Assignment 10

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

```
#include<stdio.h>
float area_of_circle(float);
int main()
{
    float r,s;
    printf("Enter a redius: ");
    scanf("%f",&r);
    s = area_of_circle(r);
    printf("area of circle %f",s);
    return 0;
}
float area_of_circle(float redius)
{
    float pi = 3.14;
    return (pi*redius*redius);
}
```

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

```
#include<stdio.h>
float simple_interest(float,float,float);
int main()
{
    float p,r,t;
    printf("Enter principal, rate and time ");
    scanf("%f%f%f", &p,&r,&t);

    printf("Simple interest is ",simple_interest(p,r,t));
    return 0;
}
float simple_interest(float P,float R, float T)
{
    float SI = (P*R*T)/100;
    return SI;
}
```

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 n;
    printf("enter a number ");
    scanf("%d",&n);
```

```
    printf("%d",Even_odd(n));
    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 PrintNnum(int);
int main()
{
    int num;
    printf("enter a number ");
    scanf("%d",&num);
    PrintNnum(num);
    return 0;
}
void PrintNnum(int n)
{
    int i;
    printf("First %d natural numbers: ",n);
    for(i=1; i<=n; i++)
    {
        printf("%d ", i);
    }
}
```

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

```
#include<stdio.h>
void PrintN_odd(int);
int main()
{
    int num;
    printf("enter a number ");
    scanf("%d",&num);
    PrintN_odd(num);
    return 0;
}
void PrintN_odd(int n)
{
    int i;
    printf("First %d ood natural numbers: ",n);
    for(i=1; i<=n; i++)
    {
```

```
        printf("%d ", i*2-1);
    }
}
```

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

```
#include<stdio.h>
int factorial(int);
int main()
{
    int n,s;
    printf("Enter a number ");
    scanf("%d", &n);
    s=factorial(n);
    printf("factorial of %d is: %d",n,s);
    return 0;
}
int factorial(int num)
{
    int fact = 1,i;
    for ( i = 1; i <= num ; 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. (TS RS)

```
#include<stdio.h>
int combination(int,int);
int fact(int n);
int main()
{
    int N,R,s;
    printf("enter the items and r selected at a time ");
    scanf("%d%d",&N,&R);
    s = combination(N,R);
    printf("Number of combination = %d",s);
    return 0;
}
int combination(int num,int r)
{
    int c = fact(num)/(fact(r)*fact(num-r));
    return c;
}
int fact(int num)
{
    int f = 1,i;
    for ( i = 1; i <= num ; i++)
```

```
    {
        f = f*i;
    }
    return f;
}
```

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

```
#include<stdio.h>
int Permutation(int,int);
int fact(int n);
int main()
{
    int N,R,s;
    printf("enter the items and r selected at a time ");
    scanf("%d%d",&N,&R);
    s = Permutation(N,R);
    printf("number of arrangements = %d",s);
    return 0;
}
int Permutation(int num,int r)
{
    int p = fact(num)/fact(num-r);
    return p;
}
int fact(int num)
{
    int f = 1,i;
    for ( i = 1; i <= num ; i++)
    {
        f = f*i;
    }
    return f;
}
```

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

```
#include<stdio.h>
int ContainDigit(int);
int main()
{
    int d, s;
    printf("Enter a digit ");
    scanf("%d",&d);
    s = ContainDigit(d);
    if (s==1)
        printf("number contains a given digit");
    else
        printf("number not contains a given digit");
    return 0;
```

```
}
int ContainDigit(int digit)
{
    int num,rem,c=0;
    printf("Enter a number ");
    scanf("%d",&num);
    while (num!=0)
    {
        rem = num%10;
        num = num/10;
        if (rem==digit)
            c=1;
    }
    return c;

}
```

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>
void prime_factors(int);
int main()
{
    int n,i;
    printf("Enter a number ");
    scanf("%d",&n);
    prime_factors(n);
    return 0;
}
void prime_factors(int num)
{
    int i,j,c;
    for(i=1; i<=num; i++)
    {
        c=0;
        if (num%i==0)
        {
            for (j = 2; j < i; j++)
            {
                if (i%j==0)
                    c++;
            }
            if (c==0)
            {
                printf("%d ",i);
            }
        }
    }
}
```