Functions in C Language

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

```c
#include<stdio.h>
float circleArea(float);
int main()
{
    float radius, area;
    printf("Enter radius of a circle : ");
    scanf("%f", &radius);
    area = circleArea(radius);
    printf("Area of a circle %f unit", area);
    return 0;
}
// below function is to calculate area of a circle
float circleArea(float r)
{
    float pi = 3.14159, Area;
    Area = pi * r * r;
    return (Area);
}
```

```
================================================================================
Output:
Enter radius of a circle : 5.5
Area of a circle 95.033104 unit
```

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

```c
#include <stdio.h>
float simpleInterest(float, float, float);
int main()
{
    float p, r, t, si;
    printf("Calculate Simple Interest\n");
    printf("Enter amount, rate and time : ");
    scanf("%f%f%f", &p, &r, &t);
    si = simpleInterest(p, r, t);
    printf("Simple Interest : Rs. %.2f /-\n", si);
    printf("Total Amount : Rs. %.2f /-", (p + si));
    return 0;
}
// below function is to calculate simple interest
float simpleInterest(float a, float b, float c)
{
    float si;
    si = (a * b * c) / 100;
    return (si);
}
```

```
================================================================================
Output:
Calculate Simple Interest
Enter amount, rate and time : 5500 6.78 5
Simple Interest : Rs. 1864.50 /-
Total Amount : Rs. 7364.50 /-
```

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)

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

```
    int n;
    printf("Enter a number to check Even or Odd: ");
    scanf("%d", &n);
    evenOdd(n) ? printf("Even number") : printf("Odd number");
    return 0;
}
// below function is to calculate even-odd number
int evenOdd(int x)
{
    if(x % 2 == 0)
        return(1);
    else
        return(0);
}
================================================================================
Output:
Enter a number to check Even or Odd: 111
Odd number
```

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

```
#include <stdio.h>
void naturalNumbers(int);
int main()
{
    int n;
    printf("Enter a number : ");
    scanf("%d", &n);
    naturalNumbers(n);
    return 0;
}
// below function is to print n natural numbers
void naturalNumbers(int x)
{
    for (int i = 1; i <= x; i++)
    {
        printf("%d ", i);
    }
}
================================================================================
Output:
Enter a number : 15
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15
```

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

```
#include <stdio.h>
void oddNatural(int);
int main()
{
    int n;
    printf("Enter a number : ");
    scanf("%d", &n);
    oddNatural(n);
    return 0;
}
// below function is to print odd natural numbers
void oddNatural(int x)
{
    int i = 1;
    while (i <= x)
    {
        printf("%d ", (i * 2) - 1);
        i++;
    }
```

```
}
================================================================================
Output:
Enter a number : 9
1 3 5 7 9 11 13 15 17
```

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

```c
#include <stdio.h>
unsigned long long fact(int);
int main()
{
    int n;
    unsigned long long factorial;
    printf("Enter a number to calculate factorial : ");
    scanf("%d", &n);
    factorial = fact(n);
    printf("Factorial of %d is %llu", n, factorial);
    return 0;
}
// below function is to calculate factorial of a number
unsigned long long fact(int x)
{
    int i = 1;
    unsigned long long s = 1;
    while (i <= x)
    {
        s = s * i;
        i++;
    }
    return (s);
}
```

```
================================================================================
Output:
Enter a number to calculate factorial : 15
Factorial of 15 is 1307674368000
```

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

```c
#include <stdio.h>
unsigned long long fact(int);
unsigned long long combination(int, int);
int main()
{
    int n, r;
    unsigned long long nCr;
    printf("Enter the value of N and R respectively : ");
    scanf("%d%d", &n, &r);
    nCr = combination(n, r);
    printf("The number of combinations = %llu", nCr);
    return 0;
}
// below function is to calculate factorial of a number
unsigned long long fact(int x)
{
    int i = 1;
    unsigned long long s = 1;
    while (i <= x)
    {
        s = s * i;
        i++;
    }
    return (s);
}
```

```
// below function is to calculate combination
unsigned long long combination(int x, int y)
{
    return (fact(x) / (fact(x - y) * fact(y)));
}
================================================================================
Output:
Enter the value of N and R respectively : 20 8
The number of combinations = 125970
```

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>
unsigned long long fact(int);
unsigned long long arrangements(int, int);
int main()
{
    int n, r;
    unsigned long long nPr;
    printf("Enter the value of N and R respectively : ");
    scanf("%d%d", &n, &r);
    nPr = arrangements(n, r);
    printf("The number of arrangements = %llu", nPr);
    return 0;
}
// below function is to calculate factorial of a number
unsigned long long fact(int x)
{
    int i = 1;
    unsigned long long int s = 1;
    while (i <= x)
    {
        s = s * i;
        i++;
    }
    return (s);
}
// below function is to calculate permutations
unsigned long long arrangements(int x, int y)
{
    return(fact(x) / fact(x - y));
}
================================================================================
Output:
Enter the value of N and R respectively : 20 6
The number of arrangements = 27907200
```

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

```
#include<stdio.h>
int digitContains(int, int);
int main()
{
    int number, digit, result;
    printf("Enter a number : ");
    scanf("%d", &number);
    printf("Enter a digit to check given number contains digit or not : ");
    scanf("%d", &digit);
    result = digitContains(number,digit);
    result == 1 ? printf("Number contains a given digit") : printf("Number
does not contains a given digit");
    return 0;
}
```

```
//below function is to check whether a given number contains a given digit or
not
int digitContains(int x, int y)
{
    int i, remainder;
    for ( i = 1; 1; i++)
    {
        remainder = x%10;
        x = x / 10;
        if (remainder == y)
            return (1);
        if (x == 0)
            return (0);
    }
}
```

```
Output:
Enter a number : 12345
Enter a digit to check given number contains digit or not : 6
Number does not contains a given digit
```

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 primeFactor(int);
int main()
{
    int n;
    printf("Enter a number : ");
    scanf("%d", &n);
    primeFactor(n);
    return 0;
}
//below function is to calculate prime factors
void primeFactor(int x)
{
    int i, j;
    for (i = 2; i <= x; i++)
    {
        for (j = 2; j < i; j++)
        {
            if (i % j == 0)
            {
                break;
            }
        } // end inner loop
        if (i == j)
        {
            if (x % i == 0)
            {
                x = x / i;
                printf("%d ", i);
                i = 1;
            }
        }
    } // end outer loop
}
```

```
Output:
Enter a number : 36
2 2 3 3
```