**DAILY ASSESSMENT FORMAT**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 25 May 2020 | **Name:** | Anupama J S |
| **Course:** | DSP | **USN:** | 4AL16EC005 |
| **Topic:** | Introduction to fourier series and fourier transform | **Semester & Section:** | 8th sem “A”section |
| **Github Repository:** | AnupamaJS |  |  |

|  |
| --- |
| **FORENOON SESSION DETAILS** |
| **Image of session**  **C:\Users\User\Pictures\Screenshots\Screenshot (208).png**    C:\Users\User\Pictures\Screenshots\Screenshot (207).png |
| **Report – Report can be typed or hand written for up to two pages.**  FOURIER SERIES  In mathematics, a Fourier series (/ˈfʊrieɪ, -iər/[1]) is a periodic function composed of harmonically related sinusoids, combined by a weighted summation. With appropriate weights, one cycle (or period) of the summation can be made to approximate an arbitrary function in that interval (or the entire function if it too is periodic). As such, the summation is a synthesis of another function. The discrete-time Fourier transform is an example of Fourier series. The process of deriving the weights that describe a given function is a form of Fourier analysis. For functions on unbounded intervals, the analysis and synthesis analogies are Fourier transform and inverse transform.  The Fourier series is named in honour of [Jean-Baptiste Joseph Fourier](https://en.wikipedia.org/wiki/Jean-Baptiste_Joseph_Fourier) (1768–1830), who made important contributions to the study of [trigonometric series](https://en.wikipedia.org/wiki/Trigonometric_series), after preliminary investigations by [Leonhard Euler](https://en.wikipedia.org/wiki/Leonhard_Euler), [Jean le Rond d'Alembert](https://en.wikipedia.org/wiki/Jean_le_Rond_d%27Alembert), and [Daniel Bernoulli](https://en.wikipedia.org/wiki/Daniel_Bernoulli). Fourier introduced the series for the purpose of solving the [heat equation](https://en.wikipedia.org/wiki/Heat_equation) in a metal plate, publishing his initial results in his 1807 [Mémoire sur la propagation de la chaleur dans les corps solides](https://en.wikipedia.org/wiki/M%C3%A9moire_sur_la_propagation_de_la_chaleur_dans_les_corps_solides" \o "Mémoire sur la propagation de la chaleur dans les corps solides) (Treatise on the propagation of heat in solid bodies), and publishing his Théorie analytique de la chaleur (Analytical theory of heat) in 1822. The Mémoire introduced Fourier analysis, specifically Fourier series. Through Fourier's research the fact was established that an arbitrary (continuous) function can be represented by a trigonometric series. The first announcement of this great discovery was made by Fourier in 1807, before the [French Academy](https://en.wikipedia.org/wiki/Acad%C3%A9mie_fran%C3%A7aise). Early ideas of decomposing a periodic function into the sum of simple oscillating functions date back to the 3rd century BC, when ancient astronomers proposed an empiric model of planetary motions, based on [deferents and epicycles](https://en.wikipedia.org/wiki/Deferent_and_epicycle" \o "Deferent and epicycle).  The [heat equation](https://en.wikipedia.org/wiki/Heat_equation) is a [partial differential equation](https://en.wikipedia.org/wiki/Partial_differential_equation). Prior to Fourier's work, no solution to the heat equation was known in the ,general case, although particular solutions were known if the heat source behaved in a simple way, in particular, if the heat source was a [sine](https://en.wikipedia.org/wiki/Sine) or [cosine](https://en.wikipedia.org/wiki/Cosine) wave. These simple solutions are now sometimes called [eigensolutions](https://en.wikipedia.org/wiki/Eigenvalue,_eigenvector_and_eigenspace" \o "Eigenvalue, eigenvector and eigenspace). Fourier's idea was to model a complicated heat source as a superposition (or [linear combination](https://en.wikipedia.org/wiki/Linear_combination)) of simple sine and cosine waves, and to write the [solution as a superposition](https://en.wikipedia.org/wiki/Superposition_principle) of the corresponding [eigensolutions](https://en.wikipedia.org/wiki/Eigenfunction" \o "Eigenfunction). This superposition or linear combination is called the Fourier series.  From a modern point of view, Fourier's results are somewhat informal, due to the lack of a precise notion of [function](https://en.wikipedia.org/wiki/Function_(mathematics)) and [integral](https://en.wikipedia.org/wiki/Integral) in the early nineteenth century. Later, [Peter Gustav Lejeune Dirichlet](https://en.wikipedia.org/wiki/Peter_Gustav_Lejeune_Dirichlet) and [Bernhard Riemann](https://en.wikipedia.org/wiki/Bernhard_Riemann) expressed Fourier's results with greater precision and formality.  Although the original motivation was to solve the heat equation, it later became obvious that the same techniques could be applied to a wide array of mathematical and physical problems, and especially those involving linear differential equations with constant coefficients, for which the eigensolutions are [sinusoids](https://en.wikipedia.org/wiki/Sine_wave). The Fourier series has many such applications in [electrical engineering](https://en.wikipedia.org/wiki/Electrical_engineering), [vibration](https://en.wikipedia.org/wiki/Oscillation) analysis, [acoustics](https://en.wikipedia.org/wiki/Acoustics), [optics](https://en.wikipedia.org/wiki/Optics), [signal processing](https://en.wikipedia.org/wiki/Signal_processing), [image processing](https://en.wikipedia.org/wiki/Image_processing), [quantum mechanics](https://en.wikipedia.org/wiki/Quantum_mechanics), [econometrics](https://en.wikipedia.org/wiki/Econometrics), [thin-walled shell](https://en.wikipedia.org/wiki/Thin-shell_structure) theory,  etc.  Consider a real-valued function, s(x), that is integrable on an interval of length P, which will be the period of the Fourier series. Common examples of analysis intervals are:  X € [0,1] and P=1  X € [-π, π ] and P=2  FOURIER TRANSFORM  A Fourier transform (FT) is a mathematical transform which decomposes a function (often a function of time, or a signal) into its constituent frequencies, such as the expression of a musical chord in terms of the volumes and frequencies of its constituent notes. The term Fourier transform refers to both the frequency domain representation and the mathematical operation that associates the frequency domain representation to a function of time.  The Fourier transform of a function of time is a complex-valued function of frequency, whose magnitude (absolute value) represents the amount of that frequency present in the original function, and whose argument is the phase offset of the basic sinusoid in that frequency. The Fourier transform is not limited to functions of time, but the domain of the original function is commonly referred to as the time domain. There is also an inverse Fourier transform that mathematically synthesizes the original function from its frequency domain representation, as proven by the Fourier inversion theorem |

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Date:** | 25 May 2020 | **Name:** | Anupama J S |
| **Course:** | Python | **USN:** | 4AL16EC005 |
| **Topic:** | 1. Fixing Programming Errors | **Semester & Section:** | 8th sem “A”section |
| **Github Repository:** | AnupamaJS |  |  |
| **AFTERNOON SESSION DETAILS** | | | |
| **Image of session**  **C:\Users\User\Pictures\Screenshots\Screenshot (210).png** | | | |
| **Report – Report can be typed or hand written for up to two pages.**  There are three kinds of programming errors: parse-time errors, run-time errors, and logical errors. It doesn't matter what language you are using (SAS/IML, MATLAB, R, C/C++, Java,....), these errors creep up everywhere. Two of these errors cause a program to report an error, whereas the third is more insidious because the program might run to completion while silently delivering the wrong answer.  This article describes ways to find and fix each kind of error.  Error, Errors, Everywhere  Suppose you are trying to write a SAS/IML program that computes the factorial of a number, n. The following statements might represent your initial attempt:  proc iml;  n = 200;  fact = 1  do k = 1 to n;  fact = fact \* n;  end;  print fact;  The program contains three errors—one of each kind—for an impressive error-to-line ratio of 50%. Can you find the three errors?  **PARSE-TIME ERRORS**  A parse-time error occurs when the syntax of the program is incorrect. (This is also called a compile-time error for languages such as C/C++ and Java.) A parse-time error is the easiest error to correct because the parser (or compiler) tells you exactly what is wrong and on what line the problem occurs.  Common parse-time errors include mistyping a statement, forgetting a semicolon, or failing to close a set of parentheses. In strongly typed languages such as C/C++, Java, and IMLPlus, you also get a parse-time error when you try to use a variable of one type when a different type is expected. For example, it is an error to pass an integer into a function that is expecting an array or an object of a class.  In PROC IML, parse-time errors are reported in the SAS log. In SAS/IML Studio, you can select Program > Check Syntax to check your program for parse-time errors.  For the example program, SAS/IML Studio reports the following error:  ERROR: The program contains a syntax error.  IMLPlus did not expect the following text: do (4, 1)  There is nothing wrong with the DO statement on Line 4, but there is a missing semicolon at the end of Line 3. As a result, the IMLPlus parser sees the statement fact = 1 do k = 1 to n; which is invalid syntax.  Fix #1: To fix the parse-time error, insert a semicolon at the end of Line 3.  **RUN-TIME ERRORS**  A run-time error does not occur until the program is actually run. Common SAS/IML run-time errors include adding matrices that are different sizes, taking the logarithm of a negative value, and using the matrix index operator to specify indices that do not exist.  A previous blog post shows how to interpret error messages that appear in the SAS log when SAS/IML software encounters a run-time error.  SAS/IML Studio has some nice features for finding and fixing parse-time and run-time errors. When you run the revised test program, the SAS log reports the following error:  »ERROR: Overflow error in \*.  You can jump directly to the location of the error in a program window, by doing the following:  Right-click the error message in the Error Log window. A pop-up menu is displayed.  Select Go to Source. LOGICAL ERRORS By far, the most difficult error to find is the logical error. A program can have a logical error due to a mistyped formula or due to an incorrectly implemented algorithm. The savvy statistical programmer can use the following techniques to find and eliminate logical errors:   * **Test the program on simple cases** for which the result of the program is known. * **Break down the program into a sequence of basic steps** and independently test each component. * **Favor clarity and simplicity** when you initially write the program. After the program is working, you can [profile the code](https://blogs.sas.com/content/iml/2010/10/22/looping-versus-loc-ing-revisited/) and go back to optimize sections that are performance bottlenecks. | | | |