**Q1. Write a function to calculate the area of a circle. (TSRS)**

float areaOfCircle(float);

#include <stdio.h>

int main(int argc, char \*argv[])

{

    float radius;

    printf("Enter radius = ");

    scanf("%f", &radius);

    printf("Area of circle = %.2f", areaOfCircle(radius));

    return 0;

}

float areaOfCircle(float radius)

{

    return (3.14 \* (radius \* radius));

}

**Q2. Write a function to calculate simple interest. (TSRS)**

#include <stdio.h>

float simpleInterest(float, float, float);

int main(int argc, char \*argv[])

{

    float principal, rateofinterest, timeperoid;

    printf("Enter principal = ");

    scanf("%f", &principal);

    printf("Enter rate of interest = ");

    scanf("%f", &rateofinterest);

    printf("Enter time period in months = ");

    scanf("%f", &timeperoid);

printf("\nSimple Interest = %.2f", simpleInterest(principal, rateofinterest, timeperoid));

return 0;

}

float simpleInterest(float principal, float rateofinterest, float timeperiod)

{

    return (principal \* (rateofinterest / 100) \* (timeperiod / 12));

}

**Q3. 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 EvenOdd(int);

int main(int argc, char \*argv[])

{

    int num;

    printf("Enter number = ");

    scanf("%d", &num);

    printf("%d", EvenOdd(num));

    return 0;

}

int EvenOdd(int num)

{

    if (num % 2 == 0)

        return 1;

    else

        return 0;

}

**Q4. Write a function to print first N natural numbers (TSRN)**

#include <stdio.h>

void naturalNumber(int);

int main(int argc, char \*argv[])

{

    int num;

    printf("Enter number = ");

    scanf("%d", &num);

    naturalNumber(num);

    return 0;

}

void naturalNumber(int num)

{

    for (int i = 1; i <= num; i++)

        printf("%d ", i);

}

**Q5. Write a function to print first N odd natural numbers. (TSRN)**

#include <stdio.h>

void oddNumber(int);

int main(int argc, char \*argv[])

{

    int num;

    printf("Enter number = ");

    scanf("%d", &num);

    oddNumber(num);

    return 0;

}

void oddNumber(int num)

{

    for (int i = 1; i <= num; i++)

        if (i % 2 != 0)

            printf("%d ", i);

}

**Q6. Write a function to calculate the factorial of a number. (TSRS)**

#include <stdio.h>

void fact(int);

int main(int argc, char \*argv[])

{

    int num;

    printf("Enter number = ");

    scanf("%d", &num);

    fact(num);

    return 0;

}

void fact(int num)

{

    for (int i = (num - 1); i > 1; i--)

        num = num \* i;

    printf("%d", num);

}

**Q7. 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 fact(int);

int comb(int, int);

int main(int argc, char \*argv[])

{

    int num, r;

    printf("Enter number = ");

    scanf("%d", &num);

    printf("Enter selected items at a time = ");

    scanf("%d", &r);

    printf("Number of combinations = %d", comb(num, r));

    return 0;

}

int fact(int num)

{

    for (int i = (num - 1); i > 1; i--)

        num = num \* i;

    return num;

}

int comb(int num, int r)

{

    return (fact(num)/(fact(r)\*fact(num-r)));

}

**Q8. 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 fun(int, int);

int fact(int);

int main(int argc, char \*argv[])

{

    int n, r;

    printf("Enter n = ");

    scanf("%d", &n);

    printf("Enter r = ");

    scanf("%d", &r);

    printf("Total arrangements = %d", fun(n, r));

    return 0;

}

int fun(int n, int r)

{

    return (fact(n) / fact(n - r));

}

int fact(int num)

{

    int x = 1;

    for (int i = 2; i <= num; i++)

    {

        x = x \* i;

    }

    return x;

}

**Q9. Write a function to check whether a given number contains a given digit or not (TSRS)**

#include <stdio.h>

int chkDigit(int, int);

int main(int argc, char \*argv[])

{

    int num, digit;

    printf("Enter number = ");

    scanf("%d", &num);

    printf("Enter digit = ");

    scanf("%d", &digit);

    if (chkDigit(num,digit) == 1)

        printf("found");

    else

        printf("not found");

    return 0;

}

int chkDigit(int num, int digit)

{

    int rem;

    while (num != 0)

    {

        rem = num % 10;

        if (rem == digit)

            return 1;

        num/=10;

    }

    return 0;

}

**Q10. 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(int);

int main(int argc, char \*argv[])

{

    int num;

    printf("Enter number = ");

    scanf("%d",&num);

    prime(num);

    return 0;

}

void prime(int num)

{

    printf("Prime factor of %d = ",num);

    for (int i = 2; i <=(num/2); i++)

    {

        if (num%i == 0)

        {

            num/=i;

            printf("%d ",i);

            i=1;

        }

    }

    printf("%d",num);

}