Centre Number: .... Student Number: ....

a)(i)

Number Of Trains	Train	X	Y
0	1	Location x	Location Y
	2		
1	1	Not Processed	Not Processed

While Number Of Trains =0, the process should not proceed.

Number Of Trains	Train	X	Y
2	1	Loc X	Loc Y
	2		

(ii) When the number of trains = 0, the WHILE loop should not proceed, when in fact it does as shown by the desk check (1 <> 0). When their is one train the ~~progre~~ WHILE loop should proceed, but does not as the number of trains = Train. Finally, when there are 2 Trains, the desk check reveals that only one of the trains is processed, as the second time round, Train = 2 and Number Of Trains = 2.

BOARD OF STUDIES
NEW SOUTH WALES

Centre Number: .... Student Number: ....

(iii) I would modify line 4 to make it a post test loop rather than a pretest loop, but first I would insert a line above line 4 ~~sat~~ stating:

IF NumberOfTrains <> 0 THEN.

The final algorithm would look like this:

```
BEGIN DisplayTrain
    read NumberOfTrains
    Train = 1
    IF NumberOfTrains <> 0 THEN                                    1
        REPEAT                                                     2
                                                                   3
                                                                   4
            ReadTrainID (TrainID)
                                                                   3
            ReadLocation (TrainID, Location X, Location Y)
            DisplayTrainID (TrainID, Location X, Location Y)
            Train = Train + 1
        Until Train = (NumberOfTrains + 1)
    END IF
END
```

The above modifications ensure that the program will operate correctly.

b) i) The team member who downloads a run-time component has not checked for ~~manufacturer~~ the licence details for that piece of code, nor has he acknowledged it's source or author. This is plagiarism as due to his lack of acknowledgement the code appears as his own work. The code which is intellectual property of the manufacturer has automatic copyright and may have possibly been registered or patented. Plagiarism is illegal and the author could prosecute him on these grounds.

ii) To prevent the software team from be irresponsible, management could ask for documentation covering the whole development cycle to prove that their work is original. Their process diary would note any code borrowed from other sources and would fully acknowledge it including website, author, company, licencing information etc. Also, the versioning of the software would allow management to see how the project progressed and what strategies were used to develop the software, proving the authenticity of the program. Old versions could be decompressed to show the case history of the program. Regular project reviews could also be instated to show progress and any discre.pencies could be checked by management. Management could also run an ethics course or workshop reteaching the software team concepts such as plagiarism, bias, inclusivity and intellectual property.