Learning Programme

Fundamentals of functional programming – A Level

Topic/Content	Objectives/Skills	Homework	Assessment	Stretch & Challenge (Thirst for Learning)
Functional programming paradigm			Q and A's Programming with a worksheet for evidence Past exam paper questions Worksheets End of topic test	Take functional programming by researching and programming other tasks in own time
Function type	• Know that a function, f, has a function type • f: A → B (where the type is A → B, A is the argument type, and B is the result type). • Know that A is called the domain and B is called the co-domain. • Know that the domain and co-domain are always subsets of objects in some data type.	Craig and Dave video on functional programming – flipped learning		
First-class object	• Know that a function is a first-class object in functional programming languages and in imperative programming languages that support such objects. This means that it can be an argument to another function as well as the result of a function call.			
Function application	• Know that function application means a function applied to its arguments.	Craig and Dave video on functional programming – flipped learning		
Partial function application	• Know what is meant by partial function application for one, two and three argument functions and be able to use the notations shown opposite.			

Composition of functions	- Know what is meant by composition of functions.			
Functional language programs	- Show experience of constructing simple programs in a functional programming language. - Higher-order functions. - Have experience of using the following in a functional programming language: ○ map ○ filter ○ reduce or fold.	Worksheet		
List processing	- Be familiar with representing a list as a concatenation of a head and a tail. - Know that the head is an element of a list and the tail is a list. - Know that a list can be empty. - Describe and apply the following operations: ○ return head of list ○ return tail of list ○ test for empty list ○ return length of list ○ construct an empty list ○ prepend an item to a list ○ append an item to a list. - Have experience writing programs for the list operations mentioned above in a functional programming language or in a language with support for the functional paradigm.			