



广东外语外贸大学国际服务外包人才培养系列教材

*An Intermediate English Reading and Writing Course
On Software Testing For IT Professionals*

软件测试中级英语阅读与写作教程

许罗迈 主编



暨南大学出版社
JINAN UNIVERSITY PRESS



广东外语外贸大学国际服务外包人才培养系列教材

*An Intermediate English Reading and Writing Course
On Software Testing For IT Professionals*

软件测试中级英语阅读与写作教程

许罗迈 主编



暨南大学出版社
JINAN UNIVERSITY PRESS

中国·广州

图书在版编目 (CIP) 数据

软件测试中级英语阅读与写作教程/许罗迈主编. —广州: 暨南大学出版社, 2014. 1

(广东外语外贸大学国际服务外包人才培养系列教材)

ISBN 978 - 7 - 5668 - 0779 - 3

I. ①软… II. ①许… III. ①软件—测试—英语—阅读教学—高等学校—教材②软件—测试—英语—写作—高等学校—教材 IV. ①H31

中国版本图书馆 CIP 数据核字(2013)第 228407 号

出版发行: 暨南大学出版社

地 址: 中国广州暨南大学

电 话: 总编室 (8620) 85221601

营销部 (8620) 85225284 85228291 85228292 (邮购)

传 真: (8620) 85221583 (办公室) 85223774 (营销部)

邮 编: 510630

网 址: <http://www.jnupress.com> <http://press.jnu.edu.cn>

排 版: 广州市天河星辰文化发展部照排中心

印 刷: 湛江日报社印刷厂

开 本: 787mm × 1092mm 1/16

印 张: 6

字 数: 110 千

版 次: 2014 年 1 月第 1 版

印 次: 2014 年 1 月第 1 次

印 数: 1—2000 册

定 价: 15.00 元

(暨大版图书如有印装质量问题, 请与出版社总编室联系调换)

“广东外语外贸大学国际服务外包人才 培训系列教材”编委会

主任：隋广军

副主任：顾也力 郑建荣

委员：（按姓氏笔画排序）

毕惠阳 李铁立 林吉双 姜灵敏 黄立军

黄永智 蒋吉频 曾 增 熊海涛 魏 青

总 序

自 21 世纪以来，我国承接美欧日等国家和地区的国际服务外包呈加速发展之势。2012 年，我国承接国际服务外包执行金额为 336.4 亿美元，现已成为全球第二大服务外包接包国。伴随着服务外包产业的迅速发展，我国能熟练从事国际服务外包业务中高端人才的短缺问题日益突显出来。因此，尽快培养国际服务外包产业所需的中高端人才，已成为促进我国服务外包产业持续、快速和健康发展的当务之急。

广东外语外贸大学国际服务外包研究院和国际服务外包人才培养基地是全国普通高等院校中最早成立的有关国际服务外包研究和人才培养的专门机构。2009 年 10 月以来，国际服务外包研究院承接国际服务外包的理论研究和政府咨询等课题 40 余项，发表论文 200 余篇。目前，广东外语外贸大学国际服务外包研究院已成为华南地区国际服务外包理论研究中心、政府决策咨询智库。四年来，广东外语外贸大学国际服务外包人才培养基地共培训软件架构师、软件测试工程师和网络工程师等 IT 类高校“双师型”教师 150 余人；培养和培训 ITO、BPO、KPO 等适用型大学毕业生 2 000 余人；为 IBM、西艾、从兴等服务外包企业定制培训服务外包商务英语和相关业务流程专业人才 500 余人；培训服务外包企业和政府中高层管理人员近 7 000 人。经过几年来对服务外包人才培养模式与实践的有益探索，广东外语外贸大学国际服务外包人才培养基地已成为广东省服务外包“双师型”教师资源库、大学毕业生适用型人才交付中心、企业和政府管理人员短期进修中心。

广东外语外贸大学作为广东省国际服务外包高端人才培养基地，为更好地发挥其在国际化人才培养上的优势，进一步提高国际服务外包人才培养的质量，特组织专家学者编写了本套教材。本套教材包括《服务外包企业战略管理》、《服务外包项目管理》、《服务外包客户关系管理》、《商务沟通英语》、《商务会谈技巧英语》、《商务谈判日语》、《商务交际日语》、《软件开发中级英语阅读与写作教程》和《软件测试中级英语阅读与写作教程》，共 9 本。

培训服务外包产业所需的中高端人才是一项系统工程，其中，编写出能





够既反映服务外包发展理论，又符合服务外包发展实践的教材就尤其重要。我们希望本套教材的出版能够为服务外包人才的培养尽一份力量；同时，我们也真诚地欢迎各位读者对本套教材的不足之处提出修改的意见和建议，以期进一步提高我们教材编写的质量。

广东外语外贸大学国际服务外包人才培养系列教材编委会
2013年5月

前 言

对于中国的软件企业来说，提高竞争力、积极参与国际软件外包服务市场的竞争是一条合适的道路，因此企业员工必须具备一定的英文写作能力。

本书作者通过对广东若干具有代表性的国际软件外包服务公司做调研，发现这些企业的软件工程师都只接受过大学（公共）英语教育，英语水平有限，但是他们却要在编写程序的同时承担英文技术文件的撰写工作。调研也发现，他们面临的最大困难是写出符合英语表达习惯的规范文章。根据目前国内软件工程师的实际英语水平，面向较大学习群体，本教材定位为中级英文阅读与写作教材。教材包括8个单元，按2周学完1个单元计算，适合一个学期使用。

本教材的教学目标：

- 掌握英语最常用的2 000个单词以及1 000个左右常用软件测试文档词汇。
- 能够读懂使用常用表达方式表述的各类软件测试文件。
- 能够借助词典等工具，使用规范英语语法写出若干种简单语法结构，如名词短语、动词短语、介词短语、简单句。能够在填写软件测试文件各个栏目时使用这些简单语法结构。
- 学会尽量使用简单语法结构，避免产生语法错误。

本教材特点：

- 阅读内容采用各类较新的软件测试专业文件，并进行必要的改编，以适合教学对象的水平。
- 侧重选择具有软件测试模板性质的文本，以便学习者在实际工作中模仿套用。
- 根据上下文提供常用词汇表，相关词汇只采用相应的专业释义。
- 提供词汇使用规则说明。
- 针对教学对象的英语水平编写各种阅读理解练习。题目指示用双语，





问题用中文，以方便学习者完全理解阅读材料。

- 为各篇范文编写写作练习。遵循从易到难的原则，提供从客观判断题到主观写作题等各种练习题型。
- 写作练习尽量模拟软件测试实际工作中常见的形式。
- 注意培养学生使用词典等工具查找并选择合适的英文表达的能力。
- 为各类练习提供参考答案。

本教材内容的选择及处理方法：

- 本书阅读材料和写作材料全部选自真实的软件测试文件。
- 为了符合中级英文阅读和写作水平，作者对原始材料全部进行了简化。
- 简化语法，确保 95% 的课文都仅使用名词短语、动词短语和简单句。
- 只教授一种常用复合句句型，且专注于通用性高、易掌握的用法。
- 用易懂的方式讲解语法，避免使用专用术语。
- 简化词汇，用常用词替换了低频词。
- 词汇表中只列入与文本内容相关的释义，易于阅读理解。
- 部分涉及语法的知识转为词汇学习形式，易于中级学习者习得。
- 简化过程中确保原文意思不变。
- 简化的课文向学习者表明，可以用简单语法和常用词汇撰写各种软件测试文件。
- 学习者可以模仿教材内容进行软件测试文件的写作。



Contents

(目 录)

总 序	1
前 言	1
Unit 1 What Is Software Testing(什么是软件测试)	1
1. Reading(阅读)	1
2. Writing(写作)	6
Unit 2 Test Plan 1(测试计划一)	10
1. Reading(阅读)	10
2. Writing(写作)	18
Unit 3 Test Plan 2(测试计划二)	21
1. Reading(阅读)	21
2. Writing(写作)	29
Unit 4 Test Cases(测试用例)	32
1. Reading(阅读)	32
2. Writing(写作)	38
Unit 5 Bug Report(缺陷报告)	41
1. Reading(阅读)	41
2. Writing(写作)	48





Unit 6 Functional/System Test (功能/系统测试)	50
1. Reading(阅读)	50
2. Writing(写作)	59
Unit 7 Performance Test (性能测试)	61
1. Reading(阅读)	61
2. Writing(写作)	71
Unit 8 Test Summary Report (测试总结报告)	74
1. Reading(阅读)	74
2. Writing(写作)	80
Keys(参考答案)	82

Unit 1 What Is Software Testing

(什么是软件测试)

1. Reading(阅读)

1.1 Read the following text

(阅读下面的文本)

Text 1

Within the Validation and Verification (V&V) process, there are two complementary approaches to system checking and analysis.

Software inspections or peer reviews analyze and check system representations such as requirements document, design diagrams and program source code. You can use inspections at all stages of a testing process. Inspections may be supplemented by some automatic analysis of the source text of a system or associated documents. Software inspections and automated analyses are static V&V techniques, so you do not need to run the software on a computer.

Software testing involves running an implementation of the software with testing data. You examine the outputs of the software and its operational behavior to check if it is performing as required. Testing is a dynamic technique of verification and validation.

Text 2

As I explained earlier, the software testing process has two distinct goals:

1. To demonstrate to the developer and the customer that the software meets its

requirements. For custom software, there should be at least one test for every requirement in the user and system requirements documents. For generic software products, there should be tests for all of the system features. Such features will be incorporated in the product release. Some systems may have an explicit acceptance testing phase. In this phase, the customer formally checks the delivered system against its specification.

2. To discover faults or defects in the software. Defect testing is concerned with rooting out all kinds of undesirable system behaviors, such as system crashes, unwanted interactions with other systems, incorrect computations and data corruption.

The first goal leads to validation testing. Here you expect the system to perform correctly. You use a given set of test cases reflecting the system's expected use. The second goal leads to defect testing. Here the test cases are designed to expose defects. The test cases can be deliberately obscure and need not reflect how the system is normally used. For validation testing, if the system performs correctly, then it is a successful test. For defect testing, if a test exposes a defect and causes the system to perform incorrectly, then it is a successful test.

The goal of software testing is to show this to system developers and customers; the software is good enough for operational use. Testing is a process intended to build confidence in the software.

Text 3

A general model of the testing process is shown in Fig. 1 - 1 below. Testing cases include test inputs, the expected output, and what is tested. Testing data are the inputs. Testing data can sometimes be generated automatically. Automatic test case generation is impossible. The output of the tests can only be predicted by people, and they understand what the system should do.

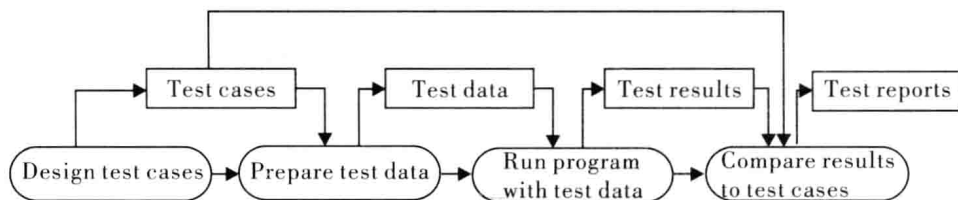


Fig. 1 - 1 A general model of the testing process

Exhaustive testing, where every possible program execution sequence is tested, is impossible. Testing has to be based on a subset of possible test cases. Ideally, software companies should have policies for choosing this subset rather than leave it to the development team. These policies might be based on general testing policies. For example, all program statements should be executed at least once. Alternatively, the making of testing policies may be based on experience of system usage and may focus on the testing of the features of the operational system. For example:

1. All system functions that can be accessed to in menus should be tested.
2. Combinations of functions (e. g. text formatting) accessed to in the same menu must be tested.
3. Where user input is provided, all functions must be tested with both correct and incorrect inputs.

1.2 High Frequency Vocabulary (常用词汇)

validation	<i>n.</i>	确认	verification	<i>n.</i>	验证
system	<i>n.</i>	系统	checking	<i>n.</i>	检查
analysis	<i>n.</i>	分析	software	<i>n.</i>	软件
inspection	<i>n.</i>	审核	analyze	<i>v.</i>	分析
requirement	<i>n.</i>	需求	document	<i>n.</i>	文档
design	<i>n.</i>	设计	diagram	<i>n.</i>	图表
program	<i>n.</i>	程序	source code		源代码
implementation	<i>n.</i>	实例, 实施	test case	<i>n.</i>	测试用例
data	<i>n. (pl.)</i>	数据	run	<i>v.</i>	运行
output	<i>n.</i>	输出	goal	<i>n.</i>	目标
developer	<i>n.</i>	开发者	customer	<i>n.</i>	客户
meet	<i>v.</i>	达到, 满足	behavior	<i>n.</i>	行为
custom software		定制软件	generic software		通用软件
product	<i>n.</i>	产品	incorporate	<i>v.</i>	整合
release	<i>n. /v.</i>	版本; 发布	acceptance	<i>n.</i>	接受, 接受度
phase	<i>n.</i>	阶段, 环节	deliver	<i>v.</i>	交货
specification	<i>n.</i>	规约	check	<i>v.</i>	检查
fault	<i>n.</i>	错误	defect	<i>n.</i>	缺陷
crash	<i>n. /v.</i>	崩溃	interaction	<i>n.</i>	交互

computation	<i>n.</i>	计算	corruption	<i>n.</i>	损坏
perform	<i>v.</i>	运行,表现	figure	<i>n.</i>	图,数字
expect	<i>v.</i>	预期,期待	statement	<i>n.</i>	说明, 程序语句
result	<i>n.</i>	结果	report	<i>n.</i>	报告
design	<i>v. /n.</i>	设计	generate	<i>v.</i>	生成
automatic	<i>adj.</i>	自动的	subset	<i>n.</i>	子集
policy	<i>n.</i>	策略,对策	execute	<i>v.</i>	执行, 运行
feature	<i>n.</i>	性能	function	<i>n.</i>	功能, 程序函数
access	<i>v.</i>	访问	menu	<i>n.</i>	菜单
user	<i>n.</i>	用户	example	<i>n.</i>	例子
process	<i>n.</i>	过程,流程	show	<i>v.</i>	显示
require	<i>v.</i>	要求,需求	against	<i>prep.</i>	对照

1.3 Language Points (语言点)

(1) 英语词汇主要可分类为名词(*n.*)、动词(*v.*)、形容词(*adj.*)等。名词表示物体、概念等,动词表示动作,形容词描述其后所接名词的性质,例如 generic。

(2) 动词通过加后缀可以变成名词或动名词。例如 test 是动词,加 ing 后成 testing,变成动名词;又例如 inspect 是动词,加 ion 后成 inspection,变成名词。学习者应该记住一些常用后缀。

(3) 规则动词词尾加 ed 或 ing 可以变成形容词。例如 expect 是动词,加 ed 后成 expected,或者加 ing 后成 expecting,变成形容词“预期的”。

(4) 英语名词分可数名词和不可数名词。可数名词如果要表示数量不止一个,一般要在词尾附加 s 或者 es,如 document 词尾加 s 成为 documents,表示“多个文件”。汉语词汇没有相应用法,因此中国人往往忘记在复数可数名词词尾加 s 或 es。不可数名词没有复数概念,不能加 s。中国人学习名词时应注意词典中有无[Un-Count]标志,如果有则是不可数名词。例如,不论你认为有多少源代码,都只能说 source code,而不能说 source codes。这也是中国人学习英语的一个难点。

(5) 有些可数名词变复数不是在单数拼写形式后面加 s 的规律变化,而是有较大差异的拼写形式。如 data 是复数,datum 是单数。注意不要在 data 后面

再加 s。这类所谓不规则名词的使用也是容易出错的。

1.4 Reading Comprehension (阅读理解)

1.4.1 Judge whether the following statements are true(T) or false (F) according to Text 1.

根据 Text 1 的内容判断下列说法正确与否。

- | | T | F |
|-------------------|-----|-----|
| (1) 软件测试都必须运行软件。 | () | () |
| (2) 软件测试包括审核各种文件。 | () | () |
| (3) 动态软件测试使用测试数据。 | () | () |

1.4.2 Judge whether the following statements are true(T) or false(F) according to Text 2.

根据 Text 2 的内容判断下列说法正确与否。

- | | T | F |
|-------------------------------|-----|-----|
| (1) 定制软件测试应该为每个需求规约设计对应的测试。 | () | () |
| (2) 缺陷测试应该消除各种系统崩溃、运算错误和数据损坏。 | () | () |
| (3) 通用软件测试可以只测试关键系统功能。 | () | () |
| (4) 缺陷测试用例不必测试正常运行情况。 | () | () |
| (5) 确认性测试应该证明软件能够正确运行。 | () | () |
| (6) 好的缺陷测试应该能发现导致软件运行出错的错误。 | () | () |

1.4.3 Judge whether the following statements are true(T) or false(F) according to Text 3.

根据 Text 3 的内容判断下列说法正确与否。

- | | T | F |
|--|-----|-----|
| (1) 测试用例通常包括输入说明、预期输出和简短说明。 | () | () |
| (2) 测试用例可以自动生成。 | () | () |
| (3) 测试过程包括设计测试用例、准备测试数据、用测试数据运行程序、比较测试用例和结果。 | () | () |
| (4) 软件公司应该安排开发人员负责编写软件测试用例。 | () | () |
| (5) 软件测试策略应该确保每条程序语句都在测试中被运行过。 | () | () |
| (6) 所有菜单项涉及的功能都应被测试。 | () | () |
| (7) 同一菜单项涉及的各种功能组合都应被测试。 | () | () |



(8)不必为所有功能都安排正确和错误的输入数据。

() ()

2. Writing(写作)

对于具备中级英语水平的软件人员来说,重要的是能够使用符合英语语法的语句表达意思。为了达到这个目标,首先应该掌握并能使用短语和简单句的语法,而避免使用复杂的复合句。时态方面掌握并使用一般现在时、一般将来时和简单过去时,而避免使用其他时态。软件文件中的主要的内容可以用名词短语或动词短语表达,其余部分可以用简单句表达。通过仅使用简单的语法结构,就可以写出基本符合英语语法而少犯错误的英文软件文件,从而使英文软件文件的质量达到足以让英语国家 IT 人员正确理解的水平。

在词汇使用方面,本教材由于篇幅有限,不可能讲授软件测试领域的所有词汇和表达法。本着授人以渔的理念,我们将介绍如何利用互联网,为头脑中的中文概念找到比较地道的英文表达。学习者利用合适的语法结构把这些英文词语连接起来,就能写出较为规范的英文。

2.1 From Chinese Ideas to English Expressions (从中文概念到英文表达)

许多英文词汇都是多义词,所以英汉词典中一个英语词汇往往对应多个中文释义。同样,许多中文词汇也都是多义词,所以汉英词典中一个中文词汇也往往对应多个英文释义。许多中式英文都是因为写作者为某些中文概念胡乱选择英文对应词而造成的。

互联网搜索引擎如百度、谷歌、必应等的问世,为我们提供了根据上下文寻找能较地道表达英文的有力工具。即使是接受过英语专业高学历教育,并具备较高的英汉口译、笔译能力和经验的人,也仍需要频繁使用搜索引擎查找、选择地道的英文专业词汇或术语。

当我们不知道某个中文概念如何用英语表达时,可以采用以下步骤进行查找:

(1)把中文词和可能同时使用的简单英文词作为关键词。例如,想找“软件需求规约”的英文说法,可以用“软件需求规约 + software”作为关键词。如果连 software 也不懂,还可以用 program,甚至 computer 之类更常见的英文词。使用步骤(1)里的关键词,在谷歌搜索的结果中,百度百科、CSDN 的内容都有标准的英文表达方式:software requirements specification。



(2) 选一个搜索引擎搜索上述关键词。推荐使用谷歌, 因为其搜索英语网站的结果相关度更高。

(3) 搜索引擎将会列举中英文词语同时存在的网页。

(4) 在搜索结果中选择比较地道的英文表达法。优先次序为外国网站的表达方式, 海外华人的网页内容, 跨国企业的中文网站, 专业网站如维基百科、百度百科、CSDN 等。避免采用中文小网站的内容, 因为那正是中式英语泛滥的地方。

(5) 有时第一轮查找得不到满意的结果, 则需要从该轮结果中选择看似更专业的英语词汇, 替换掉原来的英语词汇, 或者增加英语关键词, 减少汉语关键词, 进行下一轮搜索。

2.2 Search Exercises(搜索练习)

Follow the steps above, search for the appropriate English expressions for the Chinese ideas below.

根据上述步骤, 查找下列中文概念的英文表述。

- | | |
|------------|------------|
| (1) 软件测试计划 | (2) 测试自动化 |
| (3) 测试用例 | (4) 数据驱动测试 |
| (5) 单元测试 | (6) 错误报告 |
| (7) 测试总结报告 | (8) 负载测试 |

2.3 Noun Phrases(名词短语)

软件文件中至少 50% 的内容为短语。掌握英文短语的用法, 就可以在撰写英文软件文件时使将近 50% 的内容符合英语语法。英语中使用频率最高的短语是名词短语, 它除了可以独自表达意思外, 还可以构成其他更大的语言单位。

名词短语的核心是名词, 即词典中标注词性为 *n.* 的词。可以根据以下模式构成名词短语:

det. *o* + *adj.* *o* * + *n.* *

其中 *det.* 为冠词, 如 *a, the*; *adj.* 为形容词, 如 *simple, complex*; *n.* 为名词, 如 *software, program*。*o* 为可选, * 为可重复。

根据上述模式, 以下短语都是规范的英文名词短语:

- | | |
|----------------|-----------------------------|
| software | (<i>n.</i>) |
| software test | (<i>n.</i> *) |
| simple program | (<i>adj.</i> + <i>n.</i>) |