

## Källkoder

```
static void Main(string[] args)
{
    Console.WriteLine("Mata in en textrad:");
    string input = Console.ReadLine();

    int countSmalla = input.Count(c => c == 'a');
    int countBigA = input.Count(c => c == 'A');

    Console.WriteLine("Antal a: {0}\nAntal A: {1}", countSmalla, countBigA);
    Console.ReadLine();
}
```

## Uppgift 1A

```

static void Main(string[] args)
{
    int biggest = 0;
    int secondBiggest = 0;
    string temp = string.Empty;
    string input;

    // Skriver ut meddelandet.
    Console.Write(strings.InputMessage);

    // Vi säger att input ska vara ReadLine väntar på att användaren skriver.
    input = Console.ReadLine();

    // För varje värde i det som skrivs in så räknar
    // den upp och kollar om i är längden.
    // Även att den kollar om inte det är något samt om räkningen är inte lika med.
    int i = 0;
    foreach (var tal in input)
    {
        i++;

        if (i == input.Length)
        {
            temp += tal;
        }

        if (tal != ' ' && i != input.Length)
        {
            temp += tal;
        }
        else
        {
            int number;
            int.TryParse(temp, out number);

            // Om number är större än största talet så sätter vi secondBiggest är lika med största.
            // Samt att största talet är number.
            // Annars om inte number är större än andra största talet så säger vi att
            // Andra största ska vara number.
            if (number > biggest)
            {
                secondBiggest = biggest;
                biggest = number;
            }
            else if (number > secondBiggest)
            {
                secondBiggest = number;
            }

            // Sätter att strängen temp ska vara tom.
            temp = string.Empty;
        }
    }

    // Skriver ut en textsträng med antal.
    Console.WriteLine(strings.NextBiggest, secondBiggest);

    Console.ReadLine();
}

```

## Uppgift 1B

```

static void Main(string[] args)
{
    // Skapar upp 3 integer samt en int array som heter numbers även en text sträng som i detta fall heter input.
    int zeroes = 0;
    int even = 0;
    int odd = 0;
    int[] numbers;
    string input;

    // Skriver ut i consolen en text sträng.
    // Säger att input ska vara Readline, Även att integer numbers ska vara ett nytt list objekt med längderna i listan som skickas in.
    Console.WriteLine(message.Heltal);
    input = Console.ReadLine();
    numbers = new int[input.Length];

    // En for sats som säger att man försöker göra om strängen till text med hjälp utav TryParse. Då vi skickar ut out numbers[[]]
    for (int i = 0; i < input.Length; i++)
    {
        int.TryParse(input[i].ToString(), out numbers[i]);
    }

    // För varje siffra, om någon siffra är 0 så lägger den till nollorna samt för num % 2 == 0 annars om talet är udda så lägger den till udda tal ++
    foreach (var num in numbers)
    {
        if (num == 0)
        {
            zeroes++;
        }
        else if (num % 2 == 0)
        {
            even++;
        }
        else
        {
            odd++;
        }
    }

    // Skriver ut text sträng samt udda, nollor samt jämna tal.
    Console.WriteLine(message.HeltalAlla, zeroes, odd, even);
    Console.ReadLine();
}

```

## Uppgift 1C

```

while (true)
{
    string input;
    string reverse = "";

    // Skriver ut en text sträng samt sätter input = ReadLine
    Console.Write(message.TextRow);
    input = Console.ReadLine();

    // Om texten är tom så skriver den ut att texten är tom.
    if (input == "")
    {
        Console.BackgroundColor = ConsoleColor.Red;
        Console.WriteLine(message.TextEmpty);
        Console.ResetColor();
    }

    // Om texten innehåller några stora bokstäver så får man fel.
    else if (input.Any(c => char.IsUpper(c)))
    {
        Console.BackgroundColor = ConsoleColor.Red;
        Console.WriteLine(message.TextRowSmall);
        Console.ResetColor();
    }
    else
    {
        for (int j = input.Length - 1; j >= 0; j--)
        {
            reverse += input[j].ToString();
        }

        // Annars så går den igenom texten och plussar på värdet i arrayen (j) till strängen.
        if (reverse == input)
        {
            Console.ForegroundColor = ConsoleColor.Green;
            Console.WriteLine(message.palindrom, input);
            Console.ResetColor();
            break;
        }

        // Om namnet baklänges är lika med namnet framlänges så skriver den ut ett palindrom annars inte.
        else
        {
            Console.ForegroundColor = ConsoleColor.Red;
            Console.WriteLine(message.notPalindrom, input);
            Console.ResetColor();
        }
    }
}

Console.ReadLine();

```

## Uppgift 3A

```

static void Main(string[] args)
{
    // Skapar upp ett nytt objekt utav Fraction klassen samt skicka med argument
    Fraction firstFraction = new Fraction(1, 3);
    Fraction secondFraction = new Fraction(-1, 3);

    // Skriver ut en text sträng med Fraction.add som jag skickar med firstFraction
    // samt andra lägger till ny rad samt skickar jag med nämnaren, täljaren och andra Nämnaren.
    Console.WriteLine(calc.newAddi + Fraction.add(firstFraction, secondFraction) + calc.row,
        firstFraction.Numerator, firstFraction.Denominator, secondFraction.Numerator, secondFraction.Denominator);

    // Skriver ut en text sträng med Fraction.multiply som jag skickar med firstFraction
    // samt andra lägger till ny rad samt skickar jag med nämnaren, täljaren och andra Nämnaren.
    Console.WriteLine(calc.newMulti + Fraction.multiply(firstFraction, secondFraction) + calc.row,
        firstFraction.Numerator, firstFraction.Denominator, secondFraction.Numerator, secondFraction.Denominator);

    Console.ReadLine();
}

```

## Uppgift 3B- Program (Main metoden)

```

class Fraction
{
    private int _denominator; // Nämnare
    private int _numerator; // Täljare

    // Konstruktör som skapar och initialiserar ett nytt bråktalet.
    public Fraction(int numerator, int denominator)
    {
        // Metoden getNumerator som returnerar täljaren.
        public int Numerator
        {
            // Metoden getDenominator som returnerar nämnaren. Nämnaren får inte vara noll.
            public int Denominator
            {
                // Metoden isNegative som ger true om det är ett negativt bråktalet.
                public bool isNegative()
                {
                    // Metoden add som adderar den första bråktalet med den andra och sedan returnerar ett nytt bråktalet.
                    public static Fraction add(Fraction firstFraction, Fraction secondFraction)
                    {
                        // Metoden multiply som multiplicerar den först bråktalet med den andra bråktalet och returnerar ett nytt bråktalet.
                        public static Fraction multiply(Fraction firstFraction, Fraction secondFraction)
                        {
                            // isEqualTo som jämför två Fraction-instanser och ser om de representerar samma bråktalet.
                            public static bool isEqualTo(Fraction firstFraction, Fraction secondFraction)
                            {
                                // toString som returnerar en strängpresentation av bråktalet på form T/N.
                                public override string ToString()
                                {

```

## Uppgift 3B-Fraction klassen (ihop stukad)

## Referenser

<http://sv.wikipedia.org/wiki/Scrum>

<http://stackoverflow.com/questions/20032450/detect-if-a-string-contains-uppercase-characters>

<http://www.c-sharpcorner.com/Blogs/13822/program-to-check-whether-a-string-palindrome-is-or-not.aspx>

<https://coursepress.lnu.se/kurs/iterativ-mjukvaruutveckling/forelasning-1/>

<https://coursepress.lnu.se/kurs/iterativ-mjukvaruutveckling/forelasning-2/>