

1. 软件测试英语专业词汇

2. NLV: Nation Language Version 本地化版本

3. FVT: Functional Verification Testing 功能验证测试

4. TVT: Translation Verification Testing 翻译验证测试

5. SVT: System Verification Testing 系统验证测试

6. fault——故障

在软件中一个错误的表现。

7. feasible path——可达路径

可以通过一组输入值和条件执行到的一条路径。

8. feature testing——特性测试

参考功能测试 (Functional Testing)

9. FMEA——失效模型效果分析 (Failure Modes and Effects Analysis)

可靠性分析中的一种方法，用于在基本组件级别上确认对系统性能有重大影响的失效

10. FMECA——失效模型效果关键性分析 (Failure Modes and Effects Criticality Analysis)

FMEA 的一个扩展，它分析了失效结果的严重性。

11. FTA——故障树分析 (Fault Tree Analysis)

引起一个不需要事件产生的条件和因素的确认和分析，通常是严重影响系统性能、经济性、安全性或其它需要特性。

12. functional decomposition——功能分解

参考模块分解 (modular decomposition)

13. **Functional Specification** —— 功能规格说明书

一个详细描述产品特性的文档。

14. **Functional Testing** —— 功能测试

测试一个产品的特性和可操作行为以确定它们满足规格。

15. **glass box testing** —— 玻璃盒测试

参考白盒测试（White Box Testing）

16. **IEEE** —— 美国电子与电器工程师学会（Institute of Electrical and Electronic Engineers）

17. **incremental testing** —— 渐增测试

集成测试的一种，组件逐渐被增加到系统中直到整个系统被集成。

18. **infeasible path** —— 不可达路径

不能够通过任何可能的输入值集合执行到的路径。

19. **input domain** —— 输入域

所有可能输入的集合。

20. **inspection** —— 检视

对文档进行的一种评审形式。

21. **installability testing** —— 可安装性测试

确定系统的安装程序是否正确的测试。

22. **instrumentation** —— 插桩

在程序中插入额外的代码以获得程序在执行时行为的信息。

23. **instrumenter** —— 插装器

执行插装的工具

24. **Integration Testing**——集成测试

测试一个应用组合后的部分以确保它们的功能在组合之后正确。

该测试一般在单元测试之后进行。

25. **interface**——接口

两个功能单元的共享边界。

26. **interface analysis**——接口分析

分析软件与硬件、用户和其它软件之间接口的需求规格。

27. **interface testing**——接口测试

测试系统组件间接口的一种测试。

28. **invalid inputs**——无效输入

在程序功能输入域之外的测试数据。

29. **isolation testing**——孤立测试

组件测试（单元测试）策略中的一种，把被测组件从其上下文组件之中孤立出来，通过设计驱动和桩进行测试的一种方法。

30. **job control language**——工作控制语言

用于确定工作顺序，描述它们对操作系统要求并控制它们执行的语言。

31. **LCSAJ**——线性代码顺序和跳转（**Linear Code Sequence And Jump**）

包含三个部分：可执行语句线性顺序的起始，线性顺序的结束，在线性顺序结束处控制流跳转的目标语句。

32. **LCSAJ coverage**——LCSAJ 覆盖

在组件中被测试执行到的 LCSAJ 的百分比。

33. LCSAJ testing —— LCSAJ 测试

根据 LCSAJ 设计测试用例的一种技术。

34. Load Testing —— 负载测试

通过测试系统在资源超负荷情况下的表现，以发现设计上的错误或验证系统的负载能力。

35. logic analysis —— 逻辑分析

- (1) 评价软件设计的关键安全方程式、算法和控制逻辑的方法。
- (2) 评价程序操作的顺序并且检测可能导致灾难的错误。

36. logic-coverage testing —— 逻辑覆盖测试

参考结构化测试用例设计 (structural test case design)

37. maintainability —— 可维护性

一个软件系统或组件可以被修改的容易程度，这个修改一般是因为缺陷纠正、性能改进或特性增加引起的。

38. maintainability testing —— 可维护性测试

测试系统是否满足可维护性目标。

39. modified condition/decision coverage —— 修改条件/判定覆盖

在组件中被测试执行到的修改条件/判定的百分比。

40. modified condition/decision testing —— 修改条件/判定测试

根据 MC/DC 设计测试用例的一种技术。

41. Monkey Testing —— 跳跃式测试

随机性，跳跃式的测试一个系统，以确定一个系统是否会崩溃。

42. MTBF——平均失效间隔实际（mean time between failures）

两次失效之间的平均操作时间。

43. MTTF——平均失效时间（mean time to failure）

第一次失效之前的平均时间

44. MTTR——平均修复时间（mean time to repair）

两次修复之间的平均时间

45. multiple condition coverage——多条件覆盖

参考分支条件组合覆盖（branch condition combination coverage）

46. mutation analysis——变体分析

一种确定测试用例套完整性的方法，该方法通过判断测试用例套能够区别程序与其变体之间的程度。

47. Negative Testing——逆向测试/反向测试/负面测试

测试瞄准于使系统不能工作。

48. non-functional requirements testing——非功能性需求测试

与功能不相关的需求测试，如：性能测试、可用性测试等。

49. N-switch coverage——N 切换覆盖

在组件中被测试执行到的 N 转换顺序的百分比。

50. N-switch testing——N 切换测试

根据 N 转换顺序设计测试用例的一种技术，经常用于状态转换测试中。

51. N-transitions——N 转换

N+1 转换顺序

52. **operational testing**——可操作性测试

在系统或组件操作的环境中评价它们的表现。

53. **output domain**——输出域

所有可能输出的集合。

54. **partition testing**——分类测试

参考等价划分测试（**equivalence partition testing**）

55. **path**——路径

一个组件从入口到出口的一条可执行语句顺序。

56. **path coverage**——路径覆盖

在组件中被测试执行到的路径的百分比。

57. **path sensitizing**——路径敏感性

选择一组输入值强制组件走一个给定的路径。

58. **path testing**——路径测试

根据路径设计测试用例的一种技术，经常用于状态转换测试中。

59. **performance testing**——性能测试

评价一个产品或组件与性能需求是否符合的测试。

60. **portability testing**——可移植性

测试瞄准于证明软件可以被移植到指定的硬件或软件平台上。

61. **Positive Testing**——正向测试

测试瞄准于显示系统能够正常工作。

62. **precondition**——预置条件

环境或状态条件，组件执行之前必须被填充一个特定的输入值。

63. predicate —— 谓词

一个逻辑表达式，结果为‘真’或‘假’。

64. predicate data use —— 谓词数据使用

在谓词中的一个数据使用。

65. program instrumenter —— 程序插装

参考插装 (instrumenter)

66. progressive testing —— 递进测试

在先前特性回归测试之后对新特性进行测试的一种策略。

67. pseudo-random —— 伪随机

看似随机的，实际上是根据预先安排的顺序进行的。

68. QA —— 质量保证 (quality assurance)

(1) 已计划的系统性活动，用于保证一个组件、模块或系统遵从已确立的需求。(2) 采取的所有活动以保证一个开发组织交付的产品满足性能需求和已确立的标准和过程。

69. QC —— 质量控制 (quality control)

用于获得质量需求的操作技术和过程，如测试活动。

70. Race Condition —— 竞争状态

并行问题的根源。对一个共享资源的多个访问，至少包含了一个写操作，但是没有有一个机制来协调同时发生的访问。

71. recovery testing —— 恢复性测试

验证系统从失效中恢复能力的测试。

72. regression analysis and testing —— 回归分析和测试

一个软件验证和确认任务以确定在修改后需要重复测试和分析的范围。

73. **Regression Testing**——回归测试

在发生修改之后重新测试先前的测试以保证修改的正确性。

74. **release**——发布

一个批准版本的正式通知和分发。

75. **reliability**——可靠性

一个系统或组件在规定的条件下在指定的时间内执行其需要功能的能力。

76. **reliability assessment**——可靠性评价

确定一个已有系统或组件的可靠性级别的过程。

77. **requirements-based testing**——基于需求的测试

根据软件组件的需求导出测试用例的一种设计方法。

78. **review**——评审

在产品开发过程中，把产品提交给项目成员、用户、管理者或其它相关人员评价或批准的过程。

79. **risk**——风险

不期望效果的可能性和严重性的一个度量。

80. **risk assessment**——风险评估

对风险和风险影响的一个完整的评价。

81. **safety**——（生命）安全性

不会引起人员伤亡、产生疾病、毁坏或损失设备和财产、或者破

坏环境。

82. **safety critical**——严格的安全性

一个条件、事件、操作、过程或项，它的认识、控制或执行对生命安全性的系统来说是非常关键的。

83. **Sanity Testing**——理智测试

软件主要功能成分的简单测试以保证它是否能进行基本的测试。

参考冒烟测试

84. **SDP**——软件开发计划（**software development plan**）

用于一个软件产品开发的项目计划。

85. **security testing**——安全性测试

验证系统是否符合安全性目标的一种测试。

86. **security.**——（信息）安全性

参考计算机系统安全性（**computer system security**）

87. **serviceability testing**——可服务性测试

参考可维护性测试（**maintainability testing**）

88. **simple subpath**——简单子路径

控制流的一个子路径，其中没有不必要的部分被执行。

89. **simulation**——模拟

使用另一个系统来表示一个物理的或抽象的系统的选定行为特性。

90. **simulation**——模拟

使用一个可执行模型来表示一个对象的行为。

91. simulator——模拟器

软件验证期间的一个设备、软件程序、或系统，当它给定一个控制的输入时，表现的与一个给定的系统类似。

92. **SLA——服务级别协议**（service level agreement）

服务提供商与客户之间的一个协议，用于规定服务提供商应当提供什么服务。

93. **Smoke Testing——冒烟测试**

对软件主要功能进行快餐式测试。最早来自于硬件测试实践，以确定新的硬件在第一次使用的时候不会着火。

94. **Localization**：本地化

95. **Lag time**：延迟时间

96. **LCSAJ**：线性代码顺序和跳转（Linear Code Sequence And Jump）

97. **LCSAJ coverage**：LCSAJ 覆盖

98. **LCSAJ testing**：LCSAJ 测试

99. **Lead time**：前置时间

100. **Load testing**：负载测试

101. **Load Testing**：负载测试

102. **Localizability testing**：本地化能力测试

103. **Localization testing**：本地化测试

104. **logic analysis**：逻辑分析

105. **logic-coverage testing**：逻辑覆盖测试

106. Maintainability : 可维护性
107. maintainability testing: 可维护性测试
108. Maintenance : 维护
109. Master project schedule : 总体项目方案
110. Measurement : 度量
111. Memory leak : 内存泄漏
112. Migration testing : 迁移测试
113. Milestone : 里程碑
114. Mock up : 模型, 原型
115. modified condition/decision coverage: 修改条件/判定覆盖
116. modified condition/decision testing : 修改条件/判定测试
117. modular decomposition: 参考模块分解
118. Module testing : 模块测试
119. Monkey testing : 跳跃式测试
120. Monkey Testing: 跳跃式测试
121. mouse over: 鼠标在对象之上
122. mouse leave: 鼠标离开对象
123. MTBF: 平均失效间隔实际 (mean time between failures)
124. MTP MAIN TEST PLAN 主确认计划
125. MTTF: 平均失效时间 (mean time to failure)
126. MTTR: 平均修复时间 (mean time to repair)

127. multiple condition coverage: 多条件覆盖
128. mutation analysis: 变体分析
129. N/A(Not applicable) : 不适用的
130. Negative Testing : 逆向测试, 反向测试, 负面测试
131. negative testing:参考负面测试
132. Negative Testing: 逆向测试/反向测试/负面测试
133. off by one: 缓冲溢出错误
134. non-functional requirements testing:非功能需求测试
135. nominal load:额定负载
136. N-switch coverage: N 切换覆盖
137. N-switch testing: N 切换测试
138. N-transitions: N 转换
139. Off-the-shelf software : 套装软件
140. operational testing: 可操作性测试
141. output domain: 输出域
142. paper audit:书面审计
143. Pair Programming : 成对编程
144. partition testing: 分类测试
145. Path coverage : 路径覆盖
146. path coverage: 路径覆盖
147. path sensitizing: 路径敏感性
148. path testing: 路径测试

- 149. path: 路径
- 150. Peer review : 同行评审
- 151. Performance : 性能
- 152. Performance indicator: 性能（绩效）指标
- 153. Performance testing : 性能测试
- 154. Pilot : 试验
- 155. Pilot testing : 引导测试
- 156. Portability : 可移植性
- 157. portability testing: 可移植性测试
- 158. Positive testing : 正向测试
- 159. Postcondition : 后置条件
- 160. Precondition : 前提条件
- 161. precondition: 预置条件
- 162. predicate data use: 谓词数据使用
- 163. predicate: 谓词
- 164. Priority : 优先权
- 165. program instrumenter: 程序插装
- 166. progressive testing: 递进测试
- 167. Prototype : 原型
- 168. Pseudo code : 伪代码
- 169. pseudo-localization testing: 伪本地化测试
- 170. pseudo-random: 伪随机

171. QC: 质量控制 (quality control)
172. Quality assurance(QA): 质量保证
173. Quality Control(QC) : 质量控制
174. Race Condition: 竞争状态
175. Rational Unified Process(以下简称 RUP):瑞理统一工艺
176. Recovery testing : 恢复测试
177. recovery testing: 恢复性测试
178. Refactoring : 重构
179. regression analysis and testing: 回归分析和测试
180. Regression testing : 回归测试
181. Release note : 版本说明
182. release: 发布
183. Reliability : 可靠性
184. reliability assessment: 可靠性评价
185. reliability: 可靠性
186. Requirements management tool: 需求管理工具
187. Requirements-based testing : 基于需求的测试
188. Return of Investment (ROI) : 投资回报率
189. review: 评审
190. Risk assessment : 风险评估
191. risk: 风险
192. Robustness : 强健性

193. Root Cause Analysis(RCA): 根本原因分析
194. safety critical: 严格的安全性
195. safety: (生命) 安全性
196. Sanity testing : 健全测试
197. Sanity Testing: 理智测试
198. Schema Repository : 模式库
199. Screen shot : 抓屏、截图
200. SDP: 软件开发计划 (software development plan)
201. Security testing : 安全性测试
202. security testing: 安全性测试
203. security.: (信息) 安全性
204. serviceability testing: 可服务性测试
205. Severity : 严重性
206. Shipment : 发布
207. simple subpath: 简单子路径
208. Simulation : 模拟
209. Simulator : 模拟器
210. SLA(Service level agreement): 服务级别协议
211. SLA: 服务级别协议 (service level agreement)
212. Smoke testing : 冒烟测试
213. Software development plan(SDP): 软件开发计划
214. Software development process: 软件开发过程

- 215. software diversity: 软件多样性
- 216. software element: 软件元素
- 217. software engineering environment: 软件工程环境
- 218. software engineering: 软件工程
- 219. Software life cycle : 软件生命周期
- 220. source code: 源代码
- 221. source statement: 源语句
- 222. Specification : 规格说明书
- 223. specified input: 指定的输入
- 224. spiral model : 螺旋模型
- 225. SQAP SOFTWARE QUALITY ASSURENCE PLAN 软件质量保证计划
- 226. SQL: 结构化查询语句 (structured query language)
- 227. Staged Delivery: 分布交付方法
- 228. state diagram: 状态图
- 229. state transition testing : 状态转换测试
- 230. state transition: 状态转换
- 231. state: 状态
- 232. Statement coverage : 语句覆盖
- 233. statement testing: 语句测试
- 234. statement: 语句
- 235. Static Analysis: 静态分析

- 236. Static Analyzer: 静态分析器
- 237. Static Testing: 静态测试
- 238. statistical testing: 统计测试
- 239. Stepwise refinement : 逐步优化
- 240. storage testing: 存储测试
- 241. **Stress Testing** : 压力测试
- 242. structural coverage: 结构化覆盖
- 243. structural test case design: 结构化测试用例设计
- 244. structural testing: 结构化测试
- 245. structured basis testing: 结构化的基础测试
- 246. structured design: 结构化设计
- 247. structured programming: 结构化编程
- 248. structured walkthrough: 结构化走读
- 249. **stub**: 桩
- 250. sub-area: 子域
- 251. Summary: 总结
- 252. SVVP SOFTWARE Vevification&Validation PLAN: 软件验证和确认计划
- 253. symbolic uation: 符号评价
- 254. symbolic execution:参考符号执行
- 255. symbolic execution: 符号执行
- 256. symbolic trace: 符号轨迹

- 257. Synchronization : 同步
- 258. Syntax testing : 语法分析
- 259. system analysis: 系统分析
- 260. System design : 系统设计
- 261. system integration: 系统集成
- 262. System Testing : 系统测试
- 263. TC TEST CASE 测试用例
- 264. TCS TEST CASE SPECIFICATION 测试用例规格说明
- 265. TDS TEST DESIGN SPECIFICATION 测试设计规格说明书
- 266. technical requirements testing: 技术需求测试
- 267. Test : 测试
- 268. test automation: 测试自动化
- 269. Test case : 测试用例
- 270. test case design technique: 测试用例设计技术
- 271. test case suite: 测试用例套
- 272. test comparator: 测试比较器
- 273. test completion criterion: 测试完成标准
- 274. test coverage:测试覆盖
- 275. Test design : 测试设计
- 276. Test driver : 测试驱动
- 277. test environment:测试环境

- 278. test execution technique:测试执行技术
- 279. test execution:测试执行
- 280. test generator:测试生成器
- 281. test harness:测试用具
- 282. Test infrastructure : 测试基础建设
- 283. test log:测试日志
- 284. test measurement technique:测试度量技术
- 285. Test Metrics :测试度量
- 286. test procedure:测试规程
- 287. test records:测试记录
- 288. test report:测试报告
- 289. Test scenario : 测试场景
- 290. Test Script.:测试脚本
- 291. Test Specification:测试规格
- 292. Test strategy : 测试策略
- 293. test suite:测试套
- 294. Test target : 测试目标
- 295. Test ware : 测试工具
- 296. Testability : 可测试性
- 297. testability:可测试性
- 298. Testing bed : 测试平台
- 299. Testing coverage : 测试覆盖

- 300. Testing environment : 测试环境
- 301. Testing item : 测试项
- 302. Testing plan : 测试计划
- 303. Testing procedure : 测试过程
- 304. Thread testing : 线程测试
- 305. time sharing:时间共享
- 306. time-boxed : 固定时间
- 307. TIR test incident report 测试事故报告
- 308. ToolTip:控件提示或说明
- 309. top-down testing:自顶向下测试
- 310. TPS TEST PEOCESS SPECIFICATION 测试步骤规格说明
- 311. Traceability : 可跟踪性
- 312. traceability analysis:跟踪性分析
- 313. traceability matrix:跟踪矩阵
- 314. Trade-off : 平衡
- 315. transaction:事务/处理
- 316. transaction volume: 交易量
- 317. transform. analysis:事务分析
- 318. trojan horse:特洛伊木马
- 319. truth table:真值表
- 320. TST TEST SUMMARY REPORT 测试总结报告
- 321. Tune System : 调试系统

- 322. TW TEST WARE :测试件
- 323. Unit Testing :单元测试
- 324. Usability Testing:可用性测试
- 325. Usage scenario : 使用场景
- 326. User acceptance Test : 用户验收测试
- 327. User database :用户数据库
- 328. User interface(UI) : 用户界面
- 329. User profile : 用户信息
- 330. User scenario : 用户场景
- 331. V&V (Verification & Validation) : 验证&确认
- 332. validation :确认
- 333. verification :验证
- 334. version :版本
- 335. Virtual user : 虚拟用户
- 336. volume testing:容量测试
- 337. VSS (visual source safe) :
- 338. VTP Verification TEST PLAN 验证测试计划
- 339. VTR Verification TEST REPORT 验证测试报告
- 340. Walkthrough : 走读
- 341. Waterfall model : 瀑布模型
- 342. Web testing : 网站测试
- 343. White box testing : 白盒测试

344. Work breakdown structure (WBS)：任务分解结构

345. Zero bug bounce (ZBB)：零错误反弹

346. 测试名词

347. **fault**——故障 **bug**

在软件中一个错误的表现。

348. **feasible path**——可达路径

可以通过一组输入值和条件执行到的一条路径。

349. **feature testing**——特性测试

参考功能测试（**Functional Testing**）

350. **FMEA**——失效模型效果分析（**Failure Modes and Effects Analysis**）

351. 可靠性分析中的一种方法，用于在基本组件级别上确认对系统性能有重大影响的失效

352. **FMECA**——失效模型效果关键性分析(**Failure Modes and Effects Criticality Analysis**)

353. **FMEA** 的一个扩展，它分析了失效结果的严重性。

354. **FTA**——故障树分析(**Fault Tree Analysis**)

引起一个不需要事件产生的条件和因素的确证和分析，通常是严重影响系统性能、经济性、安全性或其它需要特性。

355. **functional decomposition**——功能分解

参考模块分解（**modular decomposition**）

356. Functional Specification ——功能规格说明书

一个详细描述产品特性的文档。

357. Functional Testing ——功能测试

测试一个产品的特性和可操作行为以确定它们满足规格。

358. glass box testing ——玻璃盒测试 参考白盒测试 (White Box Testing)

359. IEEE ——美国电子与电器工程师学会 (Institute of Electrical and Electronic Engineers)

360. incremental testing ——渐增测试

361. 集成测试的一种, 组件逐渐被增加到系统中直到整个系统被集成。

362. infeasible path ——不可达路径 不能够通过任何可能的输入值集合执行到的路径。

363. input domain ——输入域 所有可能输入的集合。

364. inspection ——检视 对文档进行的一种评审形式。

365. installability testing ——可安装性测试 确定系统的安装程序是否正确的测试。

366. instrumentation ——插装 在程序中插入额外的代码以获得程序在执行时行为的信息。

367. instrumenter ——插装器 执行插装的工具

368. Integration Testing ——集成测试 测试一个应用组合后的部分以确保它们的功能在组合之后正确。该测试一般在单元测试之后

进行。

369. **interface**——**接口** 两个功能单元的共享边界。
370. **interface analysis**——**接口分析** 分析软件与硬件、用户和其它软件之间接口的需求规格。
371. **interface testing**——**接口测试** 测试系统组件间接口的一种测试。
372. **invalid inputs**——**无效输入** 在程序功能输入域之外的测试数据。
373. **isolation testing**——**孤立测试** 组件测试（单元测试）策略中的一种，把被测组件从其上下文组件之中孤立出来，通过设计驱动和桩进行测试的一种方法。
374. **job control language**——**工作控制语言** 用于确定工作顺序，描述它们对操作系统要求并控制它们执行的语言。
375. **LCSAJ**——**线性代码顺序和跳转（Linear Code Sequence And Jump）** 包含三个部分：可执行语句线性顺序的起始，线性顺序的结束，在线性顺序结束处控制流跳转的目标语句。
376. **LCSAJ coverage**——**LCSAJ 覆盖** 在组件中被测试执行到的 LCSAJ 的百分比。
377. **LCSAJ testing**——**LCSAJ 测试** 根据 LCSAJ 设计测试用例的一种技术。
378. **Load Testing**——**负载测试** 通过测试系统在资源超负荷情况下的表现，以发现设计上的错误或验证系统的负载能力。

379. logic analysis——逻辑分析

- (1) 评价软件设计的关键安全方程式、算法和控制逻辑的方法。(2) 评价程序操作的顺序并且检测可能导致灾难的错误。

380. logic-coverage testing——逻辑覆盖测试 参考结构化测试用例设计 (structural test case design)

381. maintainability——可维护性 一个软件系统或组件可以被修改的容易程度，这个修改一般是因为缺陷纠正、性能改进或特性增加引起的。

382. maintainability testing——可维护性测试 测试系统是否满足可维护性目标。

383. modified condition/decision coverage——修改条件/判定覆盖 在组件中被测试执行到的修改条件/判定的百分比。

384. modified condition/decision testing——修改条件/判定测试 根据 MC/DC 设计测试用例的一种技术。

385. Monkey Testing——跳跃式测试 随机性，跳跃式的测试一个系统，以确定一个系统是否会崩溃。

386. MTBF——平均失效间隔实际 (mean time between failures) 两次失效之间的平均操作时间。

387. MTTF——平均失效时间 (mean time to failure)

388. 第一次失效之前的平均时间

389. MTTR——平均修复时间 (mean time to repair) 两次修复之间

的平均时间

390. multiple condition coverage——多条件覆盖 参考分支条件组合覆盖（branch condition combination coverage）
391. mutation analysis——变体分析 一种确定测试用例套完整性的方法，该方法通过判断测试用例套能够区别程序与其变体之间的程度。
392. Negative Testing——逆向测试/反向测试/负面测试 测试瞄准于使系统不能工作。
393. non-functional requirements testing——非功能性需求测试 与功能不相关的需求测试，如：性能测试、可用性测试等。
394. N-switch coverage——N 切换覆盖 在组件中被测试执行到的 N 转换顺序的百分比。
395. N-switch testing——N 切换测试 根据 N 转换顺序设计测试用例的一种技术，经常用于状态转换测试中。
396. N-transitions——N 转换 N+1 转换顺序
397. operational testing——可操作性测试 在系统或组件操作的环境中评价它们的表现。
398. output domain——输出域 所有可能输出的集合。
399. partition testing——分类测试 参考等价划分测试（equivalence partition testing）
400. path——路径 一个组件从入口到出口的一条可执行语句顺序。

- 401. **path coverage**——路径覆盖
- 402. 在组件中被测试执行到的路径的百分比。
- 403. **path sensitizing**——路径敏感性
- 404. 选择一组输入值强制组件走一个给定的路径。
- 405. **path testing**——路径测试
- 406. 根据路径设计测试用例的一种技术，经常用于状态转换测试中。
- 407. **performance testing**——性能测试
- 408. 评价一个产品或组件与性能需求是否符合的测试。
- 409. **portability testing**——可移植性
- 410. 测试瞄准于证明软件可以被移植到指定的硬件或软件平台上。
- 411. **Positive Testing**——正向测试
- 412. 测试瞄准于显示系统能够正常工作。
- 413. **precondition**——预置条件
- 414. 环境或状态条件，组件执行之前必须被填充一个特定的输入值。

415. predicate——谓词
416. 一个逻辑表达式，结果为‘真’或‘假’。
417. predicate data use——谓词数据使用
418. 在谓词中的一个数据使用。
419. program instrumenter——程序插装
420. 参考插装（instrumenter）
421. progressive testing——递进测试
422. 在先前特性回归测试之后对新特性进行测试的一种策略。
423. pseudo-random——伪随机
424. 看似随机的，实际上是根据预先安排的顺序进行的。
- 425. QA——质量保证（quality assurance）**
- （1） 已计划的系统性活动，用于保证一个组件、模块或系统遵从已确立的需求。（2）采取的所有活动以保证一个开发组织交付的产品满足性能需求和已确立的标准和过程。

426. **QC——质量控制 (quality control)**
427. 用于获得质量需求的操作技术和过程，如测试活动。
428. **Race Condition——竞争状态**
429. 并行问题的根源。对一个共享资源的多个访问，至少包含了一个写操作，但是没有有一个机制来协调同时发生的访问。
430. **recovery testing——恢复性测试**
431. 验证系统从失效中恢复能力的测试。
432. **regression analysis and testing——回归分析和测试**
433. 一个软件验证和确认任务以确定在修改后需要重复测试和分析的范围。
434. **Regression Testing——回归测试**
435. 在发生修改之后重新测试先前的测试以保证修改的正确性。
436. **release——发布**
437. 一个批准版本的正式通知和分发。
438. **reliability——可靠性**
439. 一个系统或组件在规定的条件下在指定的时间内执行其需要

功能的能力。

440. **reliability assessment**——可靠性评价

441. 确定一个已有系统或组件的可靠性级别的过程。

442. **requirements-based testing**——基于需求的测试

443. 根据软件组件的需求导出测试用例的一种设计方法。

444. **review**——评审

445. 在产品开发过程中，把产品提交给项目成员、用户、管理者或其它相关人员评价或批准的过程。

446. **risk**——风险

447. 不期望效果的可能性和严重性的一个度量。

448. **risk assessment**——风险评估

449. 对风险和风险影响的一个完整的评价。

450. **safety**——（生命）安全性

451. 不会引起人员伤亡、产生疾病、毁坏或损失设备和财产、或者破坏环境。

452. **safety critical**——严格的安全性
453. 一个条件、事件、操作、过程或项，它的认识、控制或执行对生命安全性的系统来说是非常关键的。
454. **Sanity Testing**——理智测试
455. 软件主要功能成分的简单测试以保证它是否能进行基本的测试。参考冒烟测试
456. **SDP**——软件开发计划（software development plan）
457. 用于一个软件产品开发的项目计划。
458. **security testing**——安全性测试
459. 验证系统是否符合安全性目标的一种测试。
460. **security.**——（信息）安全性
461. 参考计算机系统安全性（computer system security）
462. **serviceability testing**——可服务性测试
463. 参考可维护性测试（maintainability testing）
464. **simple subpath**——简单子路径
465. 控制流的一个子路径，其中没有不必要的部分被执行。

466. simulation——模拟
467. 使用另一个系统来表示一个物理的或抽象的系统的选定行为特性。
468. simulation——模拟
469. 使用一个可执行模型来表示一个对象的行为。
470. simulator——模拟器
471. 软件验证期间的一个设备、软件程序、或系统，当它给定一个控制的输入时，表现的与一个给定的系统类似。
472. SLA——服务级别协议（service level agreement）
473. 服务提供商与客户之间的一个协议，用于规定服务提供商应当提供什么服务。
474. **Smoke Testing——冒烟测试**
475. 对软件主要功能进行快餐式测试。最早来自于硬件测试实践，以确定新的硬件在第一次使用的时候不会着火。
476. 三、
477. 软件测试中英词汇对照
478. **Localization : 本地化**

- 479. **Lag time** : 延迟时间
- 480. LCSAJ: 线性代码顺序和跳转 (Linear Code Sequence And Jump)
- 481. LCSAJ coverage: LCSAJ 覆盖
- 482. LCSAJ testing: LCSAJ 测试
- 483. Lead time : 前置时间
- 484. **Load testing** : 负载测试
- 485. **Load Testing**: 负载测试
- 486. **Localizability testing**: 本地化能力测试
- 487. **Localization testing** : 本地化测试
- 488. logic analysis: 逻辑分析
- 489. logic-coverage testing: 逻辑覆盖测试
- 490. Maintainability : 可维护性
- 491. maintainability testing: 可维护性测试
- 492. Maintenance : 维护
- 493. **Master project schedule** : 总体项目方案
- 494. **Measurement** : 度量
- 495. **Memory leak** : 内存泄漏
- 496. Migration testing : 迁移测试
- 497. **Milestone** : 里程碑
- 498. Mock up : 模型, 原型
- 499. modified condition/decision coverage: 修改条件/判定覆盖

- 500. modified condition/decision testing : 修改条件/判定测试
- 501. modular decomposition:参考模块分解
- 502. Module testing : 模块测试
- 503. Monkey testing : 跳跃式测试
- 504. Monkey Testing: 跳跃式测试
- 505. mouse over: 鼠标在对象之上
- 506. mouse leave: 鼠标离开对象
- 507. MTBF: 平均失效间隔实际 (mean time between failures)
- 508. MTP MAIN TEST PLAN 主确认计划
- 509. MTTF: 平均失效时间 (mean time to failure)
- 510. MTTR: 平均修复时间 (mean time to repair)
- 511. multiple condition coverage: 多条件覆盖
- 512. mutation analysis: 变体分析
- 513. N/A(Not applicable) : 不适用的
- 514. Negative Testing : 逆向测试, 反向测试, 负面测试
- 515. negative testing:参考负面测试
- 516. Negative Testing: 逆向测试/反向测试/负面测试
- 517. off by one: 缓冲溢出错误
- 518. non-functional requirements testing:非功能需求测试
- 519. nominal load:额定负载
- 520. N-switch coverage: N 切换覆盖

- 521. N-switch testing: N 切换测试
- 522. N-transitions: N 转换
- 523. Off-the-shelf software : 套装软件
- 524. operational testing: 可操作性测试
- 525. output domain: 输出域
- 526. paper audit: 书面审计
- 527. Pair Programming : 成对编程
- 528. partition testing: 分类测试
- 529. Path coverage : 路径覆盖
- 530. path coverage: 路径覆盖
- 531. path sensitizing: 路径敏感性
- 532. path testing: 路径测试
- 533. path: 路径
- 534. Peer review : 同行评审
- 535. Performance : 性能
- 536. Performance indicator: 性能（绩效）指标
- 537. Performance testing : 性能测试
- 538. Pilot : 试验
- 539. Pilot testing : 引导测试
- 540. Portability : 可移植性
- 541. portability testing: 可移植性测试
- 542. Positive testing : 正向测试

- 543. **Postcondition** : 后置条件
- 544. **Precondition** : 前提条件
- 545. **precondition**: 预置条件
- 546. **predicate data use**: 谓词数据使用
- 547. **predicate**: 谓词
- 548. **Priority** : 优先权
- 549. **program instrumenter**: 程序插装
- 550. **progressive testing**: 递进测试
- 551. **Prototype** : 原型
- 552. **Pseudo code** : 伪代码
- 553. **pseudo-localization testing**: 伪本地化测试
- 554. **pseudo-random**: 伪随机
- 555. **QC: 质量控制 (quality control)**
- 556. **Quality assurance(QA)**: 质量保证
- 557. **Quality Control(QC)** : 质量控制
- 558. **Race Condition**: 竞争状态
- 559. **Rational Unified Process(以下简称 RUP)**:瑞理统一工艺
- 560. **Recovery testing** : 恢复测试
- 561. **recovery testing**: 恢复性测试
- 562. **Refactoring** : 重构
- 563. **regression analysis and testing**: 回归分析和测试
- 564. **Regression testing** : 回归测试

- 565. Release : 发布
- 566. Release note : 版本说明
- 567. release: 发布
- 568. Reliability : 可靠性
- 569. reliability assessment: 可靠性评价
- 570. reliability: 可靠性
- 571. Requirements management tool: 需求管理工具
- 572. Requirements-based testing : 基于需求的测试
- 573. Return of Investment (ROI) : 投资回报率
- 574. review: 评审
- 575. Risk assessment : 风险评估
- 576. risk: 风险
- 577. Robustness : 强健性
- 578. Root Cause Analysis(RCA): 根本原因分析
- 579. safety critical: 严格的安全性
- 580. safety: (生命) 安全性
- 581. Sanity testing : 健全测试
- 582. Sanity Testing: 理智测试
- 583. Schema Repository : 模式库
- 584. Screen shot : 抓屏、截图
- 585. SDP: 软件开发计划 (software development plan)
- 586. Security testing : 安全性测试

- 587. security testing: 安全性测试
- 588. security.: (信息) 安全性
- 589. serviceability testing: 可服务性测试
- 590. Severity : 严重性
- 591. Shipment : 发布
- 592. simple subpath: 简单子路径
- 593. Simulation : 模拟
- 594. Simulator : 模拟器
- 595. SLA(Service level agreement): 服务级别协议
- 596. SLA: 服务级别协议 (service level agreement)
- 597. Smoke testing : 冒烟测试
- 598. Software development plan(SDP): 软件开发计划
- 599. Software development process: 软件开发过程
- 600. software development process: 软件开发过程
- 601. software diversity: 软件多样性
- 602. software element: 软件元素
- 603. software engineering environment: 软件工程环境
- 604. software engineering: 软件工程
- 605. Software life cycle : 软件生命周期
- 606. source code: 源代码
- 607. source statement: 源语句
- 608. Specification : 规格说明书

- 609. specified input: 指定的输入
- 610. spiral model : 螺旋模型
- 611. SQAP SOFTWARE QUALITY ASSURENCE PLAN 软件质量保证计划
- 612. SQL: 结构化查询语句 (structured query language)
- 613. Staged Delivery: 分布交付方法
- 614. state diagram: 状态图
- 615. state transition testing : 状态转换测试
- 616. state transition: 状态转换
- 617. state: 状态
- 618. Statement coverage : 语句覆盖
- 619. statement testing: 语句测试
- 620. statement: 语句
- 621. Static Analysis: 静态分析
- 622. Static Analyzer: 静态分析器
- 623. Static Testing: 静态测试
- 624. statistical testing: 统计测试
- 625. Stepwise refinement : 逐步优化
- 626. storage testing: 存储测试
- 627. Stress Testing : 压力测试
- 628. structural coverage: 结构化覆盖
- 629. structural test case design: 结构化测试用例设计

- 630. structural testing: 结构化测试
- 631. structured basis testing: 结构化的基础测试
- 632. structured design: 结构化设计
- 633. structured programming: 结构化编程
- 634. structured walkthrough: 结构化走读
- 635. stub: 桩
- 636. sub-area: 子域
- 637. Summary: 总结
- 638. SVVP SOFTWARE Vevification&Validation PLAN: 软件验证和确认计划
- 639. symbolic evaluation: 符号评价
- 640. symbolic execution: 参考符号执行
- 641. symbolic execution: 符号执行
- 642. symbolic trace: 符号轨迹
- 643. Synchronization : 同步
- 644. Syntax testing : 语法分析
- 645. system analysis: 系统分析
- 646. System design : 系统设计
- 647. system integration: 系统集成
- 648. System Testing : 系统测试
- 649. TC TEST CASE 测试用例
- 650. TCS TEST CASE SPECIFICATION 测试用例规格说明

651. TDS TEST DESIGN SPECIFICATION 测试设计规格说明书
652. technical requirements testing: 技术需求测试
653. Test : 测试
654. test automation: 测试自动化
655. Test case : 测试用例
656. test case design technique: 测试用例设计技术
657. test case suite: 测试用例套
658. test comparator: 测试比较器
659. test completion criterion: 测试完成标准
660. test coverage:测试覆盖
661. Test design : 测试设计
662. Test driver : 测试驱动
663. test environment:测试环境
664. test execution technique:测试执行技术
665. test execution:测试执行
666. test generator:测试生成器
667. test harness:测试用具
668. Test infrastructure : 测试基础建设
669. test log:测试日志
670. test measurement technique:测试度量技术
671. Test Metrics :测试度量

- 672. test procedure:测试规程
- 673. test records:测试记录
- 674. test report:测试报告
- 675. Test scenario : 测试场景
- 676. Test Script.:测试脚本
- 677. Test Specification:测试规格
- 678. Test strategy : 测试策略
- 679. test suite:测试套
- 680. Test target : 测试目标
- 681. Test ware : 测试工具
- 682. Testability : 可测试性
- 683. testability:可测试性
- 684. Testing bed : 测试平台
- 685. Testing coverage : 测试覆盖
- 686. Testing environment : 测试环境
- 687. Testing item : 测试项
- 688. Testing plan : 测试计划
- 689. Testing procedure : 测试过程
- 690. Thread testing : 线程测试
- 691. time sharing:时间共享
- 692. time-boxed : 固定时间
- 693. TIR test incident report 测试事故报告

- 694. ToolTip:控件提示或说明
- 695. top-down testing:自顶向下测试
- 696. TPS TEST PROCESS SPECIFICATION 测试步骤规格说明
- 697. Traceability : 可跟踪性
- 698. traceability analysis:跟踪性分析
- 699. traceability matrix:跟踪矩阵
- 700. Trade-off : 平衡
- 701. transaction:事务/处理
- 702. transaction volume: 交易量
- 703. transform. analysis:事务分析
- 704. trojan horse:特洛伊木马
- 705. truth table:真值表
- 706. TST TEST SUMMARY REPORT 测试总结报告
- 707. Tune System : 调试系统
- 708. TW TEST WARE :测试件
- 709. Unit Testing :单元测试
- 710. Usability Testing:可用性测试
- 711. Usage scenario : 使用场景
- 712. User acceptance Test : 用户验收测试
- 713. User database :用户数据库
- 714. User interface(UI) : 用户界面
- 715. User profile : 用户信息

- 716. User scenario : 用户场景
- 717. V&V (Verification & Validation) : 验证&确认
- 718. validation :确认
- 719. verification :验证
- 720. version :版本
- 721. Virtual user : 虚拟用户
- 722. volume testing:容量测试
- 723. VSS (visual source safe) :
- 724. VTP Verification TEST PLAN 验证测试计划
- 725. VTR Verification TEST REPORT 验证测试报告
- 726. Walkthrough : 走读
- 727. Waterfall model : 瀑布模型
- 728. Web testing : 网站测试
- 729. White box testing : 白盒测试
- 730. Work breakdown structure (WBS) : 任务分解结构
- 731. Zero bug bounce (ZBB) : 零错误反弹
- 732. 四、
- 733. Acceptance testing : 验收测试
- 734. Acceptance Testing: 可接受性测试
- 735. Accessibility test : 软体适用性测试

- 736. **actual outcome:** 实际结果
- 737. Ad hoc testing : 随机测试
- 738. Algorithm analysis : 算法分析
- 739. **algorithm:** 算法
- 740. **Alpha testing** : α 测试
- 741. **analysis:** 分析
- 742. **anomaly:** 异常
- 743. **application software:** 应用软件
- 744. Application under test (AUT) : 所测试的应用程序
- 745. Architecture : 构架
- 746. Artifact : 工件
- 747. ASQ: 自动化软件质量 (Automated Software Quality)
- 748. Assertion checking : 断言检查
- 749. Association : 关联
- 750. Audit : 审计
- 751. **audit trail:** 审计跟踪
- 752. **Automated Testing:** 自动化测试
- 753. **Backus-Naur Form:** BNF 范式
- 754. **baseline:** 基线
- 755. **Basic Block:** 基本块
- 756. basis test set: 基本测试集
- 757. Behaviour : 行为

- 758. Bench test : 基准测试
- 759. benchmark: 标杆/指标/基准
- 760. Best practise : 最佳实践
- 761. Beta testing : β 测试
- 762. Black Box Testing: 黑盒测试
- 763. Blocking bug : 阻碍性错误
- 764. Bottom-up testing : 自底向上测试
- 765. boundary value coverage: 边界值覆盖
- 766. boundary value testing: 边界值测试
- 767. Boundary values : 边界值
- 768. Boundry Value Analysis: 边界值分析
- 769. branch condition combination coverage: 分支条件组合覆盖
- 770. branch condition combination testing: 分支条件组合测试
- 771. branch condition coverage: 分支条件覆盖
- 772. branch condition testing: 分支条件测试
- 773. branch condition: 分支条件
- 774. Branch coverage : 分支覆盖
- 775. branch outcome: 分支结果
- 776. branch point: 分支点
- 777. branch testing: 分支测试
- 778. branch: 分支
- 779. Breadth Testing: 广度测试

- 780. Brute force testing: 强力测试
- 781. Buddy test : 合伙测试
- 782. Buffer : 缓冲
- 783. Bug : 错误
- 784. Bug bash : 错误大扫除
- 785. bug fix : 错误修正
- 786. Bug report : 错误报告
- 787. Bug tracking system: 错误跟踪系统
- 788. bug: 缺陷
- 789. Build : 工作版本（内部小版本）
- 790. Build Verification tests(BVTs): 版本验证测试
- 791. Build-in : 内置
- 792. Capability Maturity Model (CMM): 能力成熟度模型
- 793. Capability Maturity Model Integration (CMMI): 能力成熟度模型整合
- 794. capture/playback tool: 捕获/回放工具
- 795. Capture/Replay Tool: 捕获/回放工具
- 796. CASE: 计算机辅助软件工程（computer aided software engineering）
- 797. CAST: 计算机辅助测试
- 798. cause-effect graph: 因果图
- 799. certification : 证明

- 800. change control: 变更控制
- 801. Change Management :变更管理
- 802. Change Request :变更请求
- 803. Character Set : 字符集
- 804. Check In :检入
- 805. Check Out :检出
- 806. Closeout : 收尾
- 807. code audit : 代码审计
- 808. Code coverage : 代码覆盖
- 809. Code Inspection: 代码检视
- 810. Code page : 代码页
- 811. Code rule : 编码规范
- 812. Code sytle : 编码风格
- 813. Code Walkthrough: 代码走读
- 814. code-based testing: 基于代码的测试
- 815. coding standards: 编程规范
- 816. Common sense : 常识
- 817. Compatibility Testing: 兼容性测试
- 818. complete path testing : 完全路径测试
- 819. completeness: 完整性
- 820. complexity : 复杂性
- 821. Component testing : 组件测试

- 822. **Component:** 组件
- 823. computation data use: 计算数据使用
- 824. computer system security: 计算机系统安全性
- 825. **Concurrency user** : 并发用户
- 826. **condition coverage:** 条件覆盖
- 827. **condition outcome:** 条件结果
- 828. **condition:** 条件
- 829. **configuration control:** 配置控制
- 830. **Configuration item** : 配置项
- 831. **configuration management:** 配置管理
- 832. **Configuration testing :** 配置测试
- 833. conformance criterion: 一致性标准
- 834. Conformance Testing: 一致性测试
- 835. **consistency** : 一致性
- 836. consistency checker: 一致性检查器
- 837. **Control flow graph** : 控制流程图
- 838. **control flow graph:** 控制流图
- 839. **control flow:** 控制流
- 840. conversion testing: 转换测试
- 841. Core team : 核心小组
- 842. corrective maintenance: 故障检修
- 843. correctness : 正确性

- 844. coverage : 覆盖率
- 845. coverage item: 覆盖项
- 846. crash: 崩溃
- 847. criticality analysis: 关键性分析
- 848. criticality: 关键性
- 849. CRM(change request management): 变更需求管理
- 850. Customer-focused mindset : 客户为中心的理念体系
- 851. Cyclomatic complexity : 圈复杂度
- 852. data corruption: 数据污染
- 853. data definition C-use pair: 数据定义 C-use 使用对
- 854. data definition P-use coverage: 数据定义 P-use 覆盖
- 855. data definition P-use pair: 数据定义 P-use 使用对
- 856. data definition: 数据定义
- 857. data definition-use coverage: 数据定义使用覆盖
- 858. data definition-use pair : 数据定义使用对
- 859. data definition-use testing: 数据定义使用测试
- 860. data dictionary: 数据字典
- 861. Data Flow Analysis : 数据流分析
- 862. data flow analysis: 数据流分析
- 863. data flow coverage: 数据流覆盖
- 864. data flow diagram: 数据流图
- 865. data flow testing: 数据流测试

- 866. data integrity: 数据完整性
- 867. data use: 数据使用
- 868. data validation: 数据确认
- 869. dead code: 死代码
- 870. Debug : 调试
- 871. Debugging: 调试
- 872. Decision condition: 判定条件
- 873. Decision coverage : 判定覆盖
- 874. decision coverage: 判定覆盖
- 875. decision outcome: 判定结果
- 876. decision table: 判定表
- 877. decision: 判定
- 878. Defect : 缺陷
- 879. defect density : 缺陷密度
- 880. Defect Tracking :缺陷跟踪
- 881. Deployment : 部署
- 882. Depth Testing: 深度测试
- 883. design for sustainability :可延续性的设计
- 884. design of experiments: 实验设计
- 885. design-based testing: 基于设计的测试
- 886. Desk checking : 桌前检查
- 887. desk checking: 桌面检查

- 888. Determine Usage Model : 确定应用模型
- 889. Determine Potential Risks : 确定潜在风险
- 890. diagnostic: 诊断
- 891. DIF(decimation in frequency) : 按频率抽取
- 892. dirty testing: 肮脏测试
- 893. disaster recovery: 灾难恢复
- 894. DIT (decimation in time) : 按时间抽取
- 895. documentation testing : 文档测试
- 896. domain testing: 域测试
- 897. domain: 域
- 898. DTP DETAIL TEST PLAN 详细确认测试计划
- 899. Dynamic analysis : 动态分析
- 900. dynamic analysis: 动态分析
- 901. Dynamic Testing: 动态测试
- 902. embedded software: 嵌入式软件
- 903. emulator: 仿真
- 904. End-to-End testing: 端到端测试
- 905. Enhanced Request :增强请求
- 906. **entity relationship diagram: 实体关系图**
- 907. Encryption Source Code Base: 加密算法源代码库
- 908. **Entry criteria : 准入条件**
- 909. **entry point : 入口点**

- 910. Envisioning Phase : 构想阶段
- 911. Equivalence class : 等价类
- 912. Equivalence Class: 等价类
- 913. equivalence partition coverage: 等价划分覆盖
- 914. Equivalence partition testing : 等价划分测试
- 915. equivalence partition testing:参考等价划分测试
- 916. equivalence partition testing: 等价划分测试
- 917. Equivalence Partitioning: 等价划分
- 918. Error : 错误
- 919. Error guessing : 错误猜测
- 920. error seeding: 错误播种/错误插值
- 921. error: 错误
- 922. Event-driven : 事件驱动
- 923. Exception handlers : 异常处理器
- 924. exception: 异常/例外
- 925. executable statement: 可执行语句
- 926. Exhaustive Testing: 穷尽测试
- 927. exit point: 出口点
- 928. expected outcome: 期望结果
- 929. Exploratory testing : 探索性测试
- 930. Failure : 失效
- 931. Fault : 故障

- 932. **fault:** 故障
- 933. **feasible path:** 可达路径
- 934. **feature testing:** 特性测试
- 935. **Field testing** : 现场测试
- 936. **FMEA:** 失效模型效果分析(Failure Modes and Effects Analysis)
- 937. **FMECA:** 失效模型效果关键性分析(Failure Modes and Effects Criticality Analysis)
- 938. **Framework** : 框架
- 939. **FTA:** 故障树分析(Fault Tree Analysis)
- 940. **functional decomposition:** 功能分解
- 941. **Functional Specification** : 功能规格说明书
- 942. **Functional testing** : 功能测试
- 943. **Functional Testing:** 功能测试
- 944. **G11N(Globalization)** : 全球化
- 945. **Gap analysis** : 差距分析
- 946. **Garbage characters** : 乱码字符
- 947. **glass box testing:** 玻璃盒测试
- 948. **Glass-box testing** : 白箱测试或白盒测试
- 949. **Glossary** : 术语表
- 950. **GUI(Graphical User Interface):** 图形用户界面
- 951. **Hard-coding** : 硬编码
- 952. **Hotfix** : 热补丁

- 953. I18N(Internationalization): 国际化
- 954. Identify Exploratory Tests - 识别探索性测试
- 955. IEEE: 美国电子与电器工程师学会 (Institute of Electrical and Electronic Engineers)
- 956. Incident 事故
- 957. Incremental testing : 渐增测试
- 958. incremental testing: 渐增测试
- 959. infeasible path: 不可达路径
- 960. input domain: 输入域
- 961. Inspection : 审查
- 962. inspection: 检视
- 963. installability testing: 可安装性测试
- 964. Installing testing : 安装测试
- 965. instrumentation: 插装
- 966. instrumenter: 插装器
- 967. Integration :集成
- 968. Integration testing : 集成测试
- 969. interface : 接口
- 970. interface analysis: 接口分析
- 971. interface testing: 接口测试
- 972. interface: 接口
- 973. invalid inputs: 无效输入

- 974. isolation testing: 孤立测试
- 975. Issue : 问题
- 976. Iteration : 迭代
- 977. Iterative development: 迭代开发
- 978. job control language: 工作控制语言
- 979. Job: 工作
- 980. Key concepts : 关键概念
- 981. Key Process Area : 关键过程区域
- 982. Keyword driven testing : 关键字驱动测试
- 983. Kick-off meeting : 动会议
- 984. L10N(Localization) : 本地化
- 985. Lag time : 延迟时间
- 986. LCSAJ: 线性代码顺序和跳转 (Linear Code Sequence And Jump)
- 987. LCSAJ coverage: LCSAJ 覆盖
- 988. LCSAJ testing: LCSAJ 测试
- 989. Lead time : 前置时间
- 990. Load testing : 负载测试
- 991. Load Testing: 负载测试
- 992. Localizability testing: 本地化能力测试
- 993. Localization testing : 本地化测试
- 994. logic analysis: 逻辑分析

- 995. logic-coverage testing: 逻辑覆盖测试
- 996. Maintainability : 可维护性
- 997. maintainability testing: 可维护性测试
- 998. Maintenance : 维护
- 999. Master project schedule : 总体项目方案
- 1000. Measurement : 度量
- 1001. Memory leak : 内存泄漏
- 1002. Migration testing : 迁移测试
- 1003. Milestone : 里程碑
- 1004. Mock up : 模型, 原型
- 1005. modified condition/decision coverage: 修改条件/判定覆盖
- 1006. modified condition/decision testing : 修改条件/判定测试
- 1007. modular decomposition: 参考模块分解
- 1008. Module testing : 模块测试
- 1009. Monkey testing : 跳跃式测试
- 1010. Monkey Testing: 跳跃式测试
- 1011. mouse over: 鼠标在对象之上
- 1012. mouse leave: 鼠标离开对象
- 1013. MTBF: 平均失效间隔实际 (mean time between failures)
- 1014. MTP MAIN TEST PLAN 主确认计划
- 1015. MTTF: 平均失效时间 (mean time to failure)

- 1016. MTTR: 平均修复时间 (mean time to repair)
- 1017. multiple condition coverage: 多条件覆盖
- 1018. mutation analysis: 变体分析
- 1019. N/A(Not applicable) : 不适用的
- 1020. Negative Testing : 逆向测试, 反向测试, 负面测试
- 1021. negative testing:参考负面测试
- 1022. Negative Testing: 逆向测试/反向测试/负面测试
- 1023. off by one: 缓冲溢出错误
- 1024. non-functional requirements testing:非功能需求测试
- 1025. nominal load:额定负载
- 1026. N-switch coverage: N 切换覆盖
- 1027. N-switch testing: N 切换测试
- 1028. N-transitions: N 转换
- 1029. Off-the-shelf software : 套装软件
- 1030. operational testing: 可操作性测试
- 1031. output domain: 输出域
- 1032. paper audit:书面审计
- 1033. Pair Programming : 成对编程
- 1034. partition testing: 分类测试
- 1035. Path coverage : 路径覆盖
- 1036. path coverage: 路径覆盖
- 1037. path sensitizing: 路径敏感性

- 1038. path testing: 路径测试
- 1039. path: 路径
- 1040. Peer review : 同行评审
- 1041. Performance : 性能
- 1042. Performance indicator: 性能（绩效）指标
- 1043. Performance testing : 性能测试
- 1044. Pilot : 试验
- 1045. Pilot testing : 引导测试
- 1046. Portability : 可移植性
- 1047. portability testing: 可移植性测试
- 1048. Positive testing : 正向测试
- 1049. Postcondition : 后置条件
- 1050. Precondition : 前提条件
- 1051. precondition: 预置条件
- 1052. predicate data use: 谓词数据使用
- 1053. predicate: 谓词
- 1054. Priority : 优先权
- 1055. program instrumenter: 程序插装
- 1056. progressive testing: 递进测试
- 1057. Prototype : 原型
- 1058. Pseudo code : 伪代码
- 1059. pseudo-localization testing: 伪本地化测试

- 1060. pseudo-random: 伪随机
- 1061. QC: 质量控制 (quality control)
- 1062. Quality assurance(QA): 质量保证
- 1063. Quality Control(QC) : 质量控制
- 1064. Race Condition: 竞争状态
- 1065. Rational Unified Process(以下简称 RUP):瑞理统一工艺
- 1066. Recovery testing : 恢复测试
- 1067. recovery testing: 恢复性测试
- 1068. Refactoring : 重构
- 1069. regression analysis and testing: 回归分析和测试
- 1070. Regression testing : 回归测试
- 1071. Release : 发布
- 1072. Release note : 版本说明
- 1073. release: 发布
- 1074. Reliability : 可靠性
- 1075. reliability assessment: 可靠性评价
- 1076. reliability: 可靠性
- 1077. Requirements management tool: 需求管理工具
- 1078. Requirements-based testing : 基于需求的测试
- 1079. Return of Investment (ROI) : 投资回报率
- 1080. review: 评审
- 1081. Risk assessment : 风险评估

- 1082. risk: 风险
- 1083. Robustness : 强健性
- 1084. Root Cause Analysis(RCA): 根本原因分析
- 1085. safety critical: 严格的安全性
- 1086. safety: (生命) 安全性
- 1087. Sanity testing : 健全测试
- 1088. Sanity Testing: 理智测试
- 1089. Schema Repository : 模式库
- 1090. Screen shot : 抓屏、截图
- 1091. SDP: 软件开发计划 (software development plan)
- 1092. Security testing : 安全性测试
- 1093. security testing: 安全性测试
- 1094. security.: (信息) 安全性
- 1095. serviceability testing: 可服务性测试
- 1096. Severity : 严重性
- 1097. Shipment : 发布
- 1098. simple subpath: 简单子路径
- 1099. Simulation : 模拟
- 1100. Simulator : 模拟器
- 1101. SLA(Service level agreement): 服务级别协议
- 1102. SLA: 服务级别协议 (service level agreement)
- 1103. Smoke testing : 冒烟测试

- 1104. Software development plan(SDP): 软件开发计划
- 1105. Software development process: 软件开发过程
- 1106. software development process: 软件开发过程
- 1107. software diversity: 软件多样性
- 1108. software element: 软件元素
- 1109. software engineering environment: 软件工程环境
- 1110. software engineering: 软件工程
- 1111. Software life cycle : 软件生命周期
- 1112. source code: 源代码
- 1113. source statement: 源语句
- 1114. Specification : 规格说明书
- 1115. specified input: 指定的输入
- 1116. spiral model : 螺旋模型
- 1117. SQAP SOFTWARE QUALITY ASSURENCE PLAN 软件质量保证计划
- 1118. SQL: 结构化查询语句 (structured query language)
- 1119. Staged Delivery: 分布交付方法
- 1120. state diagram: 状态图
- 1121. state transition testing : 状态转换测试
- 1122. state transition: 状态转换
- 1123. state: 状态
- 1124. Statement coverage : 语句覆盖

- 1125. statement testing: 语句测试
- 1126. statement: 语句
- 1127. Static Analysis: 静态分析
- 1128. Static Analyzer: 静态分析器
- 1129. Static Testing: 静态测试
- 1130. statistical testing: 统计测试
- 1131. Stepwise refinement : 逐步优化
- 1132. storage testing: 存储测试
- 1133. Stress Testing : 压力测试
- 1134. structural coverage: 结构化覆盖
- 1135. structural test case design: 结构化测试用例设计
- 1136. structural testing: 结构化测试
- 1137. structured basis testing: 结构化的基础测试
- 1138. structured design: 结构化设计
- 1139. structured programming: 结构化编程
- 1140. structured walkthrough: 结构化走读
- 1141. stub: 桩
- 1142. sub-area: 子域
- 1143. Summary: 总结
- 1144. SVVP SOFTWARE Vevification&Validation PLAN: 软件验证和确认计划
- 1145. symbolic evaluation: 符号评价

- 1146. symbolic execution:参考符号执行
- 1147. symbolic execution: 符号执行
- 1148. symbolic trace: 符号轨迹
- 1149. Synchronization : 同步
- 1150. Syntax testing : 语法分析
- 1151. system analysis: 系统分析
- 1152. System design : 系统设计
- 1153. system integration: 系统集成
- 1154. System Testing : 系统测试
- 1155. TC TEST CASE 测试用例
- 1156. TCS TEST CASE SPECIFICATION 测试用例规格说明
- 1157. TDS TEST DESIGN SPECIFICATION 测试设计规格说明书
- 1158. technical requirements testing: 技术需求测试
- 1159. Test : 测试
- 1160. test automation: 测试自动化
- 1161. Test case : 测试用例
- 1162. test case design technique: 测试用例设计技术
- 1163. test case suite: 测试用例套
- 1164. test comparator: 测试比较器
- 1165. test completion criterion: 测试完成标准
- 1166. test coverage:测试覆盖

- 1167. Test design : 测试设计
- 1168. Test driver : 测试驱动
- 1169. test environment:测试环境
- 1170. test execution technique:测试执行技术
- 1171. test execution:测试执行
- 1172. test generator:测试生成器
- 1173. test harness:测试用具
- 1174. Test infrastructure : 测试基础建设
- 1175. test log:测试日志
- 1176. test measurement technique:测试度量技术
- 1177. Test Metrics :测试度量
- 1178. test procedure:测试规程
- 1179. test records:测试记录
- 1180. test report:测试报告
- 1181. Test scenario : 测试场景
- 1182. Test Script.:测试脚本
- 1183. Test Specification:测试规格
- 1184. Test strategy : 测试策略
- 1185. test suite:测试套
- 1186. Test target : 测试目标
- 1187. Test ware : 测试工具
- 1188. Testability : 可测试性

- 1189. testability:可测试性
- 1190. Testing bed : 测试平台
- 1191. Testing coverage : 测试覆盖
- 1192. Testing environment : 测试环境
- 1193. Testing item : 测试项
- 1194. Testing plan : 测试计划
- 1195. Testing procedure : 测试过程
- 1196. Thread testing : 线程测试
- 1197. time sharing:时间共享
- 1198. time-boxed : 固定时间
- 1199. TIR test incident report 测试事故报告
- 1200. ToolTip:控件提示或说明
- 1201. top-down testing:自顶向下测试
- 1202. TPS TEST PROCESS SPECIFICATION 测试步骤规格说明
- 1203. Traceability : 可跟踪性
- 1204. traceability analysis:跟踪性分析
- 1205. traceability matrix:跟踪矩阵
- 1206. Trade-off : 平衡
- 1207. transaction:事务/处理
- 1208. transaction volume: 交易量
- 1209. transform. analysis:事务分析
- 1210. trojan horse:特洛伊木马

- 1211. truth table:真值表
- 1212. TST TEST SUMMARY REPORT 测试总结报告
- 1213. Tune System : 调试系统
- 1214. TW TEST WARE :测试件
- 1215. Unit Testing :单元测试
- 1216. Usability Testing:可用性测试
- 1217. Usage scenario : 使用场景
- 1218. User acceptance Test : 用户验收测试
- 1219. User database :用户数据库
- 1220. User interface(UI) : 用户界面
- 1221. User profile : 用户信息
- 1222. User scenario : 用户场景
- 1223. V&V (Verification & Validation) : 验证&确认
- 1224. validation :确认
- 1225. verification :验证
- 1226. version :版本
- 1227. Virtual user : 虚拟用户
- 1228. volume testing:容量测试
- 1229. VSS (visual source safe) :
- 1230. VTP Verification TEST PLAN 验证测试计划
- 1231. VTR Verification TEST REPORT 验证测试报告
- 1232. Walkthrough : 走读

- 1233. Waterfall model : 瀑布模型
- 1234. Web testing : 网站测试
- 1235. White box testing : 白盒测试
- 1236. Work breakdown structure (WBS): 任务分解结构
- 1237. Zero bug bounce (ZBB): 零错误反弹

- 1238. 五、
- 1239. 计算机测试常用英语词汇
- 1240. Actual Fix Time 实际修改时间
- 1241. Assigned To 被分配给
- 1242. Closed in Version 被关闭的版本
- 1243. Closing Date 关闭日期
- 1244. Defect ID 缺陷编号
- 1245. Description 描述
- 1246. Detected By 被（谁）发现
- 1247. Detected in Version 被发现的版本
- 1248. Detected on Date 被发现的日期
- 1249. Estimated Fix Time 估计修改的时间
- 1250. Modified 修正
- 1251. Planned Closing Version 计划关闭的版本
- 1252. Priority 优先级
- 1253. Project 项目

- 1254. R&D Comments 研发人员备注
- 1255. Reproducible 可重现
- 1256. Severity 严重程度
- 1257. Status 状态
- 1258. Summary 概要
- 1259. Creation Date 创建日期
- 1260. Description 描述
- 1261. Designer 设计人员
- 1262. Estimated DevTime 估计设计和生成测试的时间
- 1263. Execution Status 执行状态
- 1264. Modified 修正
- 1265. Path 路径
- 1266. Status 状态
- 1267. Steps 步骤
- 1268. Template 模版
- 1269. Test Name 测试名称
- 1270. Type 类型
- 1271. Actual 实际结果
- 1272. Description 描述
- 1273. Exec Date 执行日期
- 1274. Exec Time 执行时间
- 1275. Expected 期望结果

- 1276. Source Test 测试资料
- 1277. Status 状态
- 1278. Step Name 步骤名称
- 1279. Duration 执行的期限
- 1280. Exec Date 执行日期
- 1281. Exec Time 执行时间
- 1282. Host 主机
- 1283. Operating System 操作系统
- 1284. OS Build Number 操作系统生成的编号
- 1285. OS Service Pack 操作系统的服务软件包
- 1286. Run Name 执行名称
- 1287. Run VC Status 执行 VC 的状态
- 1288. Run VC User 执行 VC 的用户
- 1289. Run VC Version 执行 VC 的版本
- 1290. Status 状态
- 1291. Test Version 测试版本
- 1292. Tester 测试员
- 1293. Attachment 附件
- 1294. Author 作者
- 1295. Cover Status 覆盖状态
- 1296. Creation Date 创建日期
- 1297. Creation Time 创建时间

- 1298. Description 描述
- 1299. Modified 修正
- 1300. Name 名称
- 1301. Priority 优先级
- 1302. Product 产品
- 1303. ReqID 需求编号
- 1304. Reviewed 被检查
- 1305. Type 类型
- 1306. Exec Date 执行日期
- 1307. Modified 被修正
- 1308. Planned Exec Date 计划执行的日期
- 1309. Planned Exec Time 计划执行的时间
- 1310. Planned Host Name 计划执行的主机名称
- 1311. Responsible Tester 负责测试的人员
- 1312. Status 状态
- 1313. Test Version 测试的版本
- 1314. Tester 测试员
- 1315. Time 时间
- 1316. Close Date 关闭日期
- 1317. Description 描述
- 1318. Modified 修正
- 1319. Open Date 开放日期

1320. Status 状态

1321. Test Set 测试集合

1322. Acceptance testing (验收测试), 系统开发生命周期方法论的一个阶段, 这时相关的用户和 / 或独立测试人员根据测试计划和结果对系统进行测试和接收。它让系统用户决定是否接收系统。它是一项确定产品是否能够满足合同或用户所规定需求的测试。这是管理性和防御性控制。

1323. Ad hoc testing (随机测试), 没有书面测试用例、记录期望结果、检查列表、脚本或指令的测试。主要是根据测试者的经验对软件进行功能和性能抽查。随机测试是根据测试说明书执行用例测试的重要补充手段, 是保证测试覆盖完整性的有效方式和过程。

1324. Alpha testing (α 测试), 是由一个用户在开发环境下进行的测试, 也可以是公司内部的用户在模拟实际操作环境下进行的受控测试, Alpha 测试不能由程序员或测试员完成。

1325. Automated Testing (自动化测试), 使用自动化测试工具来进行测试, 这类测试一般不需要人干预, 通常在 GUI、性能等测试中用得较多。

1326. Beta testing (β 测试), 测试是软件的多个用户在一个或多个用户的实际使用环境下进行的测试。开发者通常不在测试现场, Beta 测试不能由程序员或测试员完成。

1327. Black box testing (黑盒测试), 指测试人员不关心程序具体如何实现的一种测试方法。根据软件的规格对软件进行各种输入和

观察软件的各种输出结果来发现软件的缺陷的测试，这类测试不考虑软件内部的运作原理，因此软件对用户来说就像一个黑盒子。

1328. **Bug**（错误），有时称作 **defect**（缺陷）或 **error**（错误），软件程序中存在的编程错误，可能会带来不必要的副作用，软件的功能和特性与设计规格说明书或用户需求不一致的方面。软件缺陷表现特征为：软件未达到产品说明书标明的功能；软件出现产品说明书指明不会出现的错误；软件功能超出产品说明书指明的范围；虽然产品说明书未指出但是软件应达到的目标；软件测试人员或用户认为软件难以理解，不易使用，运行速度缓慢等问题。

1329. **Bug report**（错误报告），也称为“**Bug record**（错误记录）”，记录发现的软件错误信息的文档，通常包括错误描述、复现步骤、抓取的错误图像和注释等。

1330. **Bug tracking system**（错误跟踪系统，**BTS**），也称为“**Defect tracking system, DTS**”，管理软件测试缺陷的专用数据库系统，可以高效率地完成软件缺陷的报告、验证、修改、查询、统计、存储等任务。尤其适用于大型多语言软件的测试管理。

1331. **Build**（工作版本），软件开发过程中用于内部测试的功能和性能等不完善的软件版本。工作版本既可以是系统的可操作版本，