

Past-paper walk-through

Language Translators ■ 5.2

eXercise

Questions in this set

- 9618/12 N25 Q10 ■ 7 marks
- 9618/13 N25 Q8 ■ 10 marks
- 9618/13 J25 Q3 ■ 15 marks
- 9618/12 N24 Q4 ■ 10 marks
- 9618/11 J24 Q3 ■ 5 marks
- 9618/12 J24 Q8 ■ 10 marks

Reading the mark scheme

- **B1** a mark that stands on its own – the point is either made or not.
- **M1** a *method* mark: the right approach, even if the number is wrong.
- **A1** an *answer* mark, and it usually needs its M mark first.
- **C1** a working mark on the way to the answer.
- **//** separates answers the examiner accepts equally.
- **ecf** = error carried forward: a later part is marked on your own earlier answer.
- **owtte** = “or words to that effect” – the wording need not match.

Question 10

9618/12 N25 ■ Q10 ■ 7 marks

A programmer is developing a computer program.

(a) Explain how the programmer can first use an interpreter **and** then a compiler to develop the computer program. [4]

(b) The programmer releases the program as Free Software.

Describe what is meant by Free Software. [2]

(c) A user downloads the computer program from the internet.

State what should be included as part of the download to make sure the program is authentic. [1]

Solution 10

(a) Mark scheme

- 1 mark per bullet point, max 4 marks
- Max 2 from each section.
- Interpreter:
 - ■ Use an interpreter while writing/coding the program
 - ■ ... to test / debug the partially completed program
 - ■ ... because errors can be corrected and processing continue from where the execution stopped // errors can be corrected in real time // errors are identified one at a time

Solution 10

- Compiler:
 - ■ Use the compiler after the program is complete
 - ■ ... to create an executable file so source code not seen
 - ■ Use the compiler to repeatedly test the same completed section without having to re-interpret every time

Solution 10

(b)

Mark scheme

- 1 mark per bullet point, max 2
- ■ The software is free of restrictions
- ■ The software is not necessarily free of charge
- ■ The source code comes with the program // A user can edit the source code to add functionality / suit their needs
- ■ Users can share the software with others

Solution 10

(c)

Mark scheme

- 1 mark for a correct answer
- ■ Digital Signature

Question 8

9618/13 N25 ■ Q8 ■ 10 marks

Compilers and interpreters are used to translate programs written in a high-level language into a low-level language.

(a) State **two** disadvantages of using a compiler compared to an interpreter during program development.

1. 2. [2]

(b) Explain how a programmer benefits from using program libraries during program development. [3]

(c) A programmer is developing an Operating System (OS).

Identify **two** types of software licence the programmer can use to allow other people to edit and redistribute the OS.

1. 2. [2]

(d) Memory management is one key management task performed by an OS.

Question 8

Give **three** other key management tasks that are performed by an OS.

1. 2. 3.

[3]

Solution 8

(a) Mark scheme

- 1 mark per bullet point, max 2
- ■ Large amounts of source code take time to compile
- ■ It can be slower to produce the object code than an interpreter
- ■ The code must be recompiled when it is changed
- ■ The program cannot run if there are errors
- ■ It is not possible to correct errors in real-time
- ■ One error can cause false reporting of multiple further errors
- ■ Sections of code / unfinished code cannot easily be tested

Solution 8

(b) Mark scheme

- 1 mark per bullet point, max 3 marks
- ■ Programming time is saved as code does not have to be written from scratch
- ■ Testing time is saved as code is already tested / documented
- ■ A library routine is more likely to work, as code is already tested
- ■ Library routines automatically update if they are changed / improved
- ■ The programmer can use library routines to perform complex functions / procedures that they may not be able to write themselves

Solution 8

(c)

Mark scheme

- 1 mark per bullet point, max 2 marks
- ■ Open Source (Initiative)
- ■ Free Software (Foundation)

Solution 8

(d)

Mark scheme

- 1 mark for each bullet point, max 3 marks
- ■ File management
- ■ Security management
- ■ Hardware / peripheral management // input / output management
- ■ Process management

Question 3

9618/13 J25 ■ Q3 ■ 15 marks

A programmer is writing a program in a high-level language.

(a) Complete the description of compilers and interpreters by writing the missing words.

A compiler checks all the code before attempting to translate the program. If any errors are found, they are all reported at the same time and the program does not translate or run. If there are no errors found, the compiler produces ____ which can run without access to the ____.

An interpreter translates one line of code and then runs it, before moving to the next line of code. If the line of code has an error, the interpreter ____ and displays the error. The programmer can correct the error ____ and then the interpreter continues translating from that point.

[4]

Question 3

	Method	How the method protects the data
during transmission	 	
on the computer	 	

[4]

Question 3

9618/13 J25 ■ Q3 ■ 15 marks

A programmer is writing a program in a high-level language.

(b) The programmer needs to keep the program files secure on their computer and during electronic data transmission over the internet. The files are protected by a password.

Complete the table by identifying **one** other method of keeping the files secure during electronic data transmission **and one** other method of keeping the files secure on the computer.

State how each method protects the data.

The methods must be different.

	Method	How the method protects the data
during transmission		
on the computer		

Question 3

[4]

Question 3

9618/13 J25 ■ Q3 ■ 15 marks

A programmer is writing a program in a high-level language.

(c) The computer program allows users to play a game in a virtual world. Users can play the game from any computer with internet access using a web browser. The game allows users to interact with other users.

Explain the reasons why the statement 'This computer game uses a client-server model' is correct.

[4]

Question 3

9618/13 J25 ■ Q3 ■ 15 marks

A programmer is writing a program in a high-level language.

(d) Describe the possible consequences of the programmer **not** joining a professional ethical body. **[3]**

Solution 3

(a) Mark scheme

- 1 mark for each correctly completed term
 - an executable file/.exe
 - source/program code
 - stops
 - immediately/in real-time
- A compiler checks all of the code before attempting to translate the program.

Solution 3

- If any errors are found, they are all reported at the same time and the program does not translate or run. If there are no errors found, the compiler produces an executable file/.exe which can run without access to the source/program code.
- An interpreter translates one line of code and then runs it, before moving to the next line of code. If the line of code has an error, the interpreter stops and displays the error. The programmer can correct the error immediately/in real-time and then the interpreter continues translating from that point.

Solution 3

(b) Mark scheme

- 1 mark for method and 1 mark for corresponding description
- During transmission e.g.
 - ■ Encryption // by example such as VPN
 - ■ Jumble / encode data so it cannot be decrypted/understood without the key
- On computer e.g.
 - ■ Firewall / proxy
 - ■ Filter incoming transmissions and stop any that could be attempting unauthorised access

Solution 3

- ■ Anti-malware
- ■ Find and delete or quarantine any malware that could delete the data / files
- ■ Encryption
- ■ Jumble / encode data so it cannot be decrypted / understood without the key
- ■ Physical method // by example
- ■ For example, the computer storing the data cannot be accessed without the key to the room

Solution 3

(c) Mark scheme

- 1 mark each to max 4
- ■ A webserver stores all the data for each player
- ■ Each player is on a client computer // The player's web browser is the client
- ■ that sends requests over the internet to the web server
- ■ the server performs the required action in the game
- ■ the server updates the data in the game
- ■ the server sends the results to the player

Solution 3

(d) Mark scheme

- 1 mark each to max 3 e.g.
- ■ There is no / limited access to legal advice
- ■ in case action is taken against them
- ■ There are fewer networking opportunities
- ■ so they could miss out on contacts / jobs
- ■ There would be less access to training
- ■ There would be no clear laid out ethical guidelines
- ■ and / or people to discuss potential ethical problems with

Solution 3

- possibly leading to inappropriate / unethical actions
- which might lead to legal proceedings / recourse

Question 4

9618/12 N24 ■ Q4 ■ 10 marks

A program is written in a high-level language by a team of three programmers using an Integrated Development Environment (IDE).

(a) Describe how the programmers can use the debugging features of a typical IDE during the development of the program. [4]

(b) The programmers created a new program library whilst developing the program. Describe the benefits to the programmers of creating a program library. [3]

(c) The file containing the final program code will be sent by email for beta testing. Identify **one** security method that can be used to protect the program code from unauthorised access during email transfer.

Explain how your chosen method protects the program code. [3]

Solution 4

(a) Mark scheme

- 1 mark for each bullet point (max 4) e.g.
- ■ Single stepping
- ■ Run the program one line at a time
- ■ ... and check the variable contents / program flow // show the effect of each line of code
- ■ Set breakpoints
- ■ ... run the code up to a set line
- ■ ... and then check the status

Solution 4

- ■ Variable/report watch window
- ■ ... view how the data changes as the program is running

Solution 4

(b) Mark scheme

- 1 mark for each bullet point (max 3) e.g.
- ■ Subroutines can be shared / reused
- ■ ... between team members who are working independently
- ■ ... without having to rewrite/re-test them which saves the programmers' time
- ■ A program library provides continuity between programs/programmers
- ■ Individual programmers can contribute their specialisms to the library //
- Individual programmers can use the specialisms of others

Solution 4

(c) Mark scheme

- 1 mark for the security method. 2 marks for explanation
- Security method:
 - Encryption
 - Explanation
 - ■ File contents are converted to cipher text
 - ■ If intercepted the data cannot be understood without the decryption key
- 3
- October/November

Solution 4

- 2024

Question 3

9618/11 J24 ■ Q3 ■ 5 marks

3 A software developer is writing a computer program.

- (a)** The developer uses an interpreter while writing the program code because it is easier for debugging.

Explain **one** reason why it is easier to debug the program code using an interpreter instead of a compiler.

Question 3

- (b)** The program is ready to be sold to customers.

The developer uses a compiler because it creates an executable file.

Explain the reasons why the need to create an executable file makes the compiler the appropriate choice when the program is complete.

Solution 3

(a) Mark scheme

- 1 mark each to max 2:
 - ■
- The interpreter will stop when an error is found
- ■
- ... so the error can be corrected in real-time, and the result of changes
- seen immediately
- ■
- Only one error is displayed at a time

Solution 3

- ■
- ... so fewer errors to correct simultaneously and no dependent errors
- 2

Solution 3

(b) Worked steps

The developer used an **interpreter while writing** because it reports errors line by line. The question now asks why a **compiler** is used for the finished product, so every point must be about **distribution**, not about debugging. Three marks from:

- The program can be distributed **without the source code** ...
- ... so it **cannot be edited, copied or plagiarised**.
- Users **do not need the translator** to run it ...
- ... so no time is spent **retranslating** every time the program is run.

Solution 3

The pattern is *fact, then consequence*, and the marks pair up that way — a bare “it is faster” is a consequence with no cause behind it.

The trade-off is worth naming, because it is why both translators exist. An interpreter gives quick feedback while the code is changing; a compiler produces a fast, self-contained executable once it has stopped changing. Neither is better in general.

Mark scheme

- 1 mark each to max 3:
 - ■
 - Program can be distributed without source code
 - ■

Solution 3

- ... so it cannot be edited/stolen/plagiarised
- ■
- Users do not require the translator to run the program
- ■
- ... so time is not spent retranslating by user
- 3

Question 8

9618/12 J24 ■ Q8 ■ 10 marks

8 A programmer uses an Integrated Development Environment (IDE) to write a computer program. The IDE has both a compiler and an interpreter as built-in translators.

(a) The programmer decides to use the compiler when testing the final program.

Describe the benefits of using the compiler during testing.

Question 8

(b) IDEs have many features other than built-in translators.

Complete the table by identifying **one other** common IDE feature that can be used for each purpose. Describe how each feature helps the user during program development.

Each feature must be different. Do **not** give translator as one of your features.

Purpose	IDE feature	Description
for coding		

Question 8

for presentation		
for debugging		

Question 8

--	--	--

[6]

- (c) The programmer uses program libraries when developing the program.

Describe **two** benefits to the programmer of using program libraries.

Question 8

[2]

Solution 8

(a) Mark scheme

- 1 mark each to max 2:
 - ■
- Creates an executable file
- ■
- ... so the code can be tested multiple times without having to recompile
- ■
- ... so repeated testing takes less time
- 2

Solution 8

(b) Worked steps

Each feature earns one mark for the **name** and one for a **matching description**, so write them as pairs and keep them in the category the question asks for.

For coding:

- **Context-sensitive prompts** — suggest the code as you type, so the exact syntax and parameters need not be remembered.
- **Auto-correct / auto-complete** — fixes or finishes what is typed, so there are fewer errors to correct later.

For presentation:

- **Prettyprint / auto-indentation** — lays the code out so its block structure is visible.

Solution 8

- **Syntax highlighting** — colours keywords, strings and comments differently, so a mistyped keyword shows up immediately.
- **Expand and collapse code blocks** — lets a long program be read one section at a time.

Keep the two categories apart. A presentation feature offered as a coding feature scores the identification and loses the description mark, because the description will not match what was asked. [Mark scheme](#)

- 1 mark for identification of each feature and 1 mark for matching description:
- e.g.
- For coding:

Solution 8

- IDE feature
- Description
- Context-sensitive
- prompts
- Gives suggestions for code as the user
- types instead of having to write/remember
- the code
- Auto-correct
- Corrects spelling mistakes so that user has
- fewer errors to correct

Solution 8

- For presentation:
- IDE feature
- Description
- Pretty-printing
- Colour code keywords so the user can
- identify any errors
- Expand/collapse (code)
- blocks
- The user can hide code that they are not
- currently working on

Solution 8

- For debugging:
- IDE feature
- Description
- Single stepping
- Run the code one line at a time // shows
- the effect of each line of code
- Breakpoints
- Stop the code running at a set point to
- check the flow/variable contents
- 6

Solution 8

(c) Worked steps

Two marks for benefits of using **library routines** rather than writing your own:

- **Saves time**, because the code does not have to be written or tested from scratch.
- The code is **already tested**, so it is more likely to work and more robust.
- The programmer **does not have to maintain it** — the library is updated by its author.
- It can perform **complex operations** the programmer might be unable to write.
- It makes the calling code **easier to read**, since one call replaces a block of detail.

Solution 8

The strongest pair to give is *tested* and *maintained*: those are the two things a library provides that time alone does not explain, and together they are why an experienced programmer reaches for one before writing anything. [Mark scheme](#)

- 1 mark each to max 2:
- ■
- Saves programming/testing time as code does not have to be written/re-
- written from scratch // code does not have to be tested
- ■
- Code is already tested so it is more robust/likely to work

Solution 8

- ■
- The programmer does not need to maintain the library // library routines
- are updated automatically
- ■
- Can perform complex calculations that the programmer may be unable to
- do
- ■
- Makes code more easily readable
- 2