

# 团 体 标 准

T/XXX XXXX—XXXX

## 5G 节点设备测试与诊断系统 嵌入式软件平台技术要求

5G node device testing and diagnosis system - Technical requirements for embedded  
software platform

(报批稿)

XXXX - XX - XX 发布

XXXX - XX - XX 实施

中国电子质量管理协会 发布

## 目 次

前言 .....	III
1 范围 .....	1
2 规范性引用文件 .....	1
3 术语和定义 .....	1
4 缩略语 .....	1
5 系统概述 .....	2
5.1 系统任务目标 .....	2
5.2 平台架构 .....	2
6 系统要求 .....	2
6.1 通用技术要求 .....	2
6.2 系统功能要求 .....	3
7 编制测试方案 .....	3
7.1 测试策略 .....	3
7.2 测试方法 .....	4
附录 A（规范性）GJB8114-2013《C/C++语言编程安全子集》 .....	8

## 前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中国电子质量管理协会软件与信息技术服务专业委员会提出。

本文件由中国电子质量管理协会归口。

本文件起草单位：国家工业信息安全发展研究中心、中国科学院微电子研究所、北京蓝威技术有限公司、中移（杭州）信息技术有限公司、重庆安标检测研究院有限公司、成都久信信息技术股份有限公司、中国电子质量管理协会、北京中标服检验技术研究院有限公司等。

本文件主要起草人：邓昌义、李志强、王潇耿、李岚清、张渊、胡通、陈晓、朱亚南、汪珺、朱平、孙林、李明时、孟嫣、吴玉平、卢涛、孙旭、张道庆、李慧强、龚关飞、游涌、朱义强、孙图南、王蕾、王文、阮进喜、周凤林、孙智江、张百川、梁峰、温章义、邢献红。

# 5G 节点设备测试与诊断系统 嵌入式软件平台技术要求

## 1 范围

本文件规定了5G节点设备测试与诊断系统嵌入式软件平台目标和任务、系统架构、系统功能模块组成以及测试方案。

本文件适用于5G节点设备测试与诊断系统，其他类型测试与诊断系统可参照执行。

## 2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GJB 8114-2013 C/C++语言编程安全子集

GB/T 11457-2006 信息技术 软件工程术语

GB/T 22239-2019 信息安全技术 网络安全等级保护基本要求

GB/T 25000.10-2006 系统与软件工程 系统与软件质量要求和评价 第10部分：系统与软件质量模型

GB/T 38674-2020 信息安全技术 应用软件安全编程指南

MISRA C:2012 工业软件C语言开发标准

MISRA C++:2018 工业软件C++语言开发标准

## 3 术语和定义

下列术语和定义适用于本文件。

### 3.1

**计算机软件配置项** computer software configuration item

为配置管理设计的软件的集合，它在配置管理过程中，作为单个实体对待。

### 3.2

**可靠性** reliability

在规定时间内和规定条件下，系统或部件执行所要求功能的能力。

### 3.3

**软件可维护性** software maintainability

软件系统或部件能修改以排除故障、改进性能或其他属性或适应变更了的环境的容易程度。

### 3.4

**冗余** redundancy

在容错系统中，系统中存在辅助的部件执行像其他元素同样的功能以防止失败或从失败恢复。

### 3.5

**应用软件** application software

设计用于实现用户的特定需要而非计算机本身问题的软件。

## 4 缩略语

以下缩略语适用于本文件。

3D：三维（three-dimensional）

5G：第五代移动通信技术（5th Generation Mobile Communication Technology）

API：应用程序接口（Application Programming Interface）

EDA：电子设计自动化（Electronics Design Automation）

PCB：印制电路板（Printed Circuit Board）

## 5 系统概述

### 5.1 系统任务目标

5G节点设备测试与诊断系统 嵌入式软件平台的关键技术研究主要为针对5G基站，网络架构设备和轻量化应用终端等节点设备关键器件的板载测试与诊断需求，基于嵌入式芯片、开展5G通信接口类、轻量化终端芯片类、射频微波类、功率类、传感类等应用电路进行测试与诊断电路扩展辅助设计以及PCB建模。

### 5.2 系统架构

嵌入式软件平台流程图详见图1。

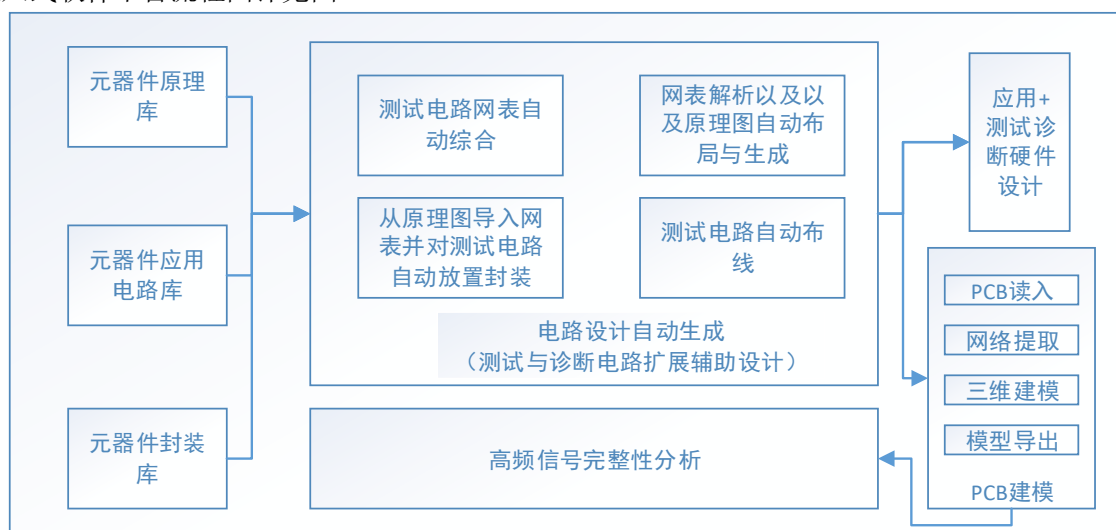


图 1 嵌入式软件平台流程图

电路设计以及建模平台基于5G基站、网络架构设备和轻量化应用终端等节点设备的关键元器件基础库（包括原理库、封装库、应用电路库），通过读取用户设计的原始电路原理图，结合嵌入式硬件平台的测试性建模模块优化后的测试点布局自动添加对应测试电路，通过电路自动生成模块生成带有测试电路的应用电路的原理图，并通过PCB自动布局布线模块生成附加测试电路PCB版图，导出优化后的PCB设计文件，可用于板级制造，或通过PCB建模模块生成模型，供其他仿真平台进行使用（可靠性仿真平台）。

## 6 系统要求

### 6.1 通用技术要求

#### 6.1.1 一般要求

- 可扩展性：对现有软件系统影响最小的情况下，软件平台功能持续扩展的能力。
- 可靠性：软件产品质量属性之可靠性特性，元件、产品、系统在一定时间内、在一定条件下无故障地执行指定功能的能力或可能性。又包含子特性：成熟性、容错性、易恢复性。
- 安全保密性：系统保护信息和数据的程度，以使系统具有与其授权类型和授权级别一致的数据访问度；确保数据只有在被授权时才能被访问的程度。
- 易用性：主要包括易学性、易操作性、易访问性、可辨识度。
- 兼容性：主要指软件在不同版本之间的兼容性。有两类基本的兼容性测试：向下兼容和交错兼容。

- f) 余量要求：要求软件存储余量大于等于 20%。
- g) 强度要求：要求软件在不正常到发生故障的情况下，不会崩溃。
- h) 时间特性。

### 6.1.2 增强要求

- a) 支持测试电路设计自动生成；
- b) 建立 5G 节点设备使用的元器件驱动程序库；
- c) 具有状态监测驱动程序参数化配置；
- d) 支持国产的操作系统和数据库；
- e) 具备国内外 EDA 设计软件接口，支持国内外热学、力学、电学等多物理场仿真软件接口不少于 10 种。

## 6.2 系统功能要求

系统功能模块及各模块功能描述见表 1。

表 1 系统功能模块功能描述

序号	功能模块	功能描述
1	测试性模型创建	a) 建立测试模型。浏览、编辑、查看模型文件的属性信息，并通过导入导出等功能完成模型文件的管理和合并等任务。
2	模型编辑	a) 支持对基本图元（连线、基本模块、可配置模块、开关、与门、测试点、功能节点、注释）进行增加、修改与删除操作； b) 支持图元的快速查找及定位，并且可以对图元的参数进行编辑； c) 模型设计界面的图元进行自动布局。
3	高级属性	a) 支持功能信号的增加、关联（关联）、映射及阻断； b) 支持多工作模式、多系统模式设定。
4	模型信息查看及保存	a) 支持模型结构树、系统日志快速查看，模型编辑窗口切换； b) 支持模型保存、另存及自动备份； c) 选择连线高亮显示等。
5	模型信息融合	a) 快捷、便利的模型导入功能； b) HTML 格式、图片格式的模型输出功能。
6	模型分析	a) 支持系统级模型快速分析； b) 支持多测试筛选、多模式、多层次的测试性分析。
7	分析结果及输出	a) 系统、合理的模型相关信息统计与输出； b) 多模糊度故障检测率、故障隔离率计算； c) 表格形式、XML 格式 D 矩阵输出及快速查看； d) 表格形式、图形格式、XML 格式的故障树查看。
8	DFT 模式	a) 快速追踪信号传递路径； b) 故障与测试关联关系图形显示。
9	RDS 功能模块	a) 对产品的基本信息、测试知识、故障知识进行查看、编辑、管理； b) 实现对已知产品的故障测试与诊断； c) 支持网络远程诊断。
10	自定义设置	a) 图元及界面显示自定义设置； b) 系统层次标签、模块分类、测试标签、开关模式及系统配置自定义设置。

## 7 编制测试方案

### 7.1 测试策略

测试方法采取动态测试方法和静态测试方法相结合的方式。动态测试方法选取功能测试、性能测试、接口测试、安全性测试、可靠性测试、可维护性测试、兼容性测试、人机交互界面测试、安装性测试和余量测试等测试类型；静态测试选取文档审查、代码审查、代码走查和静态分析四种测试类型。

- a) 静态测试策略

主要采用商业测试工具对被测源程序进行代码审查和静态分析,并结合人工的方法依据设计文档对源程序及代码审查结果进行分析。对程序存在的内存泄漏、数组越界等潜在缺陷,分类分等级给出问题报告,并依据国军标及行业标准规范给出软件质量度量分析报告。

#### b) 动态测试策略

- 功能测试,采用等价类划分、边界值分析、分解组合、因果图等测试技术全面覆盖研制总要求及软件需求规格说明的要求,并充分考虑多种因素的组合和异常情况;
- 性能测试,根据性能指标要求,设计多种性能测试场景,特别是复杂情况下的性能测试场景,采用性能或压力测试工具进行性能测试,验证性能指标符合时间及精度等要求;
- 接口测试,采用接口测试工具模拟陪测软件或外部接口信息,根据协议测试所有软件接口的正确性,包括接口的异常报文测试;
- 首轮测试策略:对软件研制任务书和软件需求规格说明规定的功能、性能等需求进行验证;
- 回归测试策略:针对软件在其他活动中暴露的问题进行影响域分析,并在回归测试中一并进行测试覆盖;在测试环境不具备或因其他原因导致测试用例无法实施的情况下,采用代码审查的方式进行测试覆盖。

## 7.2 测试方法

### 7.2.1 通用技术要求测试方法

#### 一般要求

##### a) 可扩展性测试

- 确定系统随着工作负载的增长而扩展;
- 确定系统在用户方面的限制;
- 确定客户端条件和最终用户在各种负载下对产品的体验;
- 确定系统的健壮性。

##### b) 可靠性测试

- 成熟性测试

——基于系统运行剖面设计测试用例,根据实际使用的概率分布随机选择输入,运行系统,测试系统满足需求的程度并获取失效数据,其中包括对重要输入变量值的覆盖、对相关输入变量可能组合的覆盖、对设计输入空间与实际输入空间之间区域的覆盖、对各种使用功能的覆盖、对使用环境的覆盖;

——测试系统的平均无故障时间;

——选择可靠性增长模型,通过检测到的失效数和故障数,对系统的可靠性进行预测。

- 容错性测试

——测试系统对中断发生的反应;

——测试系统在边界条件下的反应;

——测试系统的功能、性能的降级情况;

——测试系统的各种误操作模式;

——测试系统的各种故障模式;

——测试在多机系统出现故障需要切换时系统的功能和性能连续平稳性。

- 易恢复性测试

——测试具有自动修复功能的系统的自动修复的时间;

——测试软件在特定的时间范围内的平均宕机时间;

——测试软件在特定的时间范围内的平均恢复时间;

——测试软件的可重启并维持提供服务的能力;

——测试软件的还原功能的还原能力。

##### c) 安全保密性测试

- 测试软件防止非法操作的模式,包括防止非法授权的创建、删除或修改程序或信息,必要时做强化异常操作的测试;
- 测试软件防止数据被讹误和被破坏的能力;

- 测试软件的加密和解密功能。
- d) 易用性测试
  - 易学性
    - 可测试软件的在线帮助，确认在线帮助是否容易定位，是否有效；还可对照用户手册或操作手册执行系统，测试用户文档的有效性。
  - 易操作性
    - 输入数据，确认软件是否对输入数据进行有效性检查；
    - 要求具有中断执行的功能，确认它们能否在动作完成之前被取消；
    - 要求具有还原能力的功能，确认它们能否在动作完成之后被撤销；
    - 包含参数设置的功能，确认参数是否易于选择、是否有缺省值；
    - 要求具有解释的消息，确认它们是否明确；
    - 要求具有界面提示能力的界面元素，确认它们是否有效；
    - 要求具有容错能力的功能和操作，确认系统能否提示错误的风险、能否容易纠正错误的输入、能否从错误中恢复；
    - 要求具有定制能力的功能和操作，确认定制能力的有效性；
    - 要求具有运行状态监控能力的功能，确认它们的有效性。
  - 易访问性
    - 测试软件是否易于各种用户群体访问。
  - 可辨识性
    - 测试软件的各项功能及界面中的输入输出项是否易于识别。包括界面文字描述完整清晰，不含不易理解的词语，便于用户辨识软件是否符合他们的要求；软件提供产品介绍演示功能，便于用户辨识软件是否符合他们的要求。
- e) 兼容性测试
  - 向下兼容是测试软件新版本保留它早期版本的功能的情况；
  - 交错兼容测试是测试共同存在的两个相关但不同的产品之间的兼容性。
- f) 余量测试
  - 测试软件存储余量是否大于等于 20%。
- g) 强度测试：测试软件在以下不正常到发生故障的情况下，可以运行到何种程度：
  - 提供最大处理的信息量；
  - 提供数据能力的饱和实验指标；
  - 提供最大存储范围；
  - 在能力降级时进行测试；
  - 在人为错误状态下进行软件反应的测试；
  - 进行持续一段规定的时间，而且连续不能中断的测试。
- h) 时间特性测试
  - 测试系统的影响时间、平均响应时间、响应极限时间，系统的吞吐量、平均吞吐量、极限吞吐量，系统的周转时间、平均周转时间、周转时间极限；
  - 在测试时，应标识和定义适合于软件应用的任务，并对多项任务进行测试。

### 增强要求

- a) 支持测试电路设计自动生成
  - 输入电路设计要求，测试“测试电路设计”是否可以自动生成。
- b) 建立5G节点设备使用的元器件驱动程序库；
  - 测试是否建立了元器件驱动程序库；
  - 测试库中数据是否可以被5G节点设备使用。
- c) 具有状态监测驱动程序参数化配置；
  - 验证状态监测驱动程序的参数化配置功能，确保可以通过配置生成定制驱动程序源代码。
  - 测试生成的驱动程序在实际硬件环境中的功能。
- d) 支持国产的操作系统和数据库；

- 测试软件在国产操作系统上能够正常安装和运行。
  - 测试软件与国产数据库的兼容性，确保数据存储和管理功能正常。
- e) 具备国内外 EDA 设计软件接口，支持国内外热学、力学、电学等多物理场仿真软件接口不少于 10 种。
- 使用多种国内外 EDA 设计软件和多物理场仿真软件进行接口测试，确保数据传输的正确性和完整性；
  - 通过实际项目案例，验证接口功能，确保数据在不同软件之间的交互正常。

## 7.2.2 系统功能要求测试方法

测试项与系统功能模块对应关系如下**错误!未找到引用源。**。

表 2 测试项与系统功能模块对应关系

序号	功能模块	测试项	测试点
1	测试性模型创建	测试性模型文件属性编辑测试	a) 系统支持对创建的测试性模型文件属性进行浏览查看功能，同时能够对文件的属性进行编辑。
		测试性模型管理测试	a) 系统支持对测试性模型的导入、导出和删除功能； b) 在导入过程中如果文件格式不正确或者解析过程中文件内容异常，系统能够给出异常提示，并且终止导入操作；若导入过程中系统存在与导入模型文件属性一致的文件，系统会在用户选择提示继续导入的情况下对模型进行合并，或者放弃导入； c) 系统支持按特定格式导出用户选择的模型文件，并且能够在导出过程中对文件进行校验，若校验结果错误，系统支持最多 3 次的自动重新导出； d) 系统支持对用户选择的模型文件进行删除的功能，在删除过程中需要弹框提示。
2	模型编辑	模型基本图元管理测试	a) 系统支持对模型的基本图元（连线、基本模块、可配置模块、开关、与门、测试点、功能节点、注释）进行增加、修改与删除的操作，同时对本模型进行保存。
		模型基本图元查询测试	a) 系统支持对模型的基本图元按照名称、编号等方式进行查询的功能，并将符合要求的结果进行统一显示。
		模型基本图元自动布局测试	a) 系统在模型设计过程中，能够根据算法设计要求或者设置的规则对界面的图元进行位置和大小自动调整，以达到更好的可视化效果和用户体验。
3	高级属性	功能信号管理测试	a) 系统支持对功能信号的增加、关联（关联）、映射及阻断功能。
		工作模式设定测试	a) 系统支持多工作模式、多系统模式的设定。
4	模型信息查看及保存	模型信息展示查看测试	a) 系统支持对模型以结构树的方式进行展示，并且能够快速查看模型信息； b) 系统能够对记录的系统日志进行查看； c) 系统能够在多个编辑模型的窗口之间切换。
		模型保存、备份测试	a) 系统支持对新增、编辑后的模型进行手动保存，同时支持对模型进行另存； b) 系统支持对保存后的模型进行自动备份。
		基本图元高亮显示测试	a) 系统能够对用户选择的连线、基础模块、开关、与门等基本图元进行高亮显示。
5	模型信息融合	模型导入测试	a) 系统支持对模型的导入。
		模型导出融合测试	a) 系统支持对多个模型的输出结果进行融合。模型信息融合的方法包括投票法、加权平均法和堆叠法。

			输出的模型格式包含 HTML 格式和图片格式。
6	模型分析	模型快速分析测试	a) 系统支持对系统级模型快速分析功能。
		测试性分析测试	a) 系统支持多测试筛选、多模式、多层次的测试性分析。
7	分析结果及输出	模型信息统计输出测试	a) 系统支持对模型信息进行统计及输出。
		故障检测率计算测试	a) 系统能够通过多模糊度的方式对故障检测率进行计算。
		故障隔离率计算测试	a) 系统能够通过多模糊度的方式对故障隔离率进行计算。
		分析结果展示测试	a) 系统能够对模型分析结果以表格、XML、D 矩阵的快速查看的方式展示； b) 系统能够对模型分析结果以表格、图形格式、XML 格式诊断树的方式展示查看。
8	DFT 模式	DFT 模式测试	a) 系统能够快速追踪信号的传递路径，并且将检测的故障与测试关联关系以关系图的形式进行显示。
9	RDS 功能模块	产品相关信息管理测试	a) 系统支持对本产品的主要信息包括硬件配置、软件信息以及产品功能、组成等信息进行编辑、展示的功能； b) 系统支持对本产品支撑的测试类型、测试范围以及测试结果等相关信息的编辑、查询、删除功能； c) 系统支持对测试故障信息进行分析显示。
		产品故障检测测试	a) 系统支持对检测范围内的产品进行故障检测，对检测结果不通过的测试项目进行故障诊断，并提供常规可能的故障原因。
		支持网络远程诊断测试	a) 系统支持网络远程登录，并且对检测结果进行远程诊断。
10	自定义设置	图元界面显示自定义设置测试	a) 系统支持对图元的显示和界面显示的风格进行自定义设置。
		系统标签和配置自定义设置测试	a) 系统支持系统层次标签、模块分类、测试标签、开关模式及系统配置自定义设置。

附 录 A  
(规范性)

A.1 GJB8114-2013 《C/C++语言编程安全子集》

序号	规则描述	规则号	等级	适用语言
1	禁止通过宏定义改变关键字和基本类型含义。	GJB8114-R-1-1-1	重要	C/C++
2	禁止将其他标识宏定义为关键字和基本类型	GJB8114-R_1_1_2	重要	C/C++
3	用 typedef 自定义的类型禁止被重新定义。	GJB8114-R_1_1_3	重要	C/C++
4	禁止重新定义 C 或 C++ 的关键字	GJB8114-R_1_1_4	重要	C/C++
5	禁止 #define 被重复定义。	GJB8114-R_1_1_5	重要	C/C++
6	函数中的#define 和#undef 必须配对使用	GJB8114-R_1_1_6	重要	C/C++
7	以函数形式定义的宏，参数和结果必须用括号括起来	GJB8114-R_1_1_7	重要	C/C++
8	结构、联合、枚举的定义中必须定义标识名	GJB8114-R_1_1_8	重要	C/C++
9	结构体定义中禁止含有无名结构体	GJB8114-R_1_1_9	重要	C/C++
10	位定义的有符号整型变量位长必须大于 1	GJB8114-R_1_1_10	重要	C/C++
11	位定义的整型变量必须明确定义是有符号还是无符号的	GJB8114-R_1_1_11	重要	C/C++
12	位定义的变量必须是同长度的类型且定义位禁止跨越类型的长度	GJB8114-R_1_1_12	重要	C/C++
13	函数声明中必须对参数类型进行声明，并带有变量名	GJB8114-R_1_1_13	重要	C/C++
14	函数声明必须与函数原型一致	GJB8114-R_1_1_14	重要	C/C++
15	函数中的参数必须使用类型声明	GJB8114-R_1_1_15	重要	C/C++
16	外部声明的变量，类型必须与定义一致	GJB8114-R_1_1_16	重要	C/C++
17	禁止在函数体内使用外部声明	GJB8114-R_1_1_17	重要	C/C++
18	数组定义禁止没有显式的边界限定	GJB8114-R_1_1_18	重要	C/C++
19	禁止使用 extern 声明对变量初始化	GJB8114-R_1_1_19	重要	C/C++
20	用于数值计算的字符型变量必须明确定义是有符号还是无符号	GJB8114-R_1_1_20	重要	C/C++
21	禁止在 #include 语句中使用绝对路径	GJB8114-R_1_1_21	重要	C/C++
22	禁止头文件重复包含	GJB8114-R_1_1_22	重要	C/C++
23	函数参数表为空时，必须使用 void 明确说明	GJB8114-R_1_1_23	重要	C/C++
24	禁止在头文件前有可执行代码	GJB8114-R_1_2_3	重要	C/C++

25	引起二义性理解的逻辑表达式，必须使用括号显式说明优先级顺序	GJB8114-R_1_2_4	重要	C/C++
26	禁止指针的指针超过两级	GJB8114-R_1_3_1	重要	C/C++
27	函数指针的使用必须加以&明确说明	GJB8114-R_1_3_2	重要	C/C++
28	禁止对参数指针进行赋值	GJB8114-R_1_3_3	重要	C/C++
29	禁止将局部变量地址做为函数返回值返回	GJB8114-R_1_3_4	重要	C/C++
30	禁止使用或释放未分配空间或已被释放的指针	GJB8114-R_1_3_5	重要	C/C++
31	指针变量被释放后必须置为 NULL	GJB8114-R_1_3_6	重要	C/C++
32	动态分配的指针变量定义时如未被分配空间必须初始化为 NULL	GJB8114-R_1_3_7	重要	C/C++
33	动态分配的指针变量第一次使用前必须进行是否为 NULL 的判别	GJB8114-R_1_3_8	重要	C/C++
34	空指针必须使用 NULL，禁止使用整型数 0	GJB8114-R_1_3_9	重要	C/C++
35	禁止文件指针在退出时没有关闭文件	GJB8114-R_1_3_10	重要	C/C++
36	在 if-elseif 语句中必须使用 else 分支	GJB8114-R_1_4_1	重要	C/C++
37	条件判定分支如果为空，必须以单独一行的分号加注释进行明确说明	GJB8114-R_1_4_2	重要	C/C++
38	禁止使用空 switch 语句。	GJB8114-R_1_4_3	重要	C/C++
39	禁止对 bool 量使用 switch 语句。	GJB8114-R_1_4_4	重要	C/C++
40	禁止 switch 语句中只包含 default 语句。	GJB8114-R_1_4_5	重要	C/C++
41	除枚举类型列举完全外，switch 必须要有 default	GJB8114-R_1_4_6	重要	C/C++
42	switch 中的 case 和 default 必须以 break 或 return 终止，共用 case 必须加以明确注释。	GJB8114-R_1_4_7	重要	C/C++
43	switch 语句的所有分支必须具有相同的层次范围	GJB8114-R_1_4_8	重要	C/C++
44	禁止从复合语句外 goto 到复合语句内，或由下向上 goto	GJB8114-R_1_5_1	重要	C/C++
45	禁止使用 setjmp/longjmp	GJB8114-R_1_5_2	重要	C/C++
46	禁止将浮点常数赋给整型变量。	GJB8114-R_1_6_1	重要	C/C++
47	禁止将越界整数赋给整型变量。	GJB8114-R_1_6_2	重要	C/C++
48	禁止在逻辑表达式中使用赋值语句。	GJB8114-R_1_6_3	重要	C/C++
49	禁止对逻辑表达式进行位运算。	GJB8114-R_1_6_4	重要	C/C++
50	禁止在运算表达式中或函数调用参数中使用 ++ 或 -- 操作符。	GJB8114-R_1_6_5	重要	C/C++
51	对变量进行移位运算禁止超出变量长度。	GJB8114-R_1_6_6	重要	C/C++
52	禁止移位操作中的移位数为负数。	GJB8114-R_1_6_7	重要	C/C++
53	数组禁止越界使用。	GJB8114-R_1_6_8	重要	C/C++

54	数组下标必须是大于等于零的整型数。	GJB8114-R_1_6_9	重要	C/C++
55	禁止对常数值做逻辑非的运算。	GJB8114-R_1_6_10	重要	C/C++
56	禁止非枚举类型变量使用枚举类型的值。	GJB8114-R_1_6_11	重要	C/C++
57	除法运算中禁止被零除。	GJB8114-R_1_6_12	重要	C/C++
58	禁止在 sizeof 中使用赋值。	GJB8114-R_1_6_13	重要	C/C++
59	缓存区读取操作禁止越界。	GJB8114-R_1_6_14	重要	C/C++
60	缓存区写入操作禁止越界。	GJB8114-R_1_6_15	重要	C/C++
61	禁止使用已被释放了的内存空间。	GJB8114-R_1_6_16	重要	C/C++
62	被 free 的指针必须指向最初 malloc、calloc 分配的地址。	GJB8114-R_1_6_17	重要	C/C++
63	禁止使用 gets 函数，应使用 fgets 函数替代	GJB8114-R_1_6_18	重要	C/C++
64	使用字符串赋值、拷贝、追加等函数时，禁止目标字符串存储空间越界。	GJB8114-R_1_6_19	重要	C/C++
65	禁止覆盖标准函数库的函数。	GJB8114-R_1_7_1	重要	C/C++
66	禁止函数的实参和形参类型不一致。	GJB8114-R_1_7_2	重要	C/C++
67	实参与形参的个数必须一致。	GJB8114-R_1_7_3	重要	C/C++
68	禁止使用旧形式的函数参数表定义形式。	GJB8114-R_1_7_4	重要	C/C++
69	函数声明和函数定义中的参数类型必须一致。	GJB8114-R_1_7_5	重要	C/C++
70	函数声明和函数定义中的返回类型必须一致。	GJB8114-R_1_7_6	重要	C/C++
71	有返回值的函数必须通过返回语句返回。	GJB8114-R_1_7_7	重要	C/C++
72	禁止无返回值函数的返回语句带有返回值。	GJB8114-R_1_7_8	重要	C/C++
73	有返回值函数的返回语句必须带有返回值。	GJB8114-R_1_7_9	重要	C/C++
74	函数返回值的类型必须与定义一致。	GJB8114-R_1_7_10	重要	C/C++
75	具有返回值的函数，其返回值如果不被使用，调用时应有 (void) 说明。	GJB8114-R_1_7_11	重要	C/C++
76	无返回值的函数，调用时禁止再用 (void) 重复说明。	GJB8114-R_1_7_12	重要	C/C++
77	静态函数必须被使用。	GJB8114-R_1_7_13	重要	C/C++
78	禁止同一个表达式中调用多个顺序相关函数。	GJB8114-R_1_7_14	重要	C/C++
79	禁止在函数参数表中使用省略号。	GJB8114-R_1_7_15	重要	C/C++
80	禁止使用直接或间接自调用函数。	GJB8114-R_1_7_16	重要	C/C++
81	禁止不可达语句。	GJB8114-R_1_8_1	重要	C/C++
82	禁止不可达分支。	GJB8114-R_1_8_2	重要	C/C++

83	禁止使用无效语句。	GJB8114-R_1_8_3	重要	C/C++
84	使用八进制数必须明确注释。	GJB8114-R_1_8_4	重要	C/C++
85	数字类型后缀必须使用大写字母。	GJB8114-R_1_8_5	重要	C/C++
86	for 循环控制变量必须使用局部变量。	GJB8114-R_1_9_1	重要	C/C++
87	for 循环控制变量必须使用整数型变量。	GJB8114-R_1_9_2	重要	C/C++
88	禁止在 for 循环体内部修改循环控制变量。	GJB8114-R_1_9_3	重要	C/C++
89	无限循环必须使用 while (1) 语句，禁止使用 for (,;) 等其他形式的语句	GJB8114-R_1_9_4	重要	C/C++
90	浮点数变量赋给整型变量必须强制转换	GJB8114-R_1_10_1	重要	C/C++
91	长整数变量赋给短整数变量必须强制转换	GJB8114-R_1_10_2	重要	C/C++
92	double 型变量赋给 float 型变量必须强制转换	GJB8114-R_1_10_3	重要	C/C++
93	指针变量的赋值类型必须与指针变量类型一致	GJB8114-R_1_10_4	重要	C/C++
94	将指针量赋予非指针变量或非指针量赋予指针变量，必须使用强制转换	GJB8114-R_1_10_5	重要	C/C++
95	禁止使用无实质作用的类型转换。	GJB8114-R_1_10_6	重要	C/C++
96	变量禁止未赋值就使用。	GJB8114-R_1_11_1	重要	C/C++
97	变量初始化禁止隐含依赖于系统的缺省值。	GJB8114-R_1_11_2	重要	C/C++
98	结构体初始化的嵌套结构必须与定义一致。	GJB8114-R_1_11_3	重要	C/C++
99	枚举元素定义中的初始化必须完整。	GJB8114-R_1_11_4	重要	C/C++
100	禁止对逻辑量进行大于或小于的逻辑比较。	GJB8114-R_1_12_1	重要	C/C++
101	禁止对指针进行大于或小于的逻辑比较。	GJB8114-R_1_12_2	重要	C/C++
102	禁止对浮点数进行是否相等的比较。	GJB8114-R_1_12_3	重要	C/C++
103	禁止对无符号数进行大于等于零或小于零的比较。	GJB8114-R_1_12_4	重要	C/C++
104	禁止无符号数与有符号数之间的直接比较。	GJB8114-R_1_12_5	重要	C/C++
105	禁止局部变量与全局变量同名。	GJB8114-R_1_13_1	重要	C/C++
106	禁止函数形参与全局变量同名。	GJB8114-R_1_13_2	重要	C/C++
107	禁止变量名与函数名同名。	GJB8114-R_1_13_3	重要	C/C++
108	禁止变量名与标识名同名。	GJB8114-R_1_13_4	重要	C/C++
109	禁止变量名与枚举元素同名。	GJB8114-R_1_13_5	重要	C/C++
110	禁止变量名与 typedef 自定义的类型名同名。	GJB8114-R_1_13_6	重要	C/C++
111	禁止在内部块中重定义已有的变量名。	GJB8114-R_1_13_7	重要	C/C++

112	禁止仅依赖大小写区分的变量。	GJB8114-R_1_13_8	重要	C/C++
113	禁止仅依赖小写字母“l”与数字“1”区分的变量。	GJB8114-R_1_13_9	重要	C/C++
114	禁止仅依赖大写字母“O”与数字“0”区分的变量。	GJB8114-R_1_13_10	重要	C/C++
115	禁止单独使用小写字母“l”或大写字母“O”作为变量名。	GJB8114-R_1_13_11	重要	C/C++
116	程序外部可改写的变量，必须使用 volatile 类型说明。	GJB8114-R_1_13_12	重要	C/C++
117	禁止在表达式中出现多个同一 volatile 类型变量的运算。	GJB8114-R_1_13_13	重要	C/C++
118	禁止将 NULL 做为整型数 0 使用。	GJB8114-R_1_13_14	重要	C/C++
119	禁止给无符号类型变量赋负值。	GJB8114-R_1_13_15	重要	C/C++
120	用于表示字符串的数组必须以 ‘\0’ 结束。	GJB8114-R_1_13_16	重要	C/C++
121	含有动态分配成员的类，必须编写拷贝构造函数，并重载赋值操作符。	GJB8114-R_2_1_1	重要	C++
122	虚拟基类指针转换为派生类指针必须使用 d"ami 处 cast 转换。	GJB8114-R_2_1_2	重要	C++
123	菱形层次结构的派生设计，对基类派生必须使用 virtual 说明。	GJB8114-R_2_1_3	重要	C++
124	抽象类中的复制操作符重载必须是保护的或私有的。	GJB8114-R_2_1_4	重要	C++
125	构造函数中禁止使用全局变量。	GJB8114-R_2_2_1	重要	C++
126	类中必须明确定义缺省构造函数。	GJB8114-R_2_2_2	重要	C++
127	单参数构造函数必须使用 explicit 声明。	GJB8114-R_2_2_3	重要	C++
128	类中所有成员变量必须在构造函数中初始化。	GJB8114-R_2_2_4	重要	C++
129	派生类构造函数必须在初始化列表中说明直接基类构造函数。	GJB8114-R_2_2_5	重要	C++
130	具有虚拟成员函数的类，析构函数必须是虚拟的。	GJB8114-R_2_3_1	重要	C++
131	析构函数中禁止存在不是由自身捕获处理的异常。	GJB8114-R_2_3_2	重要	C++
132	基类虚拟函数的参数缺省值在派生类重写函数中禁止被改变。	GJB8114-R_2_4_1	重要	C++
133	派生类对基类虚拟函数重写的声明必须使用 virtual 显示说明。	GJB8114-R_2_4_2	重要	C++
134	禁止非纯虚函数被纯虚函数重写。	GJB8114-R_2_4_3	重要	C++
135	禁止将不相关的指针类型强制转换为对象指针类型。	GJB8114-R_2_5_1	重要	C++
136	指针或引用的类型转换中禁止移除 const 或 volatile 属性。	GJB8114-R_2_5_2	重要	C++
137	使用 new 分配的内存空间，用完后必须使用 delete 释放。	GJB8114-R_2_6_1	重要	C++
138	必须使用 delete[] 释放 new[] 分配的内存空间。	GJB8114-R_2_6_2	重要	C++
139	被 delete 的指针必须指向最初 new 分配的地址。	GJB8114-R_2_6_3	重要	C++
140	函数中固定长度数组变量的传递必须使用引用方式。	GJB8114-R_2_7_1	重要	C++

141	定义为 <code>const</code> 的成员函数禁止返回非 <code>const</code> 的指针或引用。	GJB8114-R_2_7_1	重要	C++
142	禁止可导致非资源性对象数据被外部修改的成员函数返回。	GJB8114-R_2_7_3	重要	C++
143	捕获的顺序必须按由派生类到基类的次序排序。	GJB8114-R_2_8_1	重要	C++
144	每个指定的抛出必须有与之匹配的捕获。	GJB8114-R_2_8_2	重要	C++
145	异常抛出的对象必须使用引用方式捕获。	GJB8114-R_2_8_3	重要	C++
146	缺省捕获必须放在所有指定捕获之后。	GJB8114-R_2_8_4	重要	C++
147	禁止显式直接抛出 <code>NULL</code> 。	GJB8114-R_2_8_5	重要	C++
148	模板的声明、定义与实现必须在同一个文件之中。	GJB8114-R_2_9_1	重要	C++