴

۲-۳-۲ مقایسه RCB و NVP ۸۶

۴-۳-۲ نتایج بررسی های تجربی مقایسه NVP

RCB و ۸۶

سوالات نمونه فصل سوم ۸۷

۱۱-۴	یک ارزیابی تجربی از ویژگی قابلیت اطمینان
۱۱۱	برنامه نویسی چندسخت‌های
۱۱۲	NASA 4-University پروژه
۱۱۲	RSDIMU پروژه
۳-۱۱-۴	مقایسه کیفی NASA 4-University و RSDIMU
۱۱۳	۴-۱۱-۴ تحلیل خطا در تست عملیاتی
۱۱۶	سوالات نمونه فصل چهارم
۱۱۷	فصل پنجم: تکنیک‌های تنوع داده و پیشرفته در تحمل پذیری خطا
۱۱۹	۱-۵ الگوریتم بیان مجدد داده
۱۲۰	۲-۵ بلاک‌های ترمیم
۱۲۱	۳-۵ برنامه نویسی N-کپی
۱۲۲	۴-۵ دوری دو مرحله‌ای
۱۲۳	۵-۵ تکنیک‌های پیشرفته در تحمل پذیری خطا
۱۲۴	۱-۵-۵ برنامه نویسی N-نسخه‌ای تطبیقی
۱۲۶	۲-۵-۵ رأی گیری فازی
۱۲۸	۳-۵-۵ پیگیری مجدد و جوانسازی مجدد
۱۲۹	۴-۵-۵ جوانسازی
۱۳۱	۵-۵-۵ چکیدگی
۱۳۲	۶-۵-۵ کاهش گراف موازی
۱۳۴	سوالات نمونه فصل پنجم
۱۳۷	فصل ششم: مسائل معماری در طراحی نرم افزارهای مطمئن
۱۳۹	۱-۶ رویکردهای اصلی در طراحی نرم افزارهای مطمئن
۱۴۲	۲-۶ آنالیز تحمل خطای نرم افزار
۱۴۲	۱-۲-۶ آنالیز قابلیت اتکا
۱۴۴	۱-۱-۲-۶ دسته بندی خطاها، نقص‌ها و شکست‌های بررسی شده
۱۴۵	۲-۱-۲-۶ آنالیز رفتار بلاک‌های بازایی
۱۴۶	۳-۱-۲-۶ آنالیز رفتار برنامه نویسی خودبررسی
۱۴۷	۴-۱-۲-۶ آنالیز رفتار برنامه نویسی چندسخت‌های
۱۴۸	۵-۱-۲-۶ توصیف احتمالات شکست
۱۵۰	۲-۲-۶ آنالیز هزینه
۱۵۲	۳-۶ مسائل مطرح در معماری‌های تحمل پذیر
۱۵۳	خطای نرم افزار
۱۵۳	۱-۳-۶ بلاک‌های بازایی
۱۵۴	۲-۳-۶ برنامه نویسی چندسخت‌های
۱۵۴	۳-۳-۶ برنامه نویسی خودبررسی
۱۵۵	سوالات نمونه فصل ششم
۱۵۷	فصل هفتم: مدیریت استثناء در طراحی نرم افزارهای مطمئن
۱۵۷	۱-۷ مفاهیم پایه
۱۵۸	۲-۱-۷ تشخیصات و معانی برنامه استاندارد
۱۶۱	۳-۱-۷ تشخیصات و معانی برنامه استثناء
۱۶۴	۴-۱-۷ تشخیصات، نقص‌ها و خطاهای برنامه
۱۶۸	۵-۷ بررسی استثناء در برنامه‌های ماژولار سلسله مراتبی
۱۶۸	۶-۷ تشخیصات و معانی برنامه سلسله مراتبی
۱۷۰	۷-۲-۷ تشخیصات و معانی برنامه نویسی شده
۱۷۲	۸-۲-۷ تشخیصات و معانی برنامه پیش فرض
۱۷۴	۹-۲-۷ تشخیصات و معانی برنامه مراتب انتزاع داده
۱۷۵	۱۰-۲-۷ تحمل نقص‌های طراحی
۱۷۶	۱۱-۲-۷ مدیریت استثناء
۱۸۰	سوالات نمونه فصل هفتم
۱۸۴	فصل هشتم: مروری بر تکنیک‌های تحمل پذیری خطا در سیستم عامل
۱۸۴	۱-۸ مفاهیم اولیه
۱۸۴	۱-۱-۸ سیستم عامل بلادرنگ
۱۸۵	۲-۱-۸ مهلت زمانی سخت، سخت و نرم

فصل نهم: روش‌های تست اطمینان نرم افزار ۲۱۴

۲۱۴	۱-۹ مبانی آزمون نرم افزار	۱۸۵	خطا	۲-۱-۸ ویدیوهای RTOSها و تکنیک‌های تحمل
۲۱۵	۱-۱-۹ اهداف آزمون	۱۸۵	۱-۲-۱-۸ مدیریت حافظه	
۲۱۵	۲-۱-۹ اصول آزمون	۱۸۷	۲-۲-۱-۸ ملاحظات هسته	
۲۱۶	۱-۲-۱-۹ آزمون‌پذیری نرم افزارها	۱۸۹	۳-۲-۱-۸ مدیریت نخ و فرآیند	
۲۱۸	۳-۱-۹ مدل V در تست نرم افزار	۱۹۰	۴-۲-۱-۸ زمان‌بندی	
۲۱۹	۴-۱-۹ چهار تکنیک اعتبار و صحت	۱۹۱	۵-۲-۱-۸ ارتباطات	
۲۱۹	۲-۹ سطوح مختلف آزمون	۱۹۲	۶-۲-۱-۸ مدیریت I/O	
۲۱۹	۱-۲-۹ تست واحد	۱۹۲	۷-۲-۱-۸ اراده وقفه	
۲۲۰	۲-۲-۹ تست یکپارچگی	۱۹۲	۲-۸ تحمل‌پذیری خطا در سیستم‌عامل‌های نمونه ۱۹۲	
۲۲۰	۳-۲-۹ تست سیستم	۱۹۳	۱-۲-۸ شیوه تحمل خطا در سیستم‌عامل unix	
۲۲۰	۴-۲-۹ تست پذیرش	۱۹۳	و برای سخت‌افزار	
۲۲۱	۳-۹ چرخه عمر آزمون نرم افزار	۱۹۵	۲-۲-۸ ارزیابی	
۲۲۱	۱-۳-۹ طرح‌ریزی تست	۱۹۶	۱-۲-۲-۸ Tandem GUARDIAN	
۲۲۲	۲-۳-۹ تحلیل تست	۱۹۶	۲-۲-۲-۸ IBM/MVS	
۲۲۲	۳-۳-۹ طراحی تست	۱۹۷	۳-۲-۲-۸ VAX/VMS	
۲۲۲	۴-۳-۹ ساخت و ارزیابی	۱۹۷	۳-۲-۸ مشخصه‌های اساسی خطا	
۲۲۲	۵-۳-۹ چرخه‌های تست	۱۹۸	۱-۲-۲-۸ طبقه‌بندی نقص و خطا	
۲۲۳	۶-۳-۹ تست نهایی و پیاده‌سازی	۱۹۸	۲-۳-۲-۸ GUARDIAN	
۲۲۳	۱-۳-۹ تست پس از پیاده‌سازی	۲۰۰	۴-۲-۸ MVS	
۲۲۳	۸-۳-۹ آزمون‌های محصول	۲۰۱	۱-۴-۲-۸ VMS	
۲۲۳	۴-۹ امکان‌سنجی	۲۰۲	۵-۲-۸ توزیع‌های خطا	
۲۲۴	۵-۹ روش تکراری و ارزیابی	۲۰۳	۶-۲-۹ توزیع‌های بازایی	
۲۲۴	۱-۵-۹ مدل ایکس پی (روش خیلی سریع)	۲۰۳	۲-۸ ارزیابی تحمل خطا در سیستم‌عامل‌های نمونه	
۲۲۵	۲-۵-۹ مدل اسکرام SCRUM	۲۰۴		
۲۲۵	۳-۵-۹ مدل تکامل یکپارچه‌سازی (CMMI)	۲۰۴	۱-۳-۸ ارزیابی زوج گرم‌های پردازشی	
۲۲۵	۴-۵-۹ ایزو ۹۰۰۰	۲۰۵	۲-۳-۸ اندازه‌گیری تحمل خطای نرم افزار	
۲۲۵	۵-۵-۹ ایزو ۱۵۵۰۴	۲۰۵	۳-۳-۸ قطعی به خاطر نرم افزار	
۲۲۶	۶-۹ انواع آزمون	۲۰۵	۴-۳-۸ توصیف صفات اختصاصی تحمل خطای	
۲۲۶	۱-۶-۹ تست عملکرد	۲۰۶	نرم افزار	
۲۲۶	۲-۶-۹ تست استرس	۲۰۸	۵-۳-۸ بررسی و تحلیل	
۲۲۷	۳-۶-۹ تست بار	۲۰۸	۶-۳-۸ ارزیابی روال‌های بازایی	
۲۲۷	۳-۶-۹ تست اکتشافی	۲۱۰	سوالات نمونه فصل هشتم	

فصل یازدهم: طراحی سیستم‌های با قابلیت

۲۶۱

اتکا بالا

۲۶۱

۱-۱۱ مقدمه

۲-۱۱ تحمل‌پذیری خطا در سیستم‌های با قابلیت

۲۶۲

اتکا بالا

۲-۱۱ کاربردهای سیستم‌های تحمل‌پذیر خطا

۲۶۴

۱-۳-۱۱ کاربردهای با طول عمر طولانی

۲۶۵

۲-۳-۱۱ کاربردهای با محاسبات بحرانی

۲۶۵

۳-۳-۱۱ کاربردهای با تعمیر و نگهداری دشوار

۲۶۵

۴-۳-۱۱ کاربردهای با دسترس‌پذیری بالا

۲۶۶

۴-۱۱ قابلیت اطمینان و تحمل‌پذیری خطا در

۲۶۶

سیستم‌های با قابلیت اتکای بالا

۲۶۶

۱-۴-۱۱ قابلیت اطمینان در سیستم کنترل پرواز

۲۶۶

۲-۴-۱۱ قابلیت اطمینان و تحمل‌پذیری خطا در

۲۷۰

رایانش ابری بلادرنگ

۲۷۰

۲-۱۱ معماری سیستم‌های بلادرنگ با قابلیت

۲۷۲

اتکای بالا

۲۷۲

۵-۱۱ معماری کلی

۲۷۴

۲-۵-۱۱ زمان‌بندی بلادرنگ

۲۷۴

۵-۱۱ زمان‌بندی چرخه‌ای

۲۷۵

۲-۵-۱۱ مدل زمان‌بندی همکاری

۲۷۵

۳-۵-۱۱ مدل زمان‌بندی پیشگیرانه

۲۷۵

۳-۵-۱۱ تکرار

۲۷۷

سوالات نمونه فصل یازدهم

فصل دوازدهم: تحمل‌پذیری خطا و وابستگی

۲۷۹

سخت‌افزاری

۱-۱۲ سخت‌افزارهای رای‌گیری در سیستم‌های

۲۷۹

تحمل‌پذیر خطا

۲۸۰

۱-۱۲ رای‌گیری‌های بیتی

۲۸۰

۲-۱۲ رای‌گیری‌های بیتی وزن‌دار

۲۸۱

۲-۱۲ طراحی توسط شبکه‌های انتخاب

۲۲۷

۴-۶-۹ تست رگرسیون

۲۲۹

۵-۶-۹ تست امنیت

۲۳۰

۶-۶-۹ تست پوشش

۲۳۰

۷-۷-۹ روش‌های آزمون

۲۳۱

۱-۷-۹ آزمون جعبه سیاه

۲۳۲

۲-۷-۹ آزمون پذیرش

۲۳۲

۳-۷-۹ دسته‌بندی مشابهات

۲۳۲

۴-۷-۹ تحلیل مقادیر مرزی

۲۳۲

۵-۷-۹ آزمون مانع‌ساز

۲۳۴

۶-۷-۹ آزمون ارا متعامد

۲۳۷

۷-۷-۹ آزمون جعبه سفید

۲۳۸

۸-۷-۹ آزمون مسیرهای مبنا

۲۳۸

۹-۷-۹ آزمون جعبه خاکستری

۲۳۸

سوالات نمونه فصل نهم

۲۳۸

فصل دهم: مدل‌سازی تحمل‌پذیری خطا

۲۳۹

رویکرد

۲۴۱

فضا-حالت

۲۴۲

۱-۱۰ مدل گراف

۲۴۴

۲-۱۰ تعاریف قابلیت اطمینان در شبکه

۲۴۵

۱-۲-۱۰ قابلیت اطمینان دو ترمینال

۲۴۵

۲-۲-۱۰ شمارش فضای حالت

۲۴۵

۳-۱۰ افزونگی ساخت‌یافته در مدل‌های فضا-

۲۴۹

حالت و شبکه‌های پتری

۲۵۰

۱-۳-۱۰ سیستم‌های LTI تکراری

۲۵۱

۲-۳-۱۰ معرفی سیستماتیک افزونگی

۲۵۲

۳-۳-۱۰ مدل خطا برای سیستم‌های LTI

۲۵۲

۴-۳-۱۰ مثال‌هایی از سیستم‌های متحمل خطای

۲۵۲

LTI

۲۵۵

۵-۳-۱۰ شبکه‌های پتری تکرار

۲۵۷

۶-۳-۱۰ معرفی سیستماتیک افزونگی

۲۵۸

۷-۳-۱۰ مدل خطا و نمونه‌ها

۲۵۹

سوالات نمونه فصل دهم

- ۲۸۱ ۴-۱-۱۲ رأی گیرهای کلمه‌ای
- ۲۸۲ ۲-۱۲ تکنیک‌های تزریق خطای سخت‌افزاری
- ۱-۲-۱۲ مقایسه روش‌های تزریق خطای
- ۲۸۳ سخت‌افزاری
- ۲۸۴ ۲-۲-۱۲ تکنیک‌های تزریق خطای نرم‌افزاری
- ۲۸۵ ۳-۱۲ ساختار جزئی‌تر سخت‌افزار
- ۴-۱۲ NMR جایگزینی مناسب برای TMR در
- ۲۹۱ تکنیک‌های سخت‌افزاری
- ۵-۱۲ بررسی طراحی‌های تحمل‌پذیر عیب در IC های
- ۲۹۳ حافظه
- ۱-۵-۱۲ کدهای تصحیح خطا (ECC)
- ۲-۵-۱۲ روش ترکیب افزونگی کدهای
- ۲۹۴ تصحیح خطا
- ۳-۵-۱۲ روش شرکت‌پذیری
- ۶-۱۲ استفاده از افزونگی سخت‌افزاری با ایده
- ۲۹۵ سیستم‌های بیولوژیکی
- ۱-۶-۱۲ تحمل‌پذیری خطا در آرایه‌ای از
- ۲۹۶ پردازنده‌ها
- ۱-۶-۱۲ اعمال ایده تحمل‌پذیری خطا به شیوه
- ۲۹۶ سیستم‌های بیولوژیکی
- ۲۹۷ سوالات نمونه فصل دوازدهم
- ۲۹۹ **مراجعه منتخب**

پیشگفتار

طراحی نرم افزارها، مطمئن یکی از تخصص های کلیدی مهندسين نرم افزار محسوب می شود. طراحی با قابلیت اطمینان بالا، عملکرد مطمئن می تواند کیفیت طراحی و توسعه نرم افزارهای سیستمی را ارتقا بخشد. در این راستا، این کتاب اصول و مراحل طراحی نرم افزارهای مطمئن برای مهندسين و توسعه دهندگان سیستم های نرم افزاری بسیار توصیه می شود. هدف از تالیف این کتاب، اول از همه ارائه مراحل توسعه نرم افزارهای مطمئن است. این مراحل می بایست به صورت زنجیره ای از اصول و قواعد طراحی در اختیار تیم توسعه نرم افزار قرار گیرد. دومین نکته جایگاه طراحی نرم افزار مطمئن در چرخه مهندسی نرم افزار است. این چرخه که به تدریج تغییر یافته و تکامل یافته است، تفاوتی از سیستم های کامپیوتری را شامل می شود، می بایست اطمینان پذیری را نیز در خود جای دهد. لذا طراحی نرم افزارهای مطمئن اصولی را ارائه خواهد داد که نه تنها با چرخه مهندسی نرم افزار کلاسیک تقابلی ندارد بلکه با آن تطابق کامل خواهد داشت و شاهراه ارتباطی مناسبی بین چرخه کلاسیک طراحی نرم افزار و اطمینان پذیری مورد نظر در نظر گرفته می شود. پس یکی از اصول بسیار مهم در تالیف این کتاب، ایجاد ارتباط معنایی با چرخه های مهندسی نرم افزار است که بیش از پیش از طراحی نرم افزارهای مطمئن برای مخاطبان ملموس می کند. این تطابق معنایی باعث کاربردی تر شدن طراحی نرم افزارهای مطمئن می شود که آثار مثبت آن در جای جای این کتاب قابل احساس است.

یکی از مشکلات در کتاب هایی از این دست، سطح بالای انتزاع و عدم پرداختن به جزئیات طراحی است که برای خوانندگان جزئیات سیستم را ناملموس می کند. در این کتاب سعی شده است تا راهکارهای اطمینان پذیری نرم افزار به بیانی ساده و منطبق با سرفصل های ارائه دروس استاندارد بیان شود. همچنین مثال های کاربردی بسیاری در خلال فصل های گوناگون به صورت پیوسته و منسجم بیان شده است. این انسجام در بیان نمونه ها به فهم موضوعات و مقایسه های ترکیبی کمک شایانی می نماید. در کنار ساده سازی مطالب و تنوع در بیان، همواره تلاش شده است تا مفاهیم بنیادی به صورت کامل و جامع در اختیار مخاطبان قرار گیرد.

مخاطبان

همان‌طور که بیان شد، این کتاب منطبق با سرفصل‌های استاندارد وزارت علوم، تحقیقات و فناوری تهیه شده است و از این حیث در نگاه اول مخاطبان آن دانشجویان مقاطع تحصیلات تکمیلی در حوزه مهندسی نرم‌افزار هستند. به علت پیوستگی مطالب و ایجاد رابطه متنی و معنایی با مراحل چرخه مهندسی نرم‌افزار، این کتاب برای کلیه دانشجویان مهندسی نرم‌افزار و همچنین طراحان حرفه‌ای نرم‌افزارهای کاربردی پیشنهاد می‌شود. کاربرد اصلی این کتاب در طراحی نرم‌افزاری سیستم‌های بلادرنگ، سیستم‌عامل‌های تعبیه‌شده، سیستم‌های با قابلیت اطمینان بالا، سیستم‌های کنترل از راه دور، امنیت نرم‌افزار و سیستم‌های توزیع‌شده است.

طراحی نرم‌افزارهای مطمئن برای کلیه توسعه‌دهندگان سیستم‌های نرم‌افزاری که در آن اطمینان‌پذیری در اولویت نخست است، توصیه می‌گردد. نکته دیگر مربوط به طرح‌های پژوهشی و دانشجویان تحصیلات تکمیلی است. این کتاب می‌تواند نقطه شروع خوبی برای پژوهش‌های بنیادی در حوزه سیستم‌های نرم‌افزار مطمئن و تکنیک‌های امنیت نرم‌افزار باشد. چرا که این کتاب با مطالعه چندین کتاب و مقاله مرتبط در این حوزه تالیف شده است و تمامی تعاریف و تکنیک‌های کلاسیک را شامل می‌شود. در نتیجه پژوهشگران این حوزه نیز از دیگر مخاطبان کتاب طراحی نرم‌افزارهای مطمئن محسوب می‌شوند و برای شروع طرح‌های پژوهشی مناسب خواهد بود.

پروژه‌های درسی

یکی از مسائل مهم در درس طراحی نرم‌افزارهای مطمئن به خصوص برای اساتید، تعریف پروژه‌های درسی مدون و مسائل مربوط به هر فصل است. در این کتاب لیستی از پروژه‌های کاربردی در انتهای هر فصل تعریف شده است. این پروژه‌ها در دو حوزه پژوهشی - طراحی نرم‌افزار دسته‌بندی شده‌اند که دانشجویان برای تجربه هر چه بهتر اصول طراحی نرم‌افزار مطمئن و همچنین تقویت مفاهیم آموزشی می‌توانند آنها را انتخاب نمایند. نکته مهم دیگر در طراحی این پروژه‌ها، نیازمندی‌های پیش از کار بوده که به‌طور کامل و با حمایتی بی‌نظیر در مفاهیم درسی همان‌طور بیان شده است.

تازه‌های این کتاب

این کتاب به صورت کاملاً کاربردی تهیه شده است که این موضوع در لایه‌لای مفاهیم کتاب به چشم می‌خورد. استفاده از مثال‌های کاربردی برای یادگیری مفاهیم درسی کمک بسیار زیادی به بالا بردن کیفیت آموزشی این درس می‌نماید. همچنین پیوستگی تکنیک‌های طراحی و مقایسه موضوعی آنها راحل‌هایی است که برای ارتقای کارایی فراگیران این درس به کار گرفته‌ایم. از دیدگاه مهندسی نرم‌افزار، جایگاه طراحی نرم‌افزارهای مطمئن در چرخه مهندسی نرم‌افزار به خوبی روشن شده است و مراحل تست و اصلاح خطاهای نرم‌افزاری را به تفصیل بیان کرده‌ایم. از نیازمندی‌های مهم این حوزه استفاده از پس‌زمینه‌های کاربردی و انتقال مفاهیم سطح بالا بود که با بحث‌های تکمیلی همچون تزریق

خطا، روش‌های شناسایی خطاهای حیاتی و تکنیک‌های آماری کشف خطا در فرآیند تست نرم‌افزار و پوشش دادیم.

ارتباط بین طراحی نرم‌افزارهای مطمئن با بستر سخت‌افزاری یکی دیگر از تازه‌های این کتاب به شمار می‌آید. اکثر کتاب‌های این حوزه در دنیای امروز، یا روی مفاهیم نرم‌افزاری و یا روی مفاهیم سخت‌افزاری و شبکه تأکید دارند. ما در این کتاب به بررسی آسیب‌های سخت‌افزاری و رابطه آن با مهندسی نرم‌افزار مطمئن پرداخته‌ایم. این موضوع علاوه بر رشد اطلاعات میدانی فراگیران، باعث می‌شود تا تأثیرات طراحی یک سیستم نرم‌افزاری مطمئن در تعامل با سخت‌افزار به خوبی روشن شود. این امر زمانی اهمیت پیدا می‌کند که هدف طراحی سیستم‌های تعبیه‌شده یا سیستم‌های عامل با قابلیت اطمینان بالا باشد. در این حوزه از علوم کامپیوتری، آنچه که بیش از همه اهمیت دارد، توجه به نیازمندی‌های روز این رشته به شمار می‌رود. در این کتاب سعی شده تا به نکات بیشتر از این حوزه مهم و حیاتی اشاره شود.

www.ketab.ir