Tasks

Task 1

Compute expression: $\frac{(a^2+b)}{(a+b)^2}$

For *a* and *b float* type variables input from the keyboard. Check the feasibility of the calculation.

Execution examples:

```
Input number a:2
Input number b:3
The value of the expression is:0.28
```

```
Input number a:5
Input number b:-5
Attempt to divide by zero
```

Task 2

Using *double* type variables compute the following expression:

$a^2 + b$ for *c > 0*

$a - b^2$ for *c < 0*

$\frac{1}{a-b}$ for c = 0

Check the feasibility of the calculation.

Execution examples:

```
Input number a:4
Input number b:5
Input number c:1
The value of the expression is:21
```

```
Input number a:10
Input number b:10
Input number c:0
Attempt to divide by zero
```

Task 3

Write a program to calculate the greatest common divisor of two positive integers. Use the Euclid's algorithm but without using a division operation.

Execution examples:

```
Input first number:45
Input second number:5
The greatest common divisor of numbers (45,5) is:5
```

```
Input first number:4567
Input second number:23
The greatest common divisor of numbers (4567,23) is:1
```

Task 4

Write a program to calculate the sum of the decimal digits for a given natural number.

Execution examples:

```
Input number:123
The sum of the decimal digits=6
```

```
Input number:12345
The sum of the decimal digits=15
```

Task 5

Write a program to verify if a given number is prime. Minimise number of divisions.

Execution examples:

```
Input number:35
35 is a compound number divisible by 5
```

```
Input number:17
17 is a prime number
```

Exemplary numbers to test:

3	**129324179**	**375438827**
5	**129324191**	**375438839**
7	**129324269**	**375438851**
11	**129324277**	**375438883**
13	**129324337**	**437846173**
17	**129324341**	**437846179**
19	**129324359**	**437846191**
23	**129324367**	**437846209**
29	**129324421**	**437846243**
129324073	**129324443**	**100000000000000003**
129324109	**375438751**	**100000000000000013**
129324113	**375438779**	**100000000000000019**
129324131	**375438809**	**100000000000000021**
129324133	**375438823**	**100000000000000049**

Task 6

Write a program to print a Christmas tree consisting of asterisks (*) of a given height. The Christmas tree is to start with a single star and increase the width by 2 stars with each row. A Christmas tree is to have a trunk with a height of two characters. The figure is to be padded left with spaces.

Execution examples:

```
Enter Christmas tree heigth:1
*
|
|
```

```
Enter Christmas tree heigth:5
    *
   ***
  *****
 *******
*********
    |
    |
```

Task 7

For a square matrix entered from the keyboard line-by-line, examine whether the sum of elements above the main diagonal is greater than the sum of elements below this diagonal.

Execution example:

```
Enter rank of the matrix:3
Input element [1,1]=1
Input element [1,2]=2
Input element [1,3]=3
Input element [2,1]=4
Input element [2,2]=5
Input element [2,3]=6
Input element [3,1]=7
Input element [3,2]=8
Input element [3,3]=9
The sum of the elements above the main diagonal (= 11) is not greater than the sum
of the elements below the main diagonal (= 19)
```

Task 8

Write a program to sort in ascending order integers input from the keyboard. The program should ask the user about the number of elements to enter <1,10>, verify the number entered, read the numbers and output the sorted elements. Handle exceptions.

Execution example:

```
Enter the number of items to sort <1 .. 10>: -2
Incorrect number, try again ...
Enter the number of items to sort <1 .. 10>: 4
Input number [1]=1
Input number [2]=6
Input number [3]=2
Input number [4]=4
Sorted elements:
Element [1]=1
Element [2]=2
Element [3]=4
Element [4]=6
```

Task 9

Write a program to test if the entered integer is prime. Do not use division operation (use the Eratosthenes sieve and an array of *bool* elements). Handle exceptions.

Execution examples:

```
Input number:35
Number 35 is a composite number divisible by 5
```

```
Input number:17
Number 17 is a prime number
```

Task 10
Write a program to calculate the value of a polynomial. Use a single dimension array to store coefficients. Use the Horner scheme. Handle exceptions.

Execution example:

```
Enter the degree of the polynomial:3
Enter the constant term:5
Enter the coefficient for x^1:1
Enter the coefficient for x^2:0
Enter the coefficient for x^3:1
Enter the value for the variable x:3
The value of the polynomial is:35
```

Task 11
Write a program showing each even number *x* of the given closed interval as the sum of 2 prime numbers. Use the method to check if a number is a prime. Handle exceptions.

Task 12
Write a program to print all prime numbers from 1 to the value entered from the keyboard. Handle exceptions.

Task 13
A twin prime is a prime number that is either 2 less or 2 more than another prime number, for example {3,5}, {11,13}. Write a program to print all the twin primes in the range from 1 to the value entered from the keyboard. Handle exceptions.

Task 14
A perfect number is a positive integer that is equal to the sum of its proper positive divisors, that is, the sum of its positive divisors excluding the number itself, for example: 1 + 2 + 4 + 7 + 14 = 28. Write a program to output all perfect numbers in the range from 1 to the value entered from the keyboard. Handle interrupts.

Task 15
Amicable numbers are two different numbers so related that the sum of the proper divisors of each is equal to the other number. For example, numbers 220 and 284 are amicable because the proper divisors of 220 are 1, 2, 4, 5, 10, 11, 20, 22, 44, 55 and 110, of which the sum is 284, and the proper divisors of 284 are 1, 2, 71 and 142, of which the sum is 220. Write a program to output all amicable numbers in the range from 1 to the value entered from the keyboard. Handle interrupts.

Task 16
Write a program to compute binomial coefficient $\binom{n}{k}$. Use a method to compute the value of the factorial of a given number n. Handle interrupts

---

Task 17
Write a program to calculate for each natural number from the input string the sum of the digits of its decimal expansion (☞ *String.Split*()). Use a method to compute the sum of the digits of the decimal expansion of a given natural number. Handle interrupts

---

Task 18
Write a program that finds all prime numbers that can be created from the digits of a given integer. For example, from the digits of the number 1379 you can create 31 prime numbers. Note: when building the first number, each digit can be used only once. Handle exceptions.