1 Algorithm for Implementing Beta Functions

1.1 Algorithm - 1

In this algorithm, Beta function uses Gamma function to evaluate its value. Moreover, Gamma function uses factorial method to calculate its value. Hence, the factorial of the negative numbers is not possible therefore, the domain of beta function is all positive integers. i.e. $\forall p, q \in \mathbb{Z}^+$.

$$B(p,q) = \frac{\Gamma p \Gamma q}{\Gamma(p+q)}$$

$$\Gamma p = (p-1)!$$

1.1.1 Advantages

1. This algorithm is easier to implement and debug as programmer needs to implement only one factorial function to calculate beta function.
2. This way of calculating beta function gives more accurate answer compare to other algorithm. The second algorithm depends on the value e which is infinite number.
3. This algorithm is faster to execute and performs better then other algorithm.

1.1.2 Disadvantages

1. This algorithm can only be used for positive real integers, as the factorial of the fraction numbers and negative number can not possible.

Problem 3

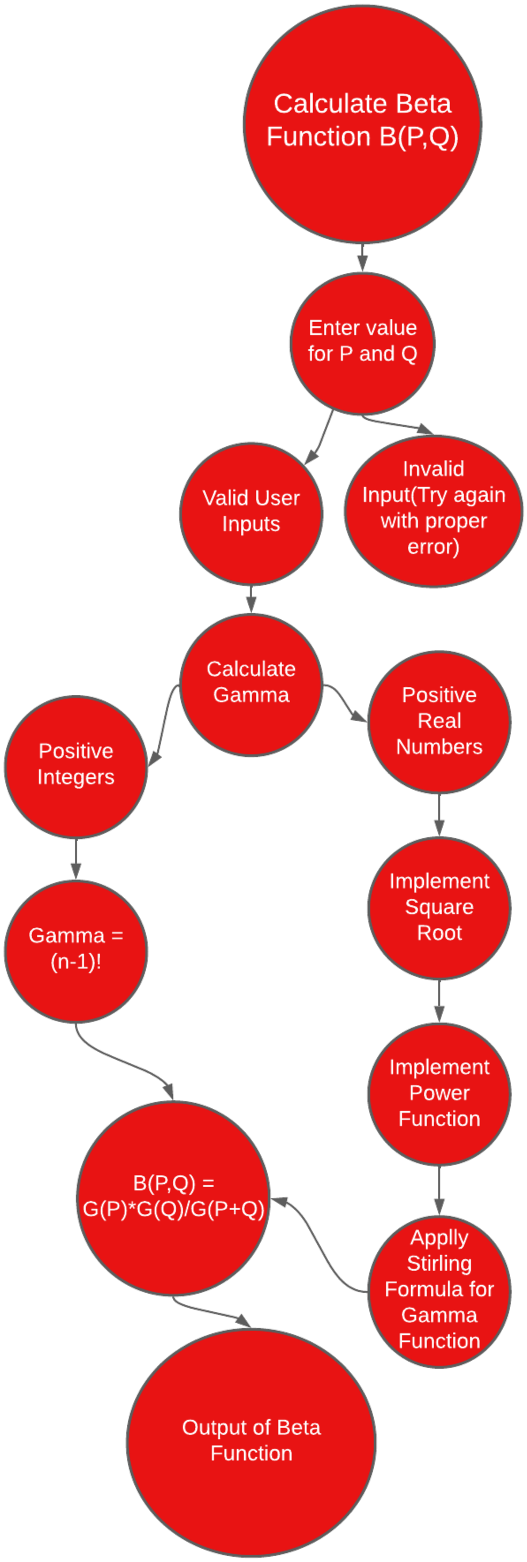

Figure 1: Mind Map for Beta Function

Algorithm 1 Calculate Beta Function with the help of Factorial

Require: $p > 0$ and $q > 0$ ▷ i.e. $p, q \in \mathbb{Z}^+$
Result: $B(p, q)$

```
procedure CalculateFactorial(value)
    result ← 1
    for i ← 2 to value do
        result ← result * i
    end for
    return result                          ▷ Return Factorial of the value
end procedure

procedure CalculateGamma(value)
    result = CalculateFactorial(value − 1)
    return gamma                           ▷ It returns the gamma value
end procedure

procedure CalculateBeta(p, q)
    value1 ← CalculateGamma(p)
    value2 ← CalculateGamma(q)
    r ← p + q
    value3 ← CalculateGamma(r)
    beta ← (value1*value2)/value3
    return beta                            ▷ It returns the beta value
end procedure

result ← CalculateBeta(p, q)               ▷ Final result of Beta(p, q)
```

1.2 Algorithm - 2

We can implement beta function with the help of Stirling's approximation for factorials. Stirling's approximation is approximation method. This method is also for accurate results for small value of p.
The sterling's approximation equation is represented as:

$$B(p, q) = \frac{\Gamma p \Gamma q}{\Gamma(p + q)}$$

$$\Gamma p = \sqrt{\frac{2\pi}{p}}(\frac{p}{e})^p$$

1.2.1 Advantages

1. The algorithm can compute beta function for all the positive real numbers i.e. $p > 0$ and $q > 0$.
2. The algorithm can evaluate result of beta function for all the positive integers i.e. $\forall p, q \in \mathbb{Z}^+$.
3. The algorithm give a approximation value for the integration formula. So, we can able to compute beta function without computing integration.

1.2.2 Disadvantages

1. This algorithm can give only accurate results for the integration formula.
2. This algorithm is quite complex compare to first algorithm even though we are not calculating integration formula.
3. There is more different between actual results and the required result for the small values when the algorithm is implemented. However, for all larger values, the difference between both values becomes narrower.

Problem 3

Algorithm 2 Calculate Beta Function with the help of Stirling's Approximation

```
Require: p > 0 and q > 0                                   ▷ i.e. p, q ∈ Z+
Result: B(p, q)
  procedure CalculateSquareRoot(value)
      squareRoot ← value/2
      repeat
          result ← squareRoot
          result = (result + (value/result))/2
      until (result − squareRoot) ≠ 0
      return squareRoot                                    ▷ Return Square Root
  end procedure

  procedure CalculatePower(value, power)
      result ← 1
      for i ← 1 to power do
          result ← result ∗ value
      end for
      return result                                        ▷ Return base to the power
  end procedure

  procedure CalculateGamma(value)
      intermediateValue1 ← CalculatePower(value/e, value)
      intermediateValue2 ← CalculateSquareRoot(2π/value)
      gamma = intermediateValue1 ∗ intermediateValue1
      return gamma                                         ▷ It returns the gamma value
  end procedure

  procedure CalculateBeta(p, q)
      value1 ← CalculateGamma(p)
      value2 ← CalculateGamma(q)
      r ← p + q
      value3 ← CalculateGamma(r)
      beta ← (value1∗value2)/value3
      return beta                                          ▷ It returns the beta value
  end procedure

  result ← CalculateBeta(p, q)                             ▷ Final result of Beta(p, q)
```

References

[1] Stirling's Approximation Accessed: 16-07-2022 URL: https://en.wikipedia.org/wiki/Stirling's_approximation#Stirling's_formula_for_the_gamma_function