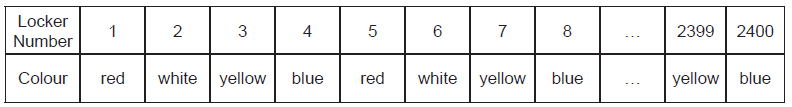
**SL Unit 4** **– Problem Solving**  
Test 1

1. In a school there are 2400 students and each student use one locker. Each locker has a unique

number from 1 to 2400.

The lockers are to be painted in four colours: red, white, yellow and blue, in order of locker

numbers, as shown in the following table.

The pattern of colours continues in this manner. For example, locker number 15 will be  
painted yellow.

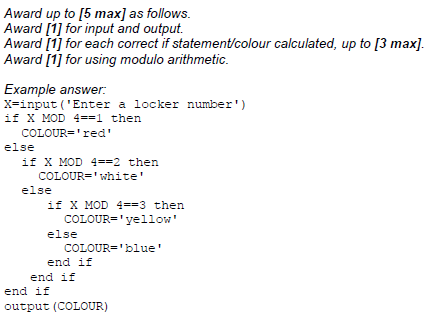
1. State the colour that locker number 442 will be painted.

White;

Each student is responsible for painting his or her locker. Some students do not know how  
to determine the colour they should use.

1. Construct, in pseudocode, an algorithm that accepts a locker number as input,

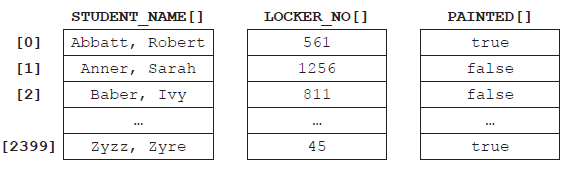
and outputs the colour that this locker should be painted.



Three arrays are used to hold the following information:

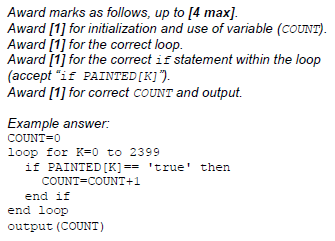
• the names of students in alphabetical order, are held in the array STUDENT\_NAME[];

• the corresponding locker number is held in the array LOCKER\_NO[];

• whether the locker has been painted or not is held in the array PAINTED[].

For example STUDENT\_NAME[1] is Sarah Anner. She is responsible for locker number 1256  
and this locker has not been painted yet.

1. (i) State the name of the student who is responsible for painting locker number 811. [1]  
     
     
     
   (ii) Construct, in pseudocode, an algorithm that counts and outputs the number of  
    lockers that have been painted so far. [4]



Baber, Ivy;

1. Describe an efficient algorithm, which accepts a student’s name as input, and outputs

the corresponding locker number and whether or not it has been painted. [4]

*Award* ***[1]*** *for stating that the position of a given name should be found in arraySTUDENT\_NAME[].*

*Award* ***[1]*** *for using this position to access the elements in the other*

*two arrays.*

*Award* ***[1]*** *for any named searching algorithm (linear or binary search).*

*Award* ***[1]*** *for explaining the efficiency – binary search used becauseSTUDENT\_NAME[] is sorted in alphabetical order.*