Assignment - 10

1. Write a function to calculate the area of a circle. (TSRS)

```
#include<stdio.h>
float areaofcircle(float);
int main()
{
   float a,result;
   printf("Enter the radius = ");
   scanf("%f",&a);
   result = areaofcircle(a);
   printf("Area of Circle = %f",result);
   return 0;
}
float areaofcircle(float r)
{
   float res;
   res = 3.14*r*r;
   return res;
}
```

2. Write a function to calculate simple interest. (TSRS)

```
#include<stdio.h>

float simpleinterest(float,float,float);
int main()
{
   float result,a,b,c;
   printf("Enter the Principle Amount = ");
   scanf("%f",&a);
   printf("Enter the Rate = ");
   scanf("%f",&b);
```

```
    printf("Enter the Time = ");
    scanf("%f",&c);
    result = simpleinterest(a,b,c);
    printf("Simple Interest is = %f",result);
    return 0;
}
float simpleinterest(float p,float r,float t)
{
    float res;
    res = (p*r*t)/100;
    return res;
}
```

3. Write a function to check whether a given number is even or odd. Return 1 if the number is even, otherwise return 0. (TSRS)

```
        #include<stdio.h>
int even_odd(int);
int main()
{
    int a,res;
    printf("Enter the Number = ");
    scanf("%d",&a);
    res = even_odd(a);

    return 0;
}
int even_odd(int num)
{
    if(num%2==0)
        return 1;
    else
        return 0;

}
```

4. Write a function to print first N natural numbers (TSRN)

```
#include<stdio.h>

void naturalNumber(int);
int main()
{
   int n,res;
   printf("Enter the number = ");
   scanf("%d",&n);
   printf("Natural Number = ");
   naturalNumber(n);
   return 0;
}
void naturalNumber(int a)
{
   int i;
   for(i=1;i<=a;i++)
   {
      printf("%d ",i);
   }
}
```

5. Write a function to print first N odd natural numbers. (TSRN)

```
#include<stdio.h>

void naturalNumber(int);
int main()
{
   int n,res;
   printf("Enter the number = ");
   scanf("%d",&n);
   printf("Natural Number = ");
   naturalNumber(n);
```

```
    return 0;
}
void naturalNumber(int a)
{
    int i;
    for(i=1;i<=a;i++)
    {
       printf("%d ",2*i-1);
    }
}
```

6. Write a function to calculate the factorial of a number. (TSRS)

```
        #include<stdio.h>
int factorial(int);
int main()
{
  int a,res;
  printf("Enter the number = ");
  scanf("%d",&a);
  res = factorial(a);
  printf("Factorial = %d",res);
  return 0;
}
int factorial(int x)
{
    int i,fact=1;
    for(i=1;i<=x;i++)
    {
        fact = fact*i;
    }

    return fact;
}
```

7. Write a function to calculate the number of combinations one can make from n items and r selected at a time. (TSRS)

```
        #include<stdio.h>
int factorial(int);
int combination(int,int);
int main()
{
  int x,y,res;
  printf("Enter the number = ");
  scanf("%d %d",&x,&y);
  //res = factorial(a);
  res = combination(x,y);
  printf("combination = %d",res);
  return 0;
}
int factorial(int x)
{
    int i,fact=1;
    for(i=1;i<=x;i++)
    {
        fact = fact*i;
    }

    return fact;
}
int combination(int n,int r)
{
    return factorial(n)/(factorial(r)*factorial(n-r));
}
```

8. Write a function to calculate the number of arrangements one can make from n items and r selected at a time. (TSRS)

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

```
int factorial(int);
int permutation(int,int);
int main()
{
  int x,y,res;
  printf("Enter the number = ");
  scanf("%d %d",&x,&y);
  //res = factorial(a);
  res = permutation(x,y);
  printf("Permutation = %d",res);
  return 0;
}
int factorial(int x)
{
    int i,fact=1;
    for(i=1;i<=x;i++)
    {
        fact = fact*i;
    }

    return fact;
}
int permutation(int n,int r)
{
    return factorial(n)/factorial(n-r);
}
```

9. Write a function to check whether a given number contains a given digit or not. (TSRS)

```
        #include<stdio.h>
int checkdigit(int);
int main()
{
```

```
    int res,num,dig;
    printf("Enter the Number = ");
    scanf("%d",&num);
    printf("Enter the digit = ");
    scanf("%d",dig);
    res = checkdigit(num);
    printf("match = %d",res);
    return 0;
}
int checkdigit(int n)
{
    int r;
    r = n%10;
    n = n/10;

    if(r==dig)
    {
        return r;
    }


}
```

10. Write a function to print all prime factors of a given number. For example, if the number is 36 then your result should be 2, 2, 3, 3. (TSRN)

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


void primeFactors(int n)
{
    while (n%2 == 0)
    {
        printf("%d ", 2);
```

```
        n = n/2;
    }
    for (int i = 3; i <= sqrt(n); i = i+2)
    {
        while (n%i == 0)
        {
            printf("%d ", i);
            n = n/i;
        }
    }

    if (n > 2)
        printf ("%d ", n);
}

int main()
{
    int n = 36;
    primeFactors(n);
    return 0;
}
```