LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

```
//q1)Check whether an input integer is a perfect square or not.
#include<stdio.h>
int main(){
   int i, number;
   printf("Enter a number: ");
   scanf("%d", &number);
   for(i = 0; i <= number; i++){
      if(number == i*i){
         printf("%d is a perfect square\n", number);
         break;
      }
      else{
         printf("%d is not a perfect square\n", number);
         break;
      }
   }
   return 0;
}
```

Output-

```
Enter a number: 145
145 is not a perfect square
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

//q2)Input any integer and print your name that many times.

```
#include<stdio.h>
int main(){
   int a;
   printf("Enter a number: ");
   scanf("%d",&a);
   for(int i=0;i<a;i++){
      printf("Abhik Samanta\n");
   }
   return 0;
}
```

Output-

```
Enter a number: 7
Abhik Samanta
Abhik Samanta
Abhik Samanta
Abhik Samanta
Abhik Samanta
Abhik Samanta
Abhik Samanta
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

//q3)Print all odd and even numbers separately within a given range. The range is input through the user.

```
#include<stdio.h>
int main() {
   int a, b;
   printf("Enter the starting number: ");
   scanf("%d", &a);
   printf("Enter the ending number: ");
   scanf("%d", &b);
   printf("Even numbers: ");
   for (int i = a; i <= b; i++) {
      if (i % 2 == 0) {
         printf("%d ", i);
      }
   }
   printf("\n");
   printf("Odd numbers: ");
   for (int i = a; i <= b; i++) {
      if (i % 2 != 0) {
         printf("%d ", i);
      }
   }
   return 0;
}
```

Output-

Enter the starting number: 4

Enter the ending number: 50

Even numbers: 4 6 8 10 12 14 16 18 20 22 24 26 28 30 32 34 36 38 40 42 44 46 48 50

Odd numbers: 5 7 9 11 13 15 17 19 21 23 25 27 29 31 33 35 37 39 41 43 45 47 49

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

```
//q4)Print the multiplication table of an inputted number.
#include <stdio.h>
int main(){
   int num;
   printf("Enter a number: ");
   scanf("%d", &num);
   printf("Multiplication table of %d:\n", num);
   for(int i=1;i<=10;i++){
      printf("%d x %d = %d\n", num, i, num*i);
   }
   return 0;
}
```

Output-

```
Enter a number: 9
Multiplication table of 9:
9 x 1 = 9
9 x 2 = 18
9 x 3 = 27
9 x 4 = 36
9 x 5 = 45
9 x 6 = 54
9 x 7 = 63
9 x 8 = 72
9 x 9 = 81
9 x 10 = 90
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

```
//q5)Check whether an input integer is a strong number or not.
#include <stdio.h>
int main(){
   int num, temp, digit, fact, sum = 0;
   printf("Enter a number: ");
   scanf("%d", &num);
   temp = num;
   while(temp>0){
      digit=temp % 10;
      fact=1;
      for(int i = 1; i <= digit; i++) {
         fact *= i;
      }
      sum += fact;
      temp /= 10;
   }
   if(num==sum){
      printf("%d is a strong number.\n", num);
   }
   else{
      printf("%d is not a strong number.\n", num);
   }
   return 0;
}
```

Output-

```
Enter a number: 145
145 is a strong number.
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

```
//q6)Find out the prime factors of a number entered through keyboard
#include <stdio.h>
int main(){
   int num;
   printf("Enter a positive integer: ");
   scanf("%d", &num);
   printf("Prime factors of %d are: ", num);
   for (int i = 2; i <= num; i++){
      int count=0;
      while (num % i == 0){
         count++;
         num /= i;
      }
      if(count>0){
         printf("%d(%d) ",i,count);
      }
   }
   return 0;
}
```

Output-

```
Enter a positive integer: 1200
Prime factors of 1200 are: 2(4) 3(1) 5(2)
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

//q7)Find the numbers between 1 to 1000, which are divisible by the sum of its digits.

```
#include <stdio.h>
int main() {
   int num,sum,i;
   printf("Numbers between 1 to 1000 which are divisible by the sum of its digits:\n");
   for (num = 1; num <= 1000; num++) {
      sum = 0;
      i = num;
      while (i>0){
         sum += i%10;
         i/= 10;
      }
      if (num % sum == 0) {
         printf("%d\t", num);
      }
   }
   return 0;
}
```

Output-

Numbers between 1 to 1000 which are divisible by the sum of its digits:

```
1    2    3    4    5    6    7    8    9    10   12   18   20   21   24   27  30   36
40   42   45   48   50   54   60   63   70   72   80   81   84   90  100   102  108
110  111  112  114  117  120  126  132  133  135  140  144  150 152   153
156  162  171  180  190  192  195  198  200  201  204  207  209  210 216
220  222  224  225  228  230  234  240  243  247  252  261  264  266  270
280   285  288  300  306  308  312  315  320  322  324  330  333  336  342
351 360   364  370  372  375  378  392  396  399  400  402  405  407  408
410  414 420   423  432  440  441  444  448  450  460  465  468  476  480
481  486  500 504   506  510  511  512  513  516  518  522  531  540  550
552  555  558  576 588   592  594  600  603  605  612  621  624  629  630
640  644  645  648  660 666   684  690  700  702  704  711  715  720  730
732  735  736  738  756  770 774   777  780  782  792  800  801  803  804
810  820  825  828  832  840  846 864   870  874  880  882  888  900  902
910  912  915  918  935  936  954  960 966   972  990  999  1000
```

LAB 10
ABHINAV ANAND ROLL-22052611 SEC-B16

//q8)Print the numbers between10 to 1000 where the digits of the numbers are same.

```
#include <stdio.h>

int main() {
   int num;
   printf("Numbers between 10 to 1000 where the digits of the numbers are the same:\n");
   for (num = 10; num <= 1000; num++) {
      if ((num<=100 && num % 11 == 0) || (num>=100 && num % 111 == 0)) {
         printf("%d\t", num);
      }
   }
   return 0;
}
```

Output-

Numbers between 10 to 1000 where the digits of the numbers are the same:

11 22 33 44 55 66 77 88 99 111 222 333 444 555 666 777 888 999