Tony Piper

Cambridge International AS and A level Computer Science Revision Guide

CAMBRIDGE UNIVERSITY PRESS

CAMBRIDGE
UNIVERSITY PRESS

University Printing House, Cambridge CB2 8BS, United Kingdom

Cambridge University Press is part of the University of Cambridge.

It furthers the University's mission by disseminating knowledge in the pursuit of education, learning and research at the highest international levels of excellence.

Information on this title: education.cambridge.org

First published 2016

Printed in Poland by Opolgraf

A catalogue record for this publication is available from the British Library

ISBN 9781107547544 Paperback

Contents

Revision Guidelines

PART I THEORY FUNDAMENTALS

Chapter 1 Information representation 2
Chapter 2 Communication and Internet technologies 12
Chapter 3 Hardware 30
Chapter 4 Processor fundamentals 41
Chapter 5 System software 55
Chapter 6 Security, privacy and data integrity 60
Chapter 7 Ethics and ownership 67
Chapter 8 Database and data modelling 74

PART II FUNDAMENTAL PROBLEM-SOLVING AND PROGRAMMING SKILLS

Chapter 9 Algorithm design and problem-solving 88
Chapter 10 Data representation 97
Chapter 11 Programming 106
Chapter 12 Software development 126

PART III ADVANCED THEORY

Chapter 13 Data representation 133
Chapter 14 Communication and Internet technologies 146
Chapter 15 Hardware 160
Chapter 16 System software 175
Chapter 17 Security 192
Chapter 18 Monitoring and control systems 200

PART IV FURTHER PROBLEM-SOLVING AND PROGRAMMING SKILLS

Chapter 19 Computational thinking and problem solving 209
Chapter 20 Algorithm design methods 236
Chapter 21 Further programming 245
Chapter 22 Software development 274

Answers 283

Revision guidelines

Revision, by the nature of the word, implies re-visiting content and topics that you have studied throughout the year. What you already have in terms of resources to help you with your revision will largely determine the way in which you set about and plan your revision programme.

Key issues include:

- Have you got a copy of the textbook you have followed throughout your course?
- Did you use it as your course progressed to make your own notes?
- Has your teacher provided you with notes as each topic has been covered?
- Have you worked through worksheets prepared by your teacher?

All of these are a good starting point and your first revision task is to gather together all the materials you have produced and accumulated throughout the course. Organise them in the same way as the 9608 syllabus, that is, by section and subsection.

When should I start revising?

Start as early as possible. Examinations are generally a stressful time and so you need to do everything possible to make this a 'stress-free' experience.

A trawl through all the materials you have should establish:

- what topics you have clear notes for and where you do not
- topics where you can do lots of practice, for example, the number systems content in Part 1, 1.01
- topics about which you are definitely confident
- topics that you are not confident with – you probably 'put it off' when the content was covered in lessons.

Preparing for examination

You must not have large gaps in your understanding and you need the skills to apply your knowledge. Both are important. The trend generally for all advanced level examinations is away from questions which only ask you to reproduce basic knowledge, for example giving a basic definition. For a question about database design a knowledge question could be 'State what is meant by a primary key and a foreign key'. However, it is a much better assessment of your ability if you are able to apply this to a given simple practical scenario. The question style you are more likely to face is:

1 Which attribute would be the primary key for table X?

2 How is the relationship formed using a foreign key to table Y?

Computing is a practical subject – probably second only to engineering – and so it is reasonable that your computing examination papers should reflect this, with questions that require answers which apply your knowledge in the context of practical scenarios.

Past examination paper questions

Looking at as many previous questions as possible can be a very valuable part of revision. Many examples from Cambridge past papers have been carefully selected and included at the end of each section in the course textbook. (Cambridge International Examinations bears no responsibility for the example answers to questions taken from its past question papers which are contained in this publication.)

Your teacher will be able to supply you with further past papers and specimen papers and guide you to relevant questions for the topic you are revising.

So, you've trawled through and organised the materials you have produced throughout the course – what next?

Specific revision materials

Cambridge International AS and A Level Computing Revision Guide

This book should provide you with a helpful structure to plan your revision around. The organisation of this guide is similar to the course textbook and it has frequent 'test yourself ' questions as you work through each chapter.

Revision cards

These are a favourite with students and have the obvious advantage that you can carry them around with you and dip into them in any odd five minutes you can find. Cards are available in different colours and so you could easily develop a system to code cards on the same general topic in the same colour.

The figures below are for Chapter 8, on databases:

Database design **Card 1 (of 10)**

Attribute – Data item recorded as part of a database design.
Entity – In database design, something about which we record data, for example, a Customer. Entities are implemented as tables.
Primary key – An attribute (or combination of attributes) chosen to ensure that all the records in a table are unique.
Relationship – A link between two tables, which can be:

- One-to-one – uncommon
- One-to-many – the most common
- Many-to-many – cannot be implemented with relational database software

Database design **Card 2 (of 10)**

Foreign key – An attribute in a table which links back to the same primary key attribute in a second table.
Candidate key – Attribute(s) which are unique in a table and so are a 'candidate' to be used as the primary key.
Secondary key – An attribute other than the primary key for which an index has been created.

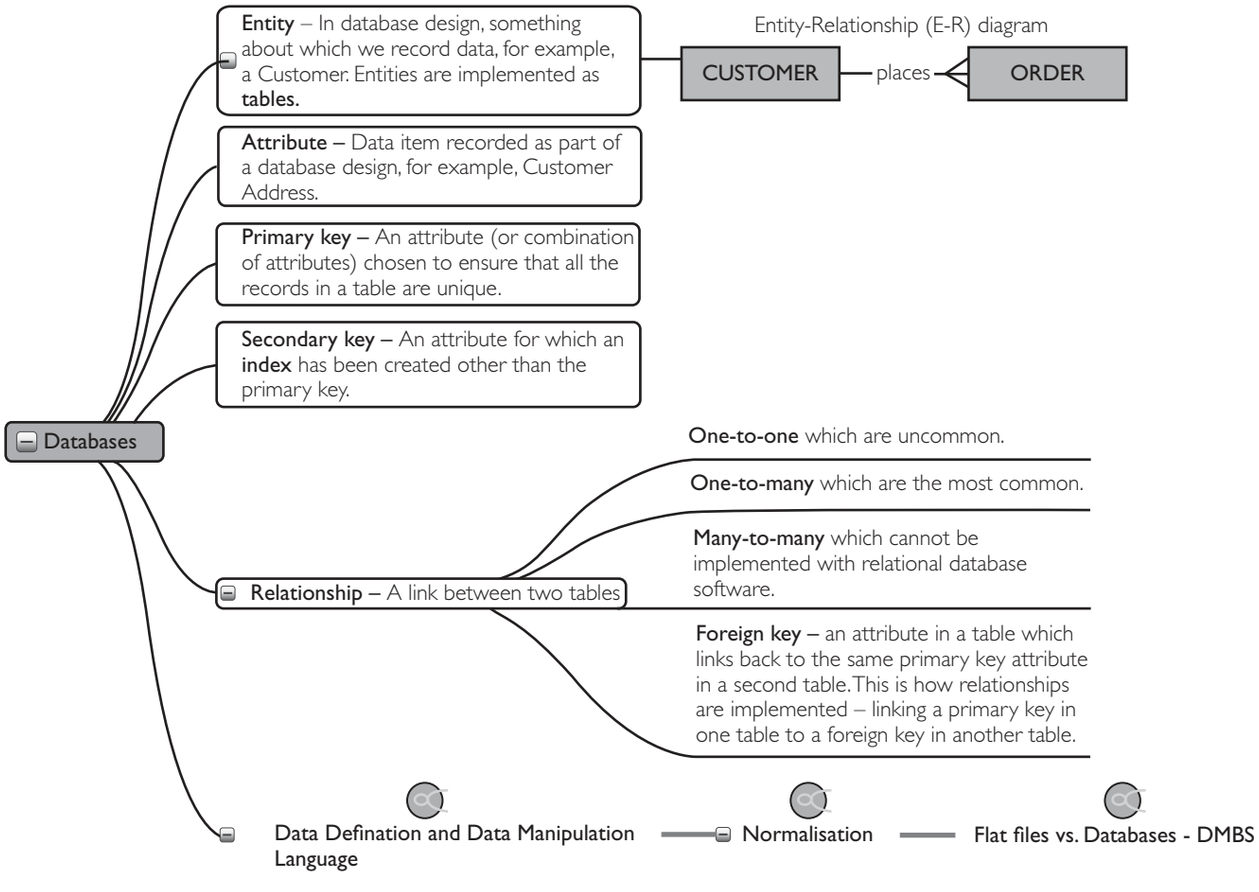
How will you organise the cards?

- a separate set for each section
- a separate set for each topic

There are some clear links between content in, for example, Part 1 and Part 3. Assembly language is introduced in Section 1 and then studied in more depth in Part 3. Can you have a system with your revision cards which allows for this?

Mind maps

Mind maps provide an effective way to break the content down into manageable amounts and if you are a person who 'thinks visually' then you will probably take to mind-mapping. My experience is that students tend to be polarised into 'I like using them' or 'I hate them' but I have found that students do agree they are a useful revision tool. A simple example for (some of) the database content for Chapter 8 is shown below:



Also there is available on the World Wide Web free mind-mapping software and this usually has features which are very appropriate for revision:

- the expansion of branches (to see detail)
- the collapsing of branches (to see the 'big picture')
- the inclusion of graphics.

How do I revise?

What time of day?

There are all sorts of conflicting evidence about when your brain is at its most receptive! You will need to decide what time of day seems to work best for you and how long each session should be.

Shall I revise on my own?

Maybe, but it will be much less daunting if you team up with a fellow student – a 'revision buddy' – and revise together. This could include:

- sharing the work of producing the revision cards or mind maps
- testing each other on some basic definitions and the factual knowledge of a topic.

I do lots of past examination questions

Consider carefully what 'doing examination questions' actually means for you. It is tempting to look at a question then, talking to yourself, recite the answer you would give – then move on to another question. That might be sufficient but, remember, the examination is a written paper; why not spend the extra time in writing out the answer on rough paper? That way when you read it back it may be clear that there are some points which you have omitted or some points where the meaning is unclear.

Seek advice

You need to be confident with all the syllabus content (remember, there is no choice of questions) so don't try to bury problems and topic areas about which you are unsure. Your revision buddy may be confident about it and after five minutes of him or her talking it through, something about which you have been unclear for six months, may become clear for the first time. Failing that, be honest that you are unsure and seek help from your teacher. Problems do not go away and solve themselves – you must be pro-active in plugging the gaps in your knowledge and understanding.

On the day – examination technique

Reading the paper

It is sensible to read the entire paper before you start to attempt any of the questions. This will give you a good idea as to the questions you are confident about and those which may need more time spent on them. The number of marks is a good indicator of how long you should spend on each question. Get used to planning how you will divide your time for questions through an examination paper using the marks as a guide. It is a good idea to read back through your answers once you finish.

Layout of the paper

Where questions are displayed in an answer booklet, the amount of space provided is an indicator of the length of answer the examiner is expecting.

Is it important to answer the questions in a paper in order? No, you can answer the questions in any order. As a general rule questions which are considered less demanding will be at the start of the paper.

Understanding the question command words

Some questions will have a short introduction (called the 'stem' of the question) and this will apply to all parts of the questions which follow. Specific questions will each have a keyword which is the indicator as to the style of answer expected.

Questions starting 'Define ...', 'State ...', 'Give ...' or 'Name …' all require an answer of only one or a few words giving a short and concise answer.

For example:

Give the attributes for the Loan table below, showing the primary key.

You should not create a `LoanID` for this table.

```
Loan (............, ............,
.............., ..............)
```
[2]

A question starting 'Describe ...' wants more detail. The indicator of precisely how much detail is the number of marks for the question: a three-mark question will usually require three different points to be made.

For example:

Describe how an assembly language program is translated into machine code. [2]

A question starting 'Explain ...' wants not only a description but an answer that contains some reasoning.

For example:

Explain why an interpreter has better diagnostics features than compiler software. [2]

A sample examination question is shown below:

(a) State what is meant by a real-time application.

There is no introductory statement.
The keyword is 'State' and what is wanted is the basic 'bookwork' definition of a real-time system.

[1]

(b) An air conditioning system is a real-time application.
Explain how sensors and actuators are used to control an air-conditioning system in an apartment.

The introductory statement applies to part (b) only
The keyword is 'Explain' and there are four marks.
The answer must make at least four clear points describing how a temperature sensor sends data values to the processor and how they are processed when an actuator is involved.

[4]

(c) Give ***one other*** example of a real-time application. Justify why your choice is a real-time application.
Example:

Justification:

The keyword is 'Give' but you are having to be more resourceful and come up with your own example of a real-time system.
The key requirements are the example and its justification. The paper makes it clear how you are to present them. You can assume there will be one mark for the example and one mark for the justification.

[2]

Cambridge International AS and A Level Computing 9691 Paper 31 Q4, June 2011