



a)

Fragment 1 - logic paradigm because it contains a series of statements and then a statement missing parts - the logic paradigm is to use the earlier statements to solve the last one

Fragment 2 - functional paradigm - this fragment contains functions - hence functional paradigm and <sup>it</sup> is missing the characteristics of the other paradigms.

- ⑥
- > Copes with much larger amounts of information unlike Basic or Cobol etc.
  - > Comes closer to plain speed unlike early generation languages
  - > deals with programs that respond to events as with a GUI
  - > stores modules of data "OBJECTS" that can be updated without updating the rest of the program
  - > like a "black box", an object can be unknown but still used by a program & programmer.



c) (i) In `Rectangle.height` has no initial value,  
~~so~~ it is not  $> 0$  or  $< 0$  so the  
program does not begin the loop and  
therefore does not function.

This could be solved by giving the  
`Rectangle.height` ~~a~~ a value before  
the loop begins or by making the  
loop a post-test rather than pre-  
test.

(ii) `height, base : integer`  
`function area : integer`

`function Triangle.area : integer`  
`begin`

`area := (height * base *  $\frac{1}{2}$ )`

`end`



D The programming paradigm which I would choose to develop the system below would be that of the Object Orientated Paradigm (oop). It would be suitable for this scenario for the following reasons:

- The system to be developed does not require complex mathematical problem as the Functional paradigm does.
- There is not an extensive list of concepts which need to be determined by the system thus there is no heavy processing required
- The system to be developed needs to be as user friendly as possible thus the emphasis on the GUI (Graphical User Interface)
- Programming of the system needs to be as simplistic as possible enabling programmers to readily identify & fix errors as they occur.
- The system will most likely be needed to be linked to a database detailing flight schedules. oop enables this to be done efficiently & effectively
- This approach could be used to design & develop the new system quickly on a small scale budget.