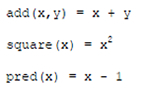
**Q1.**

The information below shows three commonly used mathematical functions: add, square and pred.



For example:

•   add(3,2) evaluates to 5

•   square(2) evaluates to 4

•   pred(8) evaluates to 7

The domain of the functions square and pred in the information above is the set of integers ℤ and the domain of the add function is ℤ × ℤ.

(a)  What is the co-domain of the pred function?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(b)  What is the result of applying square ◦ pred to the argument 3?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(c)  The add function takes two arguments.

Describe how the add function could be partially applied to the arguments 4 and 6.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(3)**

**(Total 5 marks)**

**Q2.**

In a functional programming language, four functions named fw, fx, fy and fz and a list named sales are defined as shown in the figure below.

fw [a,b] = a \* b

fx c = map fw c

fy d = fold (+) 0 d

fz e = fy (fx e)

sales = [[10,2], [2,25], [4,8]]

The sales list represents all of the sales made in a shop in 1 day. It is composed of sublists.

The values in each sublist indicate the price of a product and the quantity of the product that was sold. For example, [10,2] indicates that 10 units of a product priced at £2 were sold.

(a)  Shade **one** lozenge to indicate how many of the four functions (fw, fx, fy, fz) in the figure above use a higher-order function.



**(1)**

(b)  Calculate the results of making the function calls listed in the table below, using the functions and lists in the figure above as appropriate.

|  |  |
| --- | --- |
| **Function call** | **Result** |
| fw [4,3] |  |
| fx sales |  |
| fz sales |  |

**(3)**

(c)  In the context of the shop, explain what the result of the function call fz sales represents.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

**(Total 5 marks)**

**Q3.**

In a functional programming language a function named square and three lists a, b and c are defined as follows.

square x = x \* x

a = [1, 3, 5]

b = [1, 5, 10, 15]

c = [9, 7, 2]

(a)     What is the list or value that is the result of applying the functions head(tail(tail b))?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

(b)     Calculate the results of making the function calls listed in **Table 1** with the lists a, b and c above.

**Table 1**

|  |  |
| --- | --- |
| **Function Call** | **Result** |
| map square a |  |
| filter (<10) b |  |
| fold (+) 0 c |  |

**(3)**

(c)     map is an example of a higher-order function.

Explain what a higher-order function is.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

**(Total 5 marks)**

**Q4.**

In a functional programming language, a recursively defined function named map and a function named double are defined as follows:

map  f  []      =  []  
map  f  (x:xs)  =  f  x  :  map  f  xs

double  x       =  2  \*  x

The function x has two parameters, a function f, and a list that is either empty (indicated as []), or non-empty, in which case it is expressed as (x:xs) in which x is the head and xs is the tail, which is itself a list.

(a)     In **Table 1**, write the value(s) that are the head and tail of the list  
[ 1, 2, 3, 4 ].

**Table 1**

|  |  |
| --- | --- |
| Head |  |
| Tail |  |

(b)     The result of making the function call double 3 is 6.

**(1)**

Calculate the result of making the function call listed in **Table 2**.

**Table 2**

|  |  |
| --- | --- |
| **Function Call** | **Result** |
| map  double  [  1,  2,  3,  4  ] |  |

**(1)**

(c)     Explain how you arrived at your answer to part (**b**) and the recursive steps that you followed.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(3)**

**(Total 5 marks)**

**Q5.**

(a)     Put **one** tick on each row of the table below to classify each of (i), (ii) and (iii) as either a URL, a Domain Name, an IP address or a Protocol.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  | **URL** | **Domain Name** | **IP Address** | **Protocol** |
| (i) | http://www.guineas.co.uk |  |  |  |  |
| (ii) | 212.58.251.195 |  |  |  |  |
| (iii) | guineas.co.uk |  |  |  |  |

**(3)**

(b)     What is the purpose of a Domain Name Server on the Internet?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**(1)**

**(Total 4 marks)**

Mark schemes

**Q1.**

(a)  **Mark is AO2 (analysis)**

(The set of) integer (numbers) // ℤ // Z // the same set as the domain;

**1**

(b)  **Mark is AO2 (apply)**

4;

**1**

(c)  **2 marks AO1 (understanding) and 1 mark AO2 (analysis)**

**2 AO1 (understanding) marks for describing how partial application works:**

The function/add is applied to one of its arguments/4/6;

**A.** one of the arguments is fixed/bound

**A.** the new function takes one less argument // the new function takes one argument

The output of this function application is a new function;

**A.** a new function is created

**1 AO2 (analysis) mark for working out what the output of the partial application would be in this instance:**

The new function always adds the argument that the original function was applied to/4/6 to one argument; **A.** award this mark if it is clear what the new function does from its name eg “add4”.

**Note**: If a specific value (4 or 6) is used this must be the same value (4 or 6) for which the earlier mark point was awarded, if it was.

Accept answers given by example, mapping the relevant part of the example to the mark scheme points.

**Example of an answer by example**

add6(x)

**2 AO1 marks** for:

•   output of the partial application is a new function (eg add6)

•   the new function only has one argument(eg x)

where add6(x) = 6 + x

**1 AO2 mark** for explaining what new function does.

**Max 2** for an answer by example if there is no description.

*Answers can be a mix of marks for description and example*.

**3**

**[5]**

**Q2.**

(a)  **Mark is AO1 (understanding)**

2;

**R.** If more than one lozenge shaded

**1**

(b)  **All marks AO2 (apply)**

**One mark** per correct row in the **Result** column:

|  |  |
| --- | --- |
| **Function call** | **Result** |
| fw [4,3] | 12; |
| fx sales | [20, 50, 32];  **A.** alternative styles of bracket  **R.** no brackets  **R.** each element in a separate list |
| fz sales | 102; |

**3**

(c)  **Mark is AO2 (analyse)**

Total / one day’s sales value / income / revenue (for all products);

**A.** total / one day’s profit as BOD

**NE.** sales, total sales

**1**

**[5]**

**Q3.**

(a)  **Marks is for AO2 (apply)**

10;

**A**. [10] this time

**1**

(b)  **Mark is for AO2 (apply)**

|  |  |
| --- | --- |
| **Function Call** | **Result** |
| map square a | [1,9,25] |
| filter (<10) b | [1,5] |
| fold (+) 0 c | 18 |

**1 mark** for each correct response in the **Result** column.

**I**. Missing brackets this time or use of incorrect type of brackets

**I**. If returned values are assigned to new lists eg x = [1,9,25]

**A**. [5,1] for row 2 this time

**3**

(c)  **Mark is for AO1 (knowledge)**

A function that takes a function as an argument // returns a function as a result // takes a function as an argument and returns a function as a result;

**A**. “Parameter”, “Input” for “Argument”

**NE**. A function that uses another function

**R**. Explanations that are specifically of the map function

**1**

**[5]**

**Q4.**

(a)     **Marks is for AO1 (understanding)**

|  |  |
| --- | --- |
| Head | 1 |
| Tall | [2,3,4] |

**1 mark** for both head and tail correct.  
**I** if brackets are missing in tail.

**1**

(b)     **Mark is for AO2 (apply)**

[ 2, 4, 6, 8 ];

**I** if brackets are missing in tail.

**1**

(c)     **All marks AO1 (understanding)**

**1 mark:** Explaining that map applies the function double to each list element;

**1 mark:** Explaining that map applies double to the head of the list;

**1 mark:** and then a recursive call is made on the tail of the list;

**3**

**[5]**

**Q5.**

﻿

(a)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **URL** | **Domain Name** | **IP Address** | **Protocol** |
| (i) http://www.guineas.co.uk | ✔ |  |  |  |
| (ii) 212.58.251.195 |  |  | ✔ |  |
| (iii) guineas.co.uk |  | ✔ |  |  |

*1 mark for each correctly placed tick***R** Answers with more than one tick on a row.

**3**

(b)     To translate/convert/resolve domain names into IP addresses;  
**A** FQDN for domain name  
Answer must have the CONCEPT of an action  
**NE** To store the domain names and IP Addresses  
**NE** To access the web page without knowing the IP address  
**NE** To link the domain name to the IP address

**1**

**[4]**