

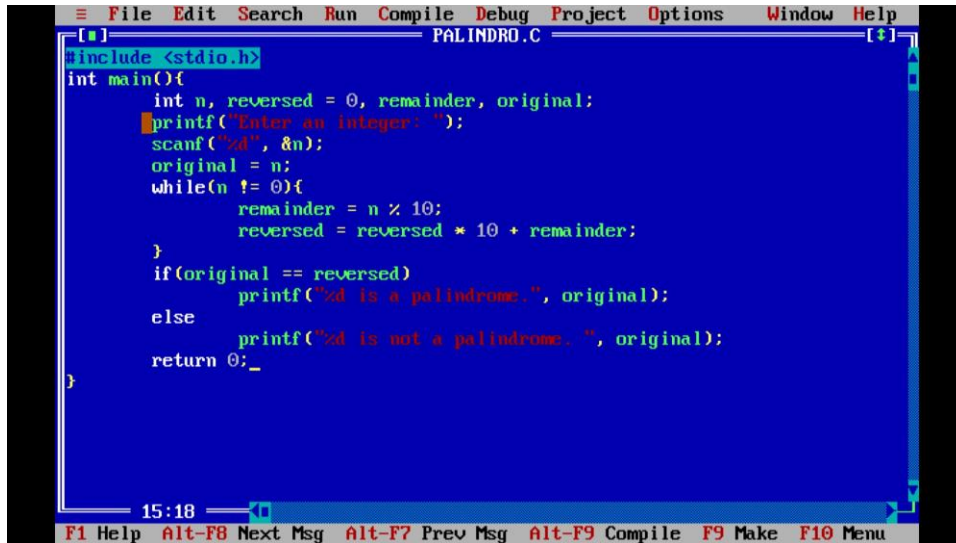
Name : Atharva R Karkar

Class : MCA DS

Roll no : 8

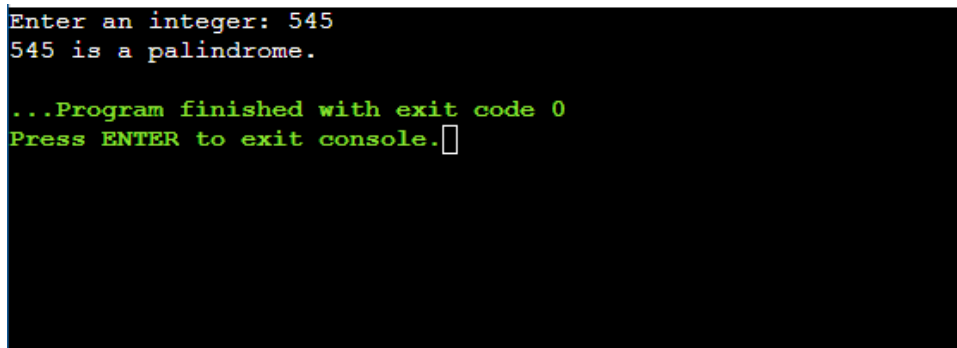
Assignment 5

1] C Program to check *number* is Palindrome or not.



```
File Edit Search Run Compile Debug Project Options Window Help
PALINDRO.C
#include <stdio.h>
int main(){
    int n, reversed = 0, remainder, original;
    printf("Enter an integer: ");
    scanf("%d", &n);
    original = n;
    while(n != 0){
        remainder = n % 10;
        reversed = reversed * 10 + remainder;
    }
    if(original == reversed)
        printf("%d is a palindrome.", original);
    else
        printf("%d is not a palindrome.", original);
    return 0;_
}
15:18
F1 Help Alt-F8 Next Msg Alt-F7 Prev Msg Alt-F9 Compile F9 Make F10 Menu
```

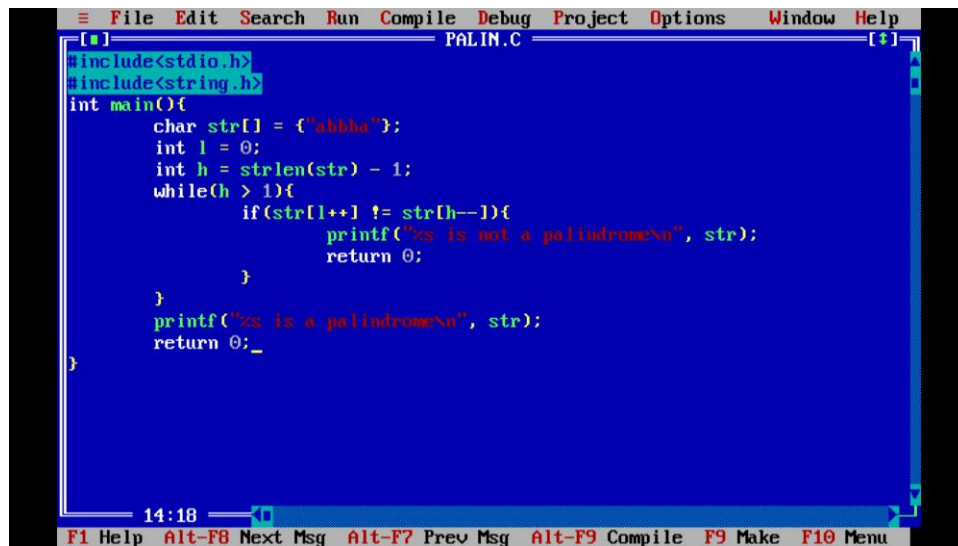
Output:



```
Enter an integer: 545
545 is a palindrome.

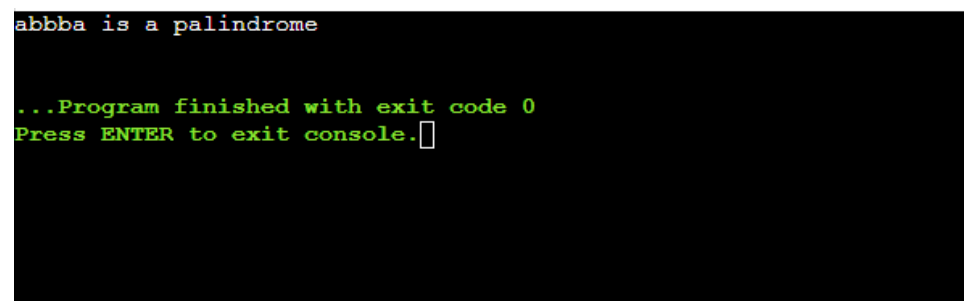
...Program finished with exit code 0
Press ENTER to exit console.
```

2] C Program to check String is Palindrome or not.



```
#include<stdio.h>
#include<string.h>
int main(){
    char str[] = {"abbba"};
    int l = 0;
    int h = strlen(str) - 1;
    while(h > l){
        if(str[l++] != str[h--]){
            printf("is not a palindrome\n", str);
            return 0;
        }
    }
    printf("is a palindrome\n", str);
    return 0;_
}
```

Output :



```
abbba is a palindrome

...Program finished with exit code 0
Press ENTER to exit console.█
```

3] C Program to Check *String* is Palindrome using Stack.

```
File Edit Search Run Compile Debug Project Options Window Help
STACK.C
#include <malloc.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
char* stack;
int top = -1;
void push(char ele){
    stack[++top] = ele;
}
char pop(){
    return stack[top--];
}
int isPalindrome(char str[]){
    int length = strlen(str);
    stack = (char*)malloc(length * sizeof(char));
    int i, mid = length / 2;
    for(i = 0; i < mid; i++){
        push(str[i]);
    }
    if(length % 2 != 0){
        i++;
    }
    1:1
```

```
File Edit Search Run Compile Debug Project Options Window Help
STACK.C
    int length = strlen(str);
    stack = (char*)malloc(length * sizeof(char));
    int i, mid = length / 2;
    for(i = 0; i < mid; i++){
        push(str[i]);
    }
    if(length % 2 != 0){
        i++;
    }
    while(str[i] != '\0'){
        char ele = pop();
        if(ele != str[i])
            return 0;
        i++;
    }
    return 1;
}
int main(){
    char str[] = "madam";
    if(isPalindrome(str)){
        printf("Yes");
    }
    1:1
```

```
File Edit Search Run Compile Debug Project Options Window Help
STACK.C
    while(str[i] != '\0'){
        char ele = pop();
        if(ele != str[i])
            return 0;
        i++;
    }
    return 1;
}
int main(){
    char str[] = "madam";
    if(isPalindrome(str)){
        printf("Yes");
    }
    else{
        printf("No");
    }
    return 0;
}
1:1
```

Output:

```
Yes  
...Program finished with exit code 0  
Press ENTER to exit console. 
```