Experiment no – 02(a)

Aim: Write a program using while loop to reverse the digits of a number.

Algorithm:

i. Ask the user to enter any number.

ii. Declare and initialize another variable reversed with 0, where reversed an integer variable.

iii. Get the last digit of the given number by performing the modulo division (%) and store the value in last_digit variable, likey last_digit= number % 10.

iv. Multiply reversed by 10 and add last_digit, like reversed = reversed*10 + last_digit.

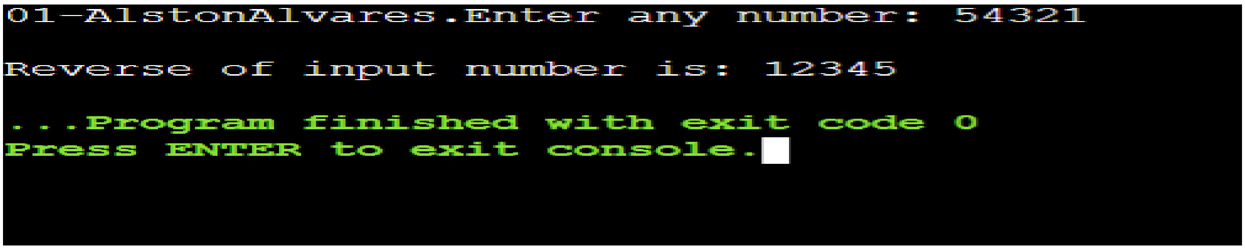
v. Divide numbered by 10, like numbered/10.

vi. Repeat the steps 3 to 5 till numbered is not equal to (or greater than) zero.

Code:

```
#include <stdio.h>
int main()
{ printf("01-AlstonAlvares.");
   int num, rnum = 0, rem;
   printf("Enter any number: ");
   scanf("%d", &num);
   while (num != 0) {
      rem = num % 10;
      rnum = rnum * 10 + rem;
      num = num / 10;
   }
   printf("\nReverse of input number is: %d", rnum);
   return 0;
}
```

Output:



```
01-AlstonAlvares.Enter any number: 54321

Reverse of input number is: 12345

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

Experiment no – 02(b)

Aim: Write a program to calculate the factorial of a given number.

Algorithm:

i. Start program

ii. Ask the user to enter an integer to find the factorial

iii. Read the integer and assign it to a variable

iv. From the value of the integer up to 1, multiply each digit and update the final value

v. The final value at the end of all the multiplication till 1 is the factorial

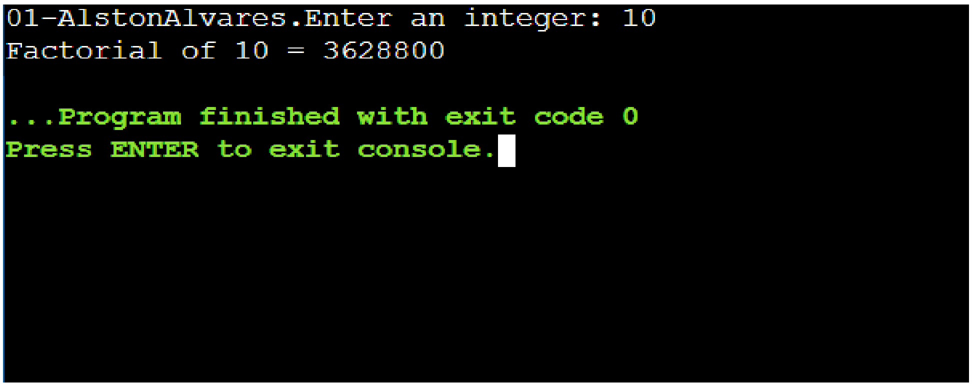
vi. End program

Code:

```
#include <stdio.h>
int main() {
  {
   printf("01-AlstonAlvares.");}
   int n, i;
   unsigned long long fact = 1;
   printf("Enter an integer: ");
   scanf("%d", &n);
   // shows error if the user enters a negative integer
   if (n < 0)
       printf("Error! Factorial of a negative number doesn't exist.");
   else {
       for (i = 1; i <= n; ++i) {
           fact *= i;
       }
       printf("Factorial of %d = %llu", n, fact);
   }
```

```
    return 0;
}
```

Output:



```
01-AlstonAlvares.Enter an integer: 10
Factorial of 10 = 3628800

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

Experiment no – 02(c)

Aim: Write a program to find the roots of quadratic equation.

Algorithm:

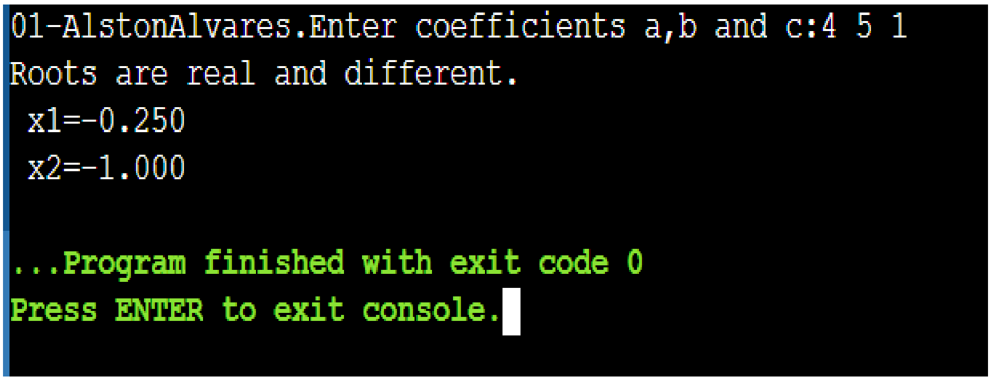
```
i.    Start
ii.   Read a, b, c values
iii.  Compute d = b2 4ac
iv.   if d > 0 then
          i.   r1 = b+ sqrt (d)/(2*a)
          ii.  r2 = b sqrt(d)/(2*a)
v.    Otherwise if d = 0 then
          i.   compute r1 = -b/2a, r2=-b/2a
          ii.  print r1,r2 values
vi.   Otherwise if d < 0 then print roots are imaginary
vii.  Stop
```

Code:

```
#include<stdio.h>

#include<math.h>
```

```
int main()
{
   printf("01-AlstonAlvares.");
    float a,b,c,x1,x2,determinant,realpart,imaginaryPart;
    printf("Enter coefficients a,b and c:");
    scanf("%f%f%f",&a,&b,&c);
    determinant=b*b - 4*a*c;
    if (determinant>0)
{
      x1=(-b + sqrt(determinant))/(2*a);
      x2=(-b - sqrt(determinant))/(2*a);
printf("Roots are real and different.");
printf("\n x1=%.3f",x1);
printf("\n x2=%.3f",x2);
}
else if (determinant==0)
{
printf("Roots are real and same.");
x1=(-b+sqrt(determinant))/(2*a);
printf("\n x1=%.ef",x1);
printf("\nx2=%.3f",x2);
}
Else
{
realpart=-b/(2*a);
imaginaryPart=sqrt(determinant)/(2*a);
printf("\n Roots are complex and differtent.");
printf("\n x1=%.3f+%.fi",realpart,imaginaryPart);
printf("\nx2 = %.3f-%3fi",realpart,imaginaryPart);
}
return 0;

}
```

Output:



```
01-AlstonAlvares.Enter coefficients a,b and c:4 5 1
Roots are real and different.
 x1=-0.250
 x2=-1.000

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

Experiment no – 02(d)

Aim: Write a program to print the Fibonacci series.

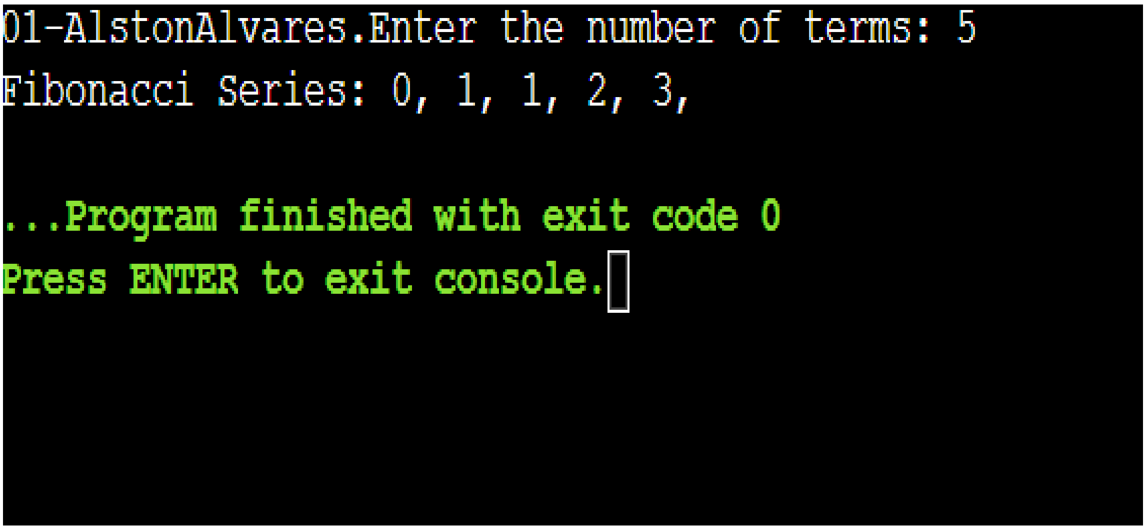
Algorithm:

```
i.     START
ii.    Take integer variable A, B, C
iii.   Set A = 0, B = 0
iv.    DISPLAY A, B
v.     C = A + B
vi.    DISPLAY C
vii.   Set A = B, B = C
viii.  REPEAT from 4 - 6, for n times
ix.    STOP
```

Code:

```
#include <stdio.h>
int main() {
printf("01-AlstonAlvares.");
  int i, n;

  // initialize first and second terms
  int t1 = 0, t2 = 1;

  // initialize the next term (3rd term)
  int nextTerm = t1 + t2;

  // get no. of terms from user
  printf("Enter the number of terms: ");
  scanf("%d", &n);

  // print the first two terms t1 and t2
  printf("Fibonacci Series: %d, %d, ", t1, t2);

  // print 3rd to nth terms
  for (i = 3; i <= n; ++i) {
    printf("%d, ", nextTerm);
    t1 = t2;
    t2 = nextTerm;
    nextTerm = t1 + t2;
  }
  return 0;
}
```

Output:



```
01-AlstonAlvares.Enter the number of terms: 5
Fibonacci Series: 0, 1, 1, 2, 3,

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