Q 24

a) Fragment 1 is using the logic paradigm because facts are stated. eg sally's hair is red and john's hair is brown.

Fragment 2 is using the ~~functional~~ imperitive paradigm. This section of code uses a contol structure, repetition in the form of a pre-test loop. It also uses a variable, 'x'. These are features of the imperitive paradigm.

b) There have been many reasons for the emergence of the object-oriented programming paradigm - These are:

1. recognition of repetilative programming tasks

Object oriented pogramming alloweds objects to carry out these tasks and are not needed to be coded from scratch.

2. Desire to solve different problems.

With the advent of event driven programming, object oriented programming has emerged because the way objects talk to each other is ~~an important~~ takes advantage of the fact that event driven programming relies on the user ~~to~~ determine what the program will do.

3. Recognition of basic building blocks.

Object ~~Oriented~~ Oreintated has become popular because it uses basic building blocks in it object so they are not needed to be created from scratch.

4. Greater productivity.

Object Oreintated langauges have become popular because it is very easy to learn them. This increases productivity.

c) i) The value of InstRectangle.height is not given a value before the loop condition therefore nothing happens. One method of correcting this error would be to set the rectangle height before the loop or to make the loop post-test so the rectangle actually has a height.

ii)

```
function TTriangle.area : integer;
begin
    area := (0.5 * base * height)
end;

begin
    while InstTriangle.height <> 0 do
```

```
begin
    write ('Enter height');
    readln( Inst triangle.height);
    ~~writln~~
    write ('Enter width');
    readln (Inst triangle.width);
    writeln ('The area is', Insttriangle.area);
end;
~~end~~
```

d) The programming paradigm I would use for ~~thus~~ this system would be logical. The system has to be able able to think for itself and decide whether or not to use 'Compression Rules', this would require an inference engine which is part of the logic paradigm. The system will determine where the bottlenecks may occur, this would involves heuristics to decide where it is most likely to happen.