3a. Write a C program to read a number and check whether the entered number is a palindrome.

Description:

This program checks whether a given number is a palindrome. A palindrome is a number that remains the same when its digits are reversed.

Example:

Case 1: Palindrome Number

Input: Enter a number: 121
Output: 121 is a palindrome.

Case 2: Not a Palindrome

Input: Enter a number: 123
Output: 123 is not a palindrome.

Algorithm:

Step 1: Start

Step 2: Declare integer variables num, originalNum, reversedNum, and remainder.

Step 3: Prompt the user to enter a number.

Step 4: Read and store the number in num.

Step 5: Store the original number in originalNum for comparison later.

Step 6: Initialize reversedNum to 0.

Step 7: Reverse the number using a loop:
• While num is not 0:
• Extract the last digit using remainder = num % 10.
• Append the digit to reversedNum using reversedNum = reversedNum * 10 + remainder.
• Remove the last digit from num using num = num / 10.

Step 8: Compare originalNum with reversedNum:
• If they are equal, print "The number is a palindrome."
• Else, print "The number is not a palindrome."

Step 9: Stop

Source Code:

```
#include <stdio.h>
int main()
{
   int n, rev = 0, rem, num;

   printf("Enter an integer: ");
   scanf("%d", &n);

   num = n;

   /* reversed integer is stored in ‘rev’ variable */

   while (n != 0) {
      rem = n % 10;
      rev = rev * 10 +
      rem; n = n / 10;
   }

   /* It is a palindrome if original and reversed are equal

   */ if (num == rev) {
      printf("%d is a palindrome\n", num);
   }
   else {
      printf("%d is not a palindrome\n", num);
   }
   return 0;
}
```

Sample Output:

Enter an integer: 123321
123321 is a palindrome

Enter an integer: 1234
1234 is not a palindrome