

[illegible]

Functional programming languages support higher-order functions such as `map` and `fold`.

0 1 . 3 Explain what a higher-order function is.

[2 marks]

0 1 . 4 What is the result of this application of the `fold` function?

`fold (*) 1 [2, 3, 2]`

[1 mark]

Result

0 2

In a functional programming language, six functions named `fu`, `fv`, `fw`, `fx`, `fy` and `fz` and a list of temperatures in Fahrenheit named `temps` are defined as shown in **Figure 8**.

Figure 8

```

temps = [50, 68, 95, 86]

fu a = (a - 32) * 5 / 9

fv b = map fu b

fw [] = 0
fw (x:xs) = 1 + fw (xs)

fx [] = 0
fx (x:xs) = x + fx (xs)

fy c = fx (c) / fw (c)

fz d = fy (fv (d))

```

A temperature can be converted from degrees Fahrenheit to degrees centigrade using the following method:

$$\text{centigrade} = (\text{Fahrenheit} - 32) \times \frac{5}{9}$$

For example, 59 degrees Fahrenheit is equivalent to 15 degrees centigrade.

In the functions `fw` and `fx`:

- `[]` is the empty list
- `(x:xs)` lets the function definition refer to the head of the list as `x` and the tail as `xs`.

0 2 . 1

Shade **one** lozenge to indicate which of the listed functions from **Figure 8** includes a higher-order function in its definition.

[1 mark]

`fu` ☐ `fv` ☐ `fx` ☐ `fy` ☐

0 2 . 2

Shade **two** lozenges to indicate which of the listed functions from **Figure 8** use recursion in their definitions.

[1 mark]

`fu` ☐ `fv` ☐ `fw` ☐ `fx` ☐

02.3

Calculate the results of making the function calls listed in **Table 2**, using the functions and list in **Figure 8** as appropriate.

[4 marks]

Table 2

Function call	Result
<code>fu 50</code>	
<code>fv temps</code>	
<code>fw temps</code>	
<code>fz temps</code>	

02.4

Explain the purpose of the function `fz`.

[1 mark]

02.5

It is proposed that the definition of the function `fz` is changed to:

$$fz\ d = fu\ (fy\ (d))$$

Explain why this new definition of `fz` could be considered to be an improvement over the definition of `fz` in **Figure 8**.

[1 mark]

0	3	1
---	---	---

One approach to dealing with Big Data is to write code that can be distributed to run across more than one server.

State **two** features of functional programming languages that make it easier to write code that can be distributed to run across more than one server.

[2 marks]

Feature 1 _____

Feature 2 _____

A functional programming function f has the function type:

$$f: \mathbb{N} \rightarrow \mathbb{R}$$

0	4	.	1
---	---	---	---

Describe **two** features of functional programming languages that make it easier to write code that can be distributed to run across multiple servers.

[2 marks]

Feature 1 _____

Feature 2 _____
