1.2 Types of Software and Development Methodologies: 30-minute Pack

Total Marks

OCR H446 · Calculators not allowed · about 30 minutes

Name: _____ Date: ____ / ____ / _____

Answer ALL questions. Show all your working.

1 Software can also be classified by whether or not its source code is publicly available.

(a) State what is meant by the term 'open source software'. **(2)**

(b) State what is meant by the term 'closed source (proprietary) software'. **(2)**

(c) A small business is choosing a content management system for its website and is considering an open source option. Give two advantages and two disadvantages of using open source software in this context. **(4)**

(Total for Question 1 is 8 marks)

2 Utility software carries out specific maintenance tasks on a computer system.

(a) State what is meant by the term 'utility software'. **(2)**

(b) For each scenario below, identify the type of utility software that would be used. **(4)**

i) Compressing a large video file before emailing it.

ii) Protecting a computer from malicious software downloaded from the internet.

iii) Reorganising fragmented files on a hard disk drive to improve access speed.

iv) Creating a copy of important files in case the originals are lost or damaged.

(c) Explain why defragmentation utility software improves performance on a hard disk drive (HDD) but has little or no benefit on a solid-state drive (SSD). **(3)**

(Total for Question 2 is 9 marks)

© Revision Library | revisionlibrary.co.uk

3 Rapid Application Development (RAD) focuses on fast delivery through prototyping.

(a) State two characteristics of RAD. **(2)**

(b) Explain why RAD may not be suitable for a project where users are not available to give regular feedback. **(3)**

(c) Compare the use of prototyping in RAD with the waterfall model. **(3)**

(Total for Question 3 is 8 marks)

© Revision Library | revisionlibrary.co.uk

Mark scheme · 1.2 Types of Software and Development Methodologies: 30-minute Pack

Question 1

(a) Source code is made publicly available/can be viewed by anyone (oe) **B1**

(a) ...and can usually be modified and redistributed freely (often at no cost) (oe) **B1**

(a) Answer: Open source software is software whose source code is publicly available so that it can be viewed, modified and redistributed freely.

(b) Source code is kept private/hidden by the owner/developer (oe) **B1**

(b) ...and users cannot legally view, modify or redistribute it without permission/a licence (oe) **B1**

(b) Answer: Closed source (proprietary) software is software whose source code is kept private by the owner, and cannot legally be viewed, modified or redistributed by users without permission.

(c) Advantage 1: any one from - free/low cost to acquire; can be customised to fit the business's needs; large community support/development; code can be inspected for security flaws; no vendor lock-in **B1**

(c) Advantage 2: a second distinct point from the list above **B1**

(c) Disadvantage 1: any one from - may lack official/guaranteed technical support; may require in-house technical expertise to customise/maintain; documentation may be poor or inconsistent; security vulnerabilities may be publicly known before being patched **B1**

(c) Disadvantage 2: a second distinct point from the list above **B1**

(c) Answer: Advantages: low/no cost; can be freely customised to the business's needs. Disadvantages: no guaranteed official support; may need in-house technical skill to maintain.

Question 2

(a) Software designed to help maintain, configure, analyse or optimise a computer system (oe) **B1**

(a) ...it carries out a single specific task, rather than a general task requested by the user (oe) **B1**

(a) Answer: Utility software is software designed to carry out a specific task that helps maintain, configure or optimise a computer system.

(b) i) Compression/archiving utility (oe) **B1**

(b) ii) Anti-virus/anti-malware utility (oe) **B1**

(b) iii) Defragmentation utility (oe) **B1**

(b) iv) Backup utility (oe) **B1**

(b) Answer: i) Compression utility ii) Anti-virus/anti-malware utility iii) Defragmentation utility iv) Backup utility

(c) HDDs use a moving read/write head over spinning platters, so fragmented data (split into non-contiguous blocks) increases seek/access time (oe) **B1**

(c) Defragmentation reorders data into contiguous blocks, reducing head movement and so reducing access time (oe) **B1**

(c) SSDs have no moving parts and can access any memory location in (near) constant time, so fragmentation does not increase access time / reorganising data brings little benefit and may cause unnecessary wear (oe) **B1**

(c) Answer: HDDs suffer from fragmentation because the read/write head must physically move further to reach scattered data, increasing seek time; defragmentation groups files into contiguous blocks to reduce this. SSDs have no moving head and access any location in roughly the same time, so fragmentation barely affects performance and defragmenting can add unnecessary wear.

© Revision Library | revisionlibrary.co.uk

Question 3

(a) One correct characteristic of RAD, from: heavy/repeated use of prototyping, refined based on user feedback; short development time/fast delivery; makes use of reusable components/tools; requires active, continuous user involvement **B1**

(a) A second, different correct characteristic from the same list. Maximum 2 marks; do not credit the same characteristic twice **B1**

(a) Answer: RAD makes heavy use of prototyping that is refined through repeated user feedback, and aims for fast delivery of a working system.

(b) RAD depends on regular/continuous input and feedback from users to refine prototypes at each stage (oe) **B1**

(b) Without this feedback, prototypes cannot be validated or improved against real user needs (oe, ft) **B1**

(b) This can lead to a final system that does not meet requirements, or to delays while the team waits for input, undermining the speed that is RAD's main benefit (oe, ft) **B1**

(b) Answer: RAD relies on continuous user feedback to refine each prototype; without available users, prototypes cannot be validated against real needs, risking a final system that does not meet requirements or delays that remove RAD's main speed advantage.

(c) RAD uses prototyping throughout/repeatedly, central to the process, refined via user feedback across the whole project (oe) **B1**

(c) Waterfall does not typically use prototyping, or uses it (if at all) only briefly at the start, since each stage is completed once, in order, before moving on (oe) **B1**

(c) Comparative point: RAD can therefore adapt to user feedback more easily/quickly than waterfall, which is more rigid once requirements/design are signed off (oe) **B1**

(c) Answer: RAD uses prototyping repeatedly and continuously throughout the project, refined by ongoing user feedback, whereas waterfall makes little or no use of prototyping since each stage is completed once before moving to the next; as a result RAD can adapt to user feedback far more easily than the more rigid waterfall model.

© Revision Library | revisionlibrary.co.uk