NAME: - Jaydeep Solanki

ROLL NO: - 22ECG060 | 22BEC059

COURSE CODE: - 1CS501

SUBJECT: - COMPUTER PROGRAMMING

PRACTICAL NO 4: C programs to demonstrate use of loop constructs

---

a)

To s following patterns:

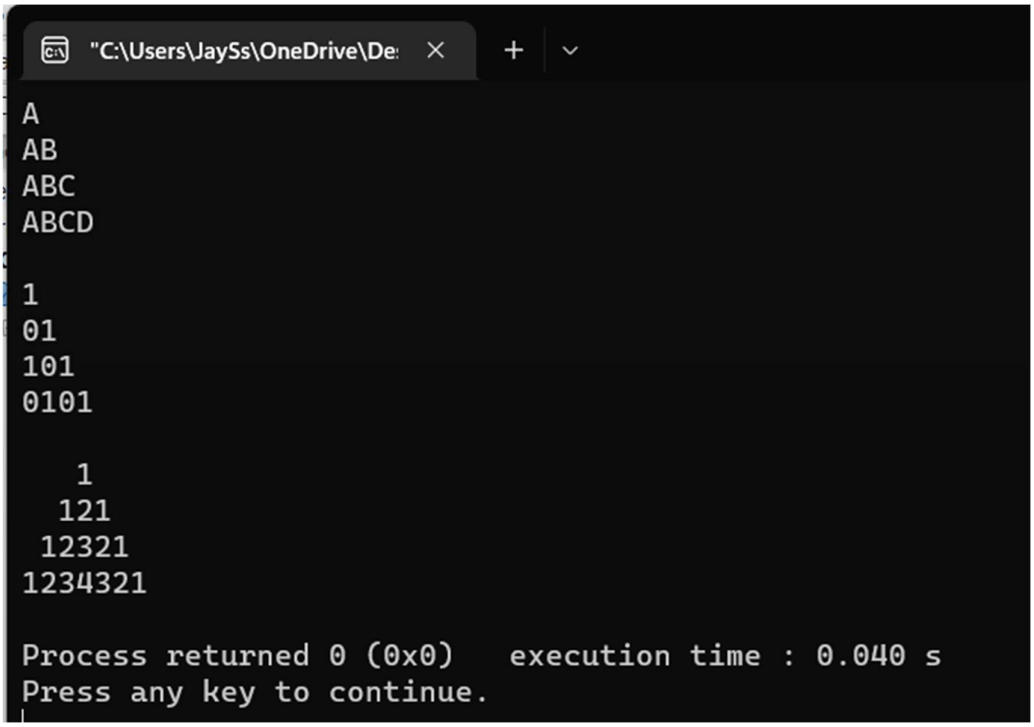
A A B A B C A B C D	1 01 101 0101	1 1 2 1 1 2 3 2 1 1 23432 1

Code :

```
#include <stdio.h>
int main() {
 for (int i = 0; i < 4; i++) {
  char a = 'A';
  for (int j = 0; j <= i; j++) {
   printf("%c", a);
   a++;
  }
  printf("\n");
 }
 printf("\n");

 for (int i = 0; i < 4; i++) {
  int sum;
  for (int j = 0; j <= i; j++) {
   sum = i + j;

   if (sum % 2 == 0) {
    printf("1");
   } else {
    printf("0");
   }
  }
  printf("\n");
 }
```

```
  printf("\n");


  for (int i = 1; i < 5; i++) {
    int n = 5;
    for (int k = 1; k < n - i; k++) {
      printf(" ");
    }
    for (int j = 1; j <= i; j++) {
      printf("%d", j);
    }
    for (int l = i - 1; l > 0; l--) {
      printf("%d", l);
    }
    printf("\n");
  }
  return 0;
}
```

Output:

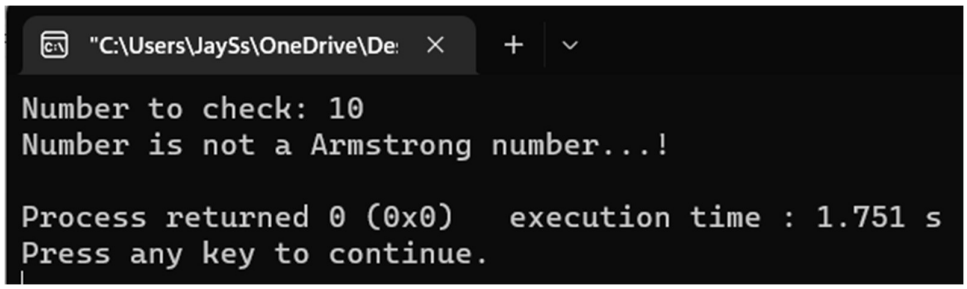


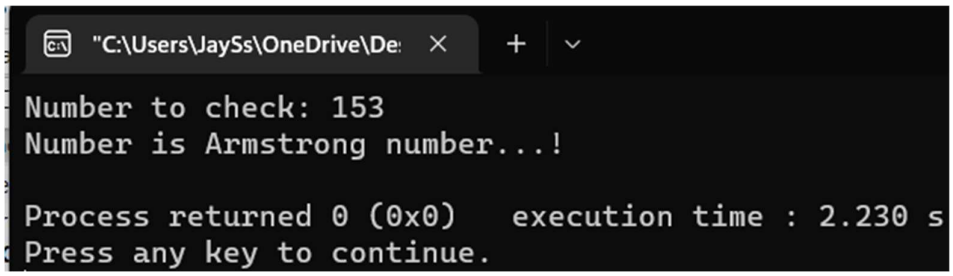
b) To determine whether the input number is an Armstrong number

Code:

```
#include<stdio.h>
int main()
{
   int num1,num2=0,num3,num4;
   printf("Number to check: ");
   scanf("%d",&num1);
   num3=num1;
   while(num1>0)
   {
      num4=num1%10;
      num2+=(num4*num4*num4);
      num1=num1/10;
   }
   if(num3==num2)
   {
      printf("Number is Armstrong number...!\n");
   }
   else
   {
      printf("Number is not a Armstrong number...!\n ");
   }
   return 0;
}
```

Output:



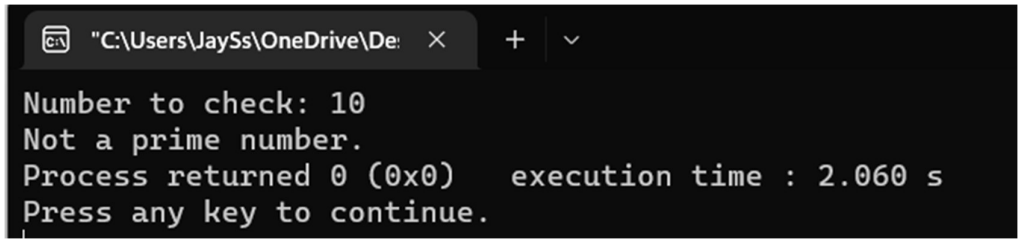


c) To determine whether the entered number is Prime

Code:

```
#include <stdio.h>
int main() {
 int number_to_ck, num2, temp_num = 0;
 printf("Number to check: ");
 scanf("%d", &number_to_ck);
 num2 = number_to_ck / 2;

 for (int i = 2; i <= num2; i++) {
  if (number_to_ck % i == 0) {
   printf("Not a prime number.");
   temp_num = 1;
   break;
  }
 }
 if (temp_num == 0) {
  printf("Number is prime.");
 }
 return 0;
}
```

Output:

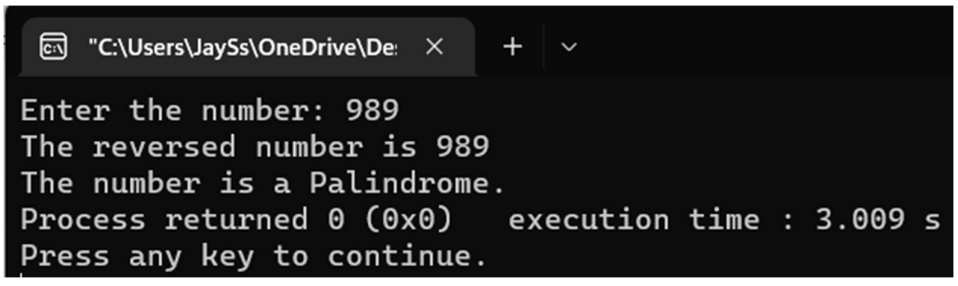


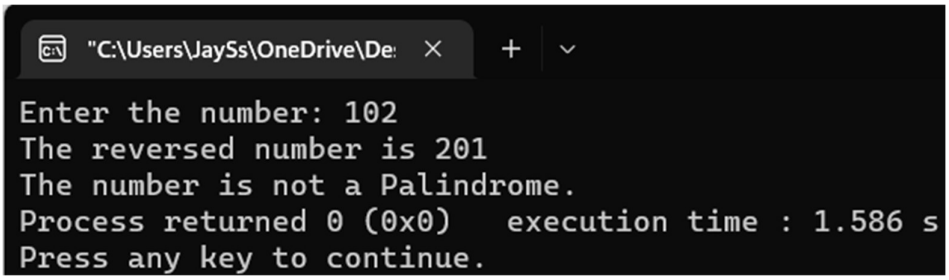
d) To determine whether the entered number is Palindrome.

Code:

```
#include <stdio.h>


int main() {
  int number, rem, original_number, reverse = 0;
  printf("Enter the number: ");
  scanf("%d", &number);
  original_number = number;
  while (number != 0) {
    rem = number % 10;
    reverse = reverse * 10+rem;
    number /= 10;
  }
  printf("The reversed number is %d\n", reverse);
  if (original_number == reverse) {
    printf("The number is a Palindrome.");
  } else {
    printf("The number is not a Palindrome.");
  }
  return 0;
}
```

Output:





e) Enhance the number guessing game developed earlier. The program should now display more appropriate message (Greater, Smaller or Correct). It should allow maximum 5 attempts from the user and still if the user cannot guess the number correctly, it should display “Sorry”.

Code:

```
#include <math.h>
#include <stdio.h>

int main() {
 int guess, tries = 0;
 srand(time(NULL));
 int num = rand() % 100;
 printf("====Number Guessing Game====\n\n");

 for (int i = 1; i <= 5; i++) {
  printf("Enter a guess number between 1 to 100 : ");
  scanf("%d", &guess);
  tries++;

  if (guess > num) {
   printf("Please guess smaller number.\n\n");
  } else if (guess < num) {
   printf("Please guess greater number.\n\n");
  } else {
   printf("\nCongratulations!\nYou got it in %d guesses,\n", tries);
  }
 }
 if (guess != num) {
  printf("Sorry..!\nBetter luck next time..!");
  printf("The number was : %d\n", num);
```

```
    }


    return 0;

}
```

Output: