CONCORDIA UNIVERSITY

SOEN 6011: Software Engineering Processes

ETERNITY:FUNCTION F4
Γ(x)

Tavtej Singh Lehri
Student ID: 40121745

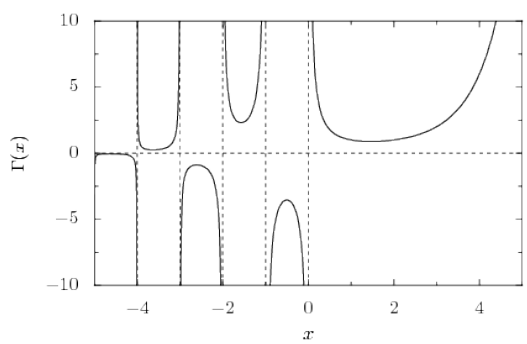

Figure 1: Graph of Gamma Function.[2]

https://github.com/tavtejS07/SOEN-6011

Contents

1 Problem 1

1.1 Description

Gamma function is said to be an extension of the factorial function used for complex numbers. For any positive integer Gamma function is defined as below.

$$\Gamma(x) = (x-1)\Gamma(x-1) \Rightarrow \Gamma(x) = (x-1)! \tag{1}$$

Definition: The function which is improper integral of another function is defined as Gamma function.[1] Using the integral formula below we can define Gamma function. **Note:** For all positive real number x(i.e $Re(x) > 0$), the integral follows absolute convergence.[1] This was derived by **Daniel Bernoulli**.

$$\Gamma(x) = \int\limits_0^\infty t^{x-1}e^{-t}dt \tag{2}$$

1.2 Characteristics

1. $\Gamma(x)$ is defined and analytic for it's domain.[1]
2. $\Gamma(x)$ displays the recursive property for $x > 0$. This is displayed in equation 1 above.
3. $\Gamma(x)$ is a meromorphic function, with $\mathbb{Z} \leq 0$ as poles.[2]

1.3 Domain

Set of positive real numbers. $x \in \mathbb{R}$ and $x > 0$

1.4 Co-Domain

For a specific domain, co-domain for Gamma function is

$$\mathbb{R} > 0 = \{x \in \mathbb{R} | x > 0\}$$

2 Problem 2

2.1 Assumptions

Assumption 1

- **ID** = Gamma_Asump_01
  - Input for the function $\Gamma(x)$ is a real number, where $\mathbb{R} > 0$.

Assumption 2

- **ID** = Gamma_Asump_02
  - At no point is the value of $\Gamma(x) = 0$

Assumption 3

- **ID** = Gamma_Asump_03
  - For the input values which are not within the domain, system throws an error and we get an undefined result.

2.2 Functional Requirements

Requirement 1

- **ID** = Gamma_FR_001
- **Type** = Functional Requirement
- **Version** = 1.0
- **Difficulty** = High
- **Description** = All the inputs should be within the domain. If the value lies outside the domain, error should be thrown by the system.

Requirement 2

- **ID** = Gamma_FR_002
- **Type** = Functional Requirement
- **Version** = 1.0
- **Difficulty** = High
- **Description** = All the inputs which are positive integers should calculate the factorial of that integer. This will be the output of gamma function.
- **Rationale** = As gamma function follows recursive property for $\mathbb{Z} > 0$. $\Gamma(x) = (x-1)\Gamma(x-1)$

Requirement 3

- **ID** = Gamma_FR_003
- **Type** = Functional Requirement
- **Version** = 1.0
- **Difficulty** = Medium
- **Description** = User should provide a single input only.
- **Rationale** = input is either a non-negative integer or a real number greater than 0.

3 Problem 3

3.1 Algorithms

Algorithm 1 Lanczos Approximation for StrictMath

1: **if** x is negative or is NaN **then**
2: NaN
3: **if** x is equal to 0 **then**
4: **return** 1
5: **if** x is greater than 0 **then**
6: *double* $d = (x - 0.5) * log(x + 4.5) - (x + 4.5)$
7: *double* $e = 1.0 + X1/(x + 0) - X2/(x + 1) + X3/(x + 2)-$
8: $X4/(x + 3) + X5/(X + 4) - X6/(x + 5)$
9: **return** $d + log(d1 * sqrt(2 * \pi)) - - > logGamma$
10: **Using the value returned in Line 9 calculate** $\Gamma(x)$
11: **return** $exp(logGamma(x))$

Algorithm 2 Stirling's Approximation for StrictMath

1: **if** x is negative or is NaN **then**
2: NaN
3: **if** x is equal to 0 **then**
4: **return** 1
5: **if** x is greater than 0 **then**
6: **return** sqrt(2*$\frac{\pi}{x}$)*($\frac{x}{e}$)^{x}

3.2 Technical Aspects

Algorithm 1 is Lanczos approximation of the function $\Gamma(x)$. This is an alternative to the Stirling's approximation. The advantage of this is the less number of single or double floating point precision required. If a real constant is known then we can easily calculate the coefficients in advance and use a single formula.

Algorithm 2 is the approximation of factorials. It has Big Oh! of O(nlogn).

Calculation factorial for larger numbers take time if we implement n!. Using the Stirling's approximation it reduces the time of calculation.

4 Problem 4

4.1 Debugger

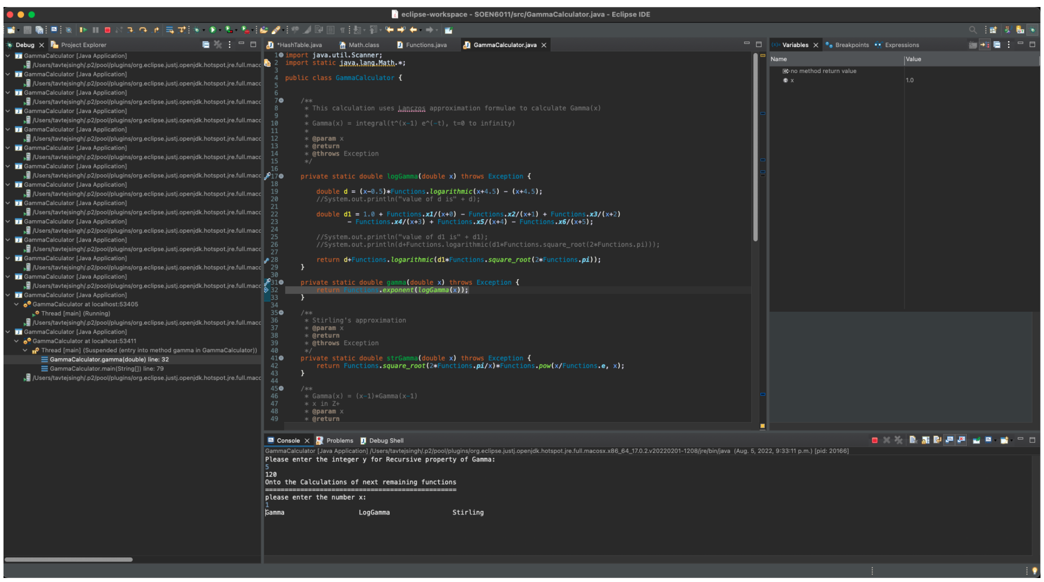

Figure 2: Debugger Eclipse.

The purpose of debugger is to run the program interactively. One can view the variable by variable execution in the debugging perspective. For the program ANT debugger was used which has features such as Breakpoints, Variable tracking, parameter specification and so on.

4.2 Checkstyle

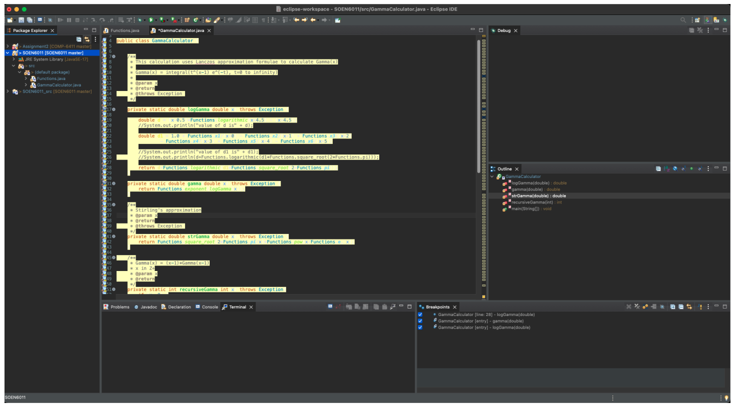

Figure 3: Checkstyle Eclipse.

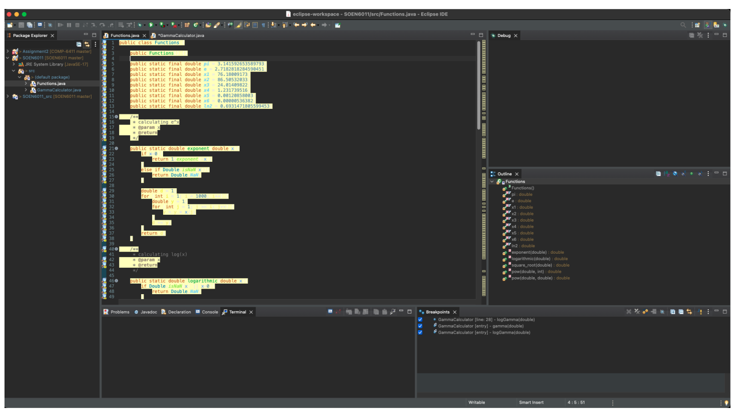

Figure 4: Checkstyle Eclipse.

Bibliography

[1] Libretexts. (2022, February 27). *E14.2: Definition and properties of the gamma function.*Mathematics LibreTexts. Retrieved July 25, 2022, from `https://math.libretexts.org/Bookshelves/Analysis/Complex_Variables_with_Applications_(Orloff)/14%3A_Analytic_Continuation_and_the_Gamma_Function/14.02%3A_Definition_and_properties_of_the_Gamma_function`

[2] Gamma function. (2011, July 25). *Gamma function*- Knowino. (n.d.). Retrieved July 27, 2022, from `https://www.tau.ac.il/ tsirel/dump/Static/knowino.org/wiki/Gamma_function.html`

[3] The Trustees of Princeton University. (n.d.). *Gamma.java.* Princeton University. Retrieved August 5, 2022, from `https://introcs.cs.princeton.edu/java/91float/Gamma.java.html`