

Definitions and Concepts for AQA Computer Science A-level

Topic 12: Fundamentals of Functional Programming

12.1 Functional Programming Paradigm

12.1.1 Function Type

Co-domain: A set from which the function's output values are chosen. †

Domain: A set from which the function's input values are chosen. †

Functions: A rule that, for each element in some set A of inputs, assigns an output chosen from set B, but without necessarily using every member of B. †

12.1.2 First-Class Object

First-Class Objects: First-class objects (or values) are objects which may appear in expressions, be assigned to a variable, be assigned as arguments or be returned in function calls. †

12.1.3 Function Application

Function Application: The process of giving particular inputs to a function. †

12.1.4 Partial Function Application

Partial Function Application: A function application scheme where a function is decomposed into intermediate partial functions. A partial function takes some of the arguments and returns a function awaiting the remaining arguments.

12.1.5 Composition of Functions

Composition of Functions: The combination of two functions to produce a new function that performs the two functions in a specified order.

12.2 Writing Functional Programs

12.2.1 Functional Language Programs

Filter: A higher-order function that processes a data structure, typically a list, in some order to produce a new data structure containing exactly those elements of the original data structure that match a given condition. †

This work by [PMT Education](https://www.pmt.education) is licensed under [CC BY-NC-ND 4.0](https://creativecommons.org/licenses/by-nc-nd/4.0/)



Functional Programming: A programming paradigm where programs are constructed through the application and composition of functions.

Higher Order Functions: Functions which take a function as an argument or return a function as a result, or both. †

Map: A higher-order function that applies a given function to each element of a list, returning a list of results. †

Reduce/Fold: A higher-order function which reduces a list of values to a single value by repeatedly applying a combining function to the list values. †

12.3 Lists in Functional Programming

12.3.1 List Processing

Append: Add an item to the end of a list.

Head-Tail Representation: A method of representing a list in two parts: a head which is the first element, and a tail which is itself a list of all the other elements.

Prepend: Add an item to the beginning of a list.

Definitions with a '†' taken from [AQA AS and A-level Computer Science specification version 1.5](#)

