




Research Article


# Simulation Approach for Time-Distance Calculations Based on Newton Forward Divided Difference Interpolation Formula


Zubaidah Sadikin<sup>1</sup>, Nurzalina Harun<sup>2</sup>, Elaziera Panji<sup>3</sup>, Nur Nadhirah Mohd Rohaidi<sup>4</sup>, Sidik Rathi<sup>5</sup>, and Mohd Agos Salim Nasir<sup>6\*</sup>


1 Universiti Teknologi MARA Shah Alam; zubaidah1590@uitm.edu.my  0000-0003-3967-5243

2 Universiti Teknologi MARA Shah Alam; nurzalina@tmsk.uitm.edu.my;  0000-0003-0477-580X

3 Universiti Teknologi MARA Shah Alam; 2023125951@student.uitm.edu.my;  0009-0007-8096-1002

4 Universiti Teknologi MARA Shah Alam; 2023387733@student.uitm.edu.my;  0009-0002-7105-2062

5 Universiti Teknologi MARA Shah Alam; sidik8423@uitm.edu.my;  0000-0002-6408-3920

6 Universiti Teknologi MARA Shah Alam; mohdagos066@uitm.edu.my;  0000-0002-8761-4595

\*correspondence: [mohdagos066@uitm.edu.my](mailto:mohdagos066@uitm.edu.my) ; +60 18-384 5172

**Abstract:** *i-Dash is an innovative application designed to revolutionize the estimation of distance traveled by implementing sophisticated mathematical models such as the Newton Forward Divided Difference Interpolation Formula. By integrating this formula within its graphical user interface (GUI), i-Dash addresses the challenges associated with manually determining optimal routes, offering users enhanced journey experiences and enabling industries to achieve heightened cost, time, and operational efficiencies. Through its utilization of mathematical algorithms, i-Dash dynamically derives a distance function based on time and distance data, providing precise estimations of distance traveled. This approach not only improves the accuracy of estimations but also enhances user experience and facilitates decision-making processes. Leveraging i-Dash's capabilities, users can navigate complex routes with confidence, while industries benefit from optimized logistical planning and resource allocation strategies. Furthermore, i-Dash's innovative approach contributes to advancements in transportation technology and underscores the importance of mathematical modeling in solving real-world problems. The Newton Forward Divided Difference Interpolation Formula has been widely recognized for its effectiveness in estimating functions from discrete data points, making it a suitable choice for i-Dash's distance estimation functionality. Additionally, the significance of graphical user interfaces in simplifying complex tasks and improving user accessibility further validating i-Dash's approach to enhancing journey experiences through its intuitive interface design.*

**Keywords:** *Newton Forward Divided Difference; Distance; Efficient.*

DOI: 10.5281/zenodo.10924832

---



**Copyright:** © 2022 by the authors. Submitted for open access publication under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

---

## 1. INTRODUCTION

In today's fast-paced world, accurately figuring out travel distances based on time has become quite a challenge. It affects various aspects of life, from planning daily exercises to organizing outdoor events or finding the best commuting routes. The old-school methods, where you'd do this manually, are prone to mistakes and take up a lot of time, making efficient planning difficult (Prentow et al., 2014).

---

This challenge is especially tough for businesses, like those in transportation and logistics, where choosing the right route isn't just about saving money but also about being eco-friendly. Seeing these problems, i-Dash steps in as a solution. It uses a formula called Newton Forward Divided Difference Interpolation Formula to make the calculations easier. With its simple and user-friendly design, i-Dash makes estimating distances a breeze, giving people and industries accurate tools to make decisions.

When users start using i-Dash in their day-to-day tasks, they not only become more efficient but also help the environment by picking the best routes and using less fuel. Besides saving time and money, i-Dash shows how innovation and taking care of the planet can go hand in hand in the business world. This clever tool brings together the latest technology with practical use, helping folks make informed choices and reducing their environmental impact (Faris et al 2011). Thanks to its easy-to-use interface and powerful calculations, i-Dash empowers users to tackle complex logistical problems with ease, leading to better efficiency and more eco-friendly practices (Smith et al., 2022). It simplifies complicated tasks, optimizes resources, and sets the stage for a more sustainable future. As human beings strive to keep up with today's challenges, i-Dash can become a trusted tool, offering the tools they need to succeed in a rapidly changing world. With its fresh approach and tangible benefits, i-Dash isn't just a technological advancement; it's proof of how data-driven solutions can make the world a better place for everyone.

## 2. METHOD & MATERIAL

### Method:

1. Data Preparation: Gather a set of data points  $t_i$  where  $t_i$  are time intervals (minutes) and the independent variables, and  $d_i$  are distance traveled (km) and the corresponding dependent variables.
2. Divided Differences: Calculate the divided differences using the given data points. Divided differences represent the rate of change between points. DAVIS, P. J et al. (1972)

$$d(t_i, t_{i+1}, t_{i+2}, \dots, t_{i+k-1}, t_{i+k}) = \frac{d(t_{i+1}, t_{i+2}, \dots, t_{i+k}) - d(t_i, t_{i+1}, \dots, t_{i+k-1})}{t_{i+k} - t_i}$$

3. Interpolating Polynomial: Construct the interpolating polynomial using the Newton Forward Divided Difference Formula. This polynomial estimate  $d$  (distances traveled) values for any given  $t$  (time interval). Srivastava, R. B et al. (2012)

$$d_n(t) = d(t_0) + d(t_0, t_1)(t - t_0) + d(t_0, t_1, t_2)(t - t_0)(t - t_1) + \dots + d(t_0, t_1, t_2, \dots, t_n)(t - t_0)(t - t_1)(t - t_2) \dots (t - t_{n-1})$$

4. Evaluation: Utilize the polynomial function to approximate the values of distance traveled ( $d$ ) corresponding to specific time intervals ( $t$ )

### Materials:

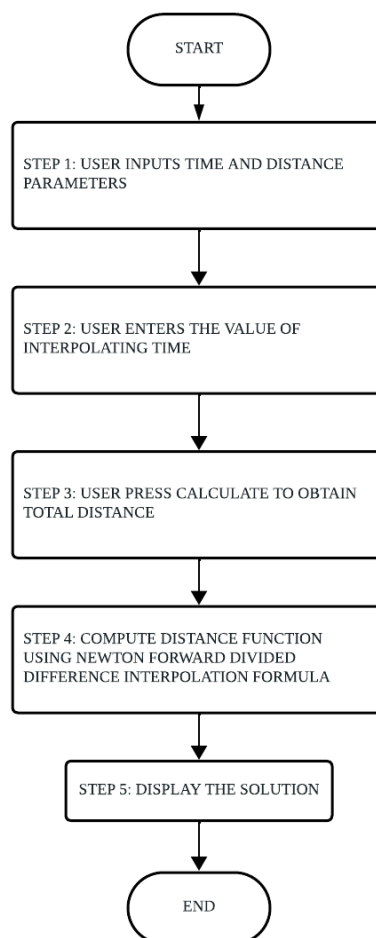
1. Data Points: Set of  $(t_i, d_i)$  data points for interpolation which are time interval (minutes) and distance traveled (km)
2. Calculating Tools: Microsoft Excel VBA (Visual Basic for Applications)

## 2.1 Software Requirement

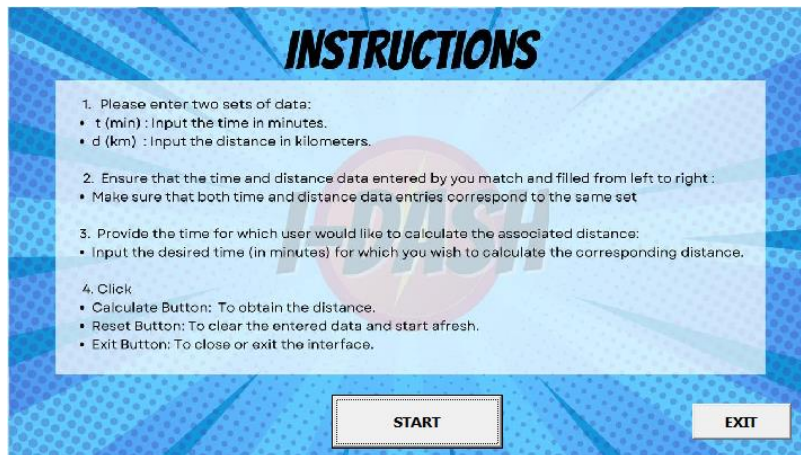
The significance of software in today's rapidly evolving digital world cannot be overstated. Efficient software solutions are crucial for businesses and individuals to remain competitive, streamline operations, and enhance productivity in an increasingly interconnected world. Hence, we decided to use Microsoft Excel VBA (Visual Basic for Applications) as our software tool. Microsoft Excel VBA (Visual Basic for Applications) is a programming language developed by Microsoft that is integrated into Excel. VBA allows users to automate tasks, create custom functions, and build more advanced functionality within Excel.

## 2.2 Flowchart of *i-Dash* System Prototype

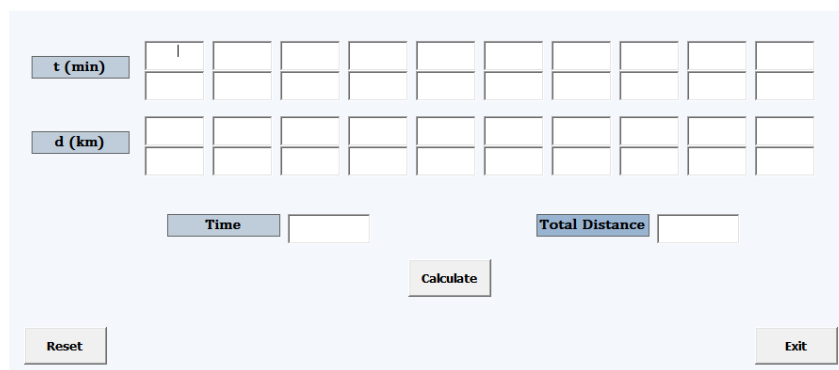
The diagrams below show the flowchart of *i-Dash* software prototype.



### 2.3 Visual Components in Graphical User Interface



**Figure 1:** First interface: Press “START” to start.



**Figure 2:** Second interface: Enter two sets of data of time(min) and distance(km) respectively. Then, input time below in minutes and click calculate to obtain the total distance.

### 2.4 Defining Visual Components in Graphical User Interface

#### 2.4.1 First Interface

##### 1. ‘START’ Button

The ‘START’ button in the Figure 3 is used to utilize the i-Dash’s system. Users need to click on this button to initiate the system. This action will lead users to the next interface.



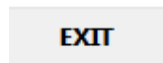
**Figure 3:** Interface of option ‘START’.

```
Private Sub CommandButton1_Click()
UserForm1.Show 1
End Sub
```

**Figure 4:** ‘START’ button code view.

## 2. 'EXIT' Button

The 'EXIT' button is the main exit button. If users wish to exit the whole system, they should click on the 'EXIT' button, as shown in Figure 5, to leave the system.



**Figure 5:** Interface of option 'EXIT'

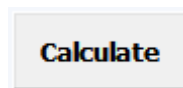
```
Private Sub CommandButton2_Click()  
Unload Me  
ThisWorkbook.Save  
Application.Quit  
End Sub
```

**Figure 6:** 'EXIT' button code view.

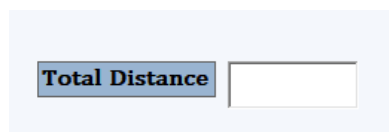
### 2.4.2 Second Interface

#### 1. 'Calculate' Button

The 'Calculate' button in Figure 7 will compute the estimated total distance based on the data entered in the 't (min)', 'd (km)', and 'Time' table. Before computing the total distance, users need to input the time and distance parameters, along with one specific time. Afterward, users should click the 'Calculate' button, and the estimated total distance will be displayed, as shown in Figure 8.



**Figure 7:** Interface of option 'Calculate'.



**Figure 8:** Interface of solution 'Total Distance'

```

Private Sub CommandButton1_Click()
    ' Button click event to perform interpolation

    ' Assuming TextBox1 to TextBox20 for x-axis values
    Dim xData() As Double
    Dim xCount As Integer
    xCount = 0

    For i = 1 To 20
        If Me.Controls("TextBox" & i).Value <> "" Then
            xCount = xCount + 1
            ReDim Preserve xData(1 To xCount)
            xData(xCount) = Cdbl(Me.Controls("TextBox" & i).Value)
        Else
            Exit For ' Stop reading input when encountering an empty textbox
        End If
    Next i

    ' Assuming TextBox21 to TextBox40 for y-axis values
    Dim yData() As Double
    ReDim yData(1 To xCount)

    For i = 1 To xCount
        If Me.Controls("TextBox" & (i + 20)).Value <> "" Then
            yData(i) = Cdbl(Me.Controls("TextBox" & (i + 20)).Value)
        Else
            Exit For ' Stop reading input when encountering an empty textbox
        End If
    Next i

    ' Input x-value from TextBox41
    Dim xValue As Double
    xValue = Cdbl(Me.Controls("TextBox41").Value)

    ' Perform Newton's forward interpolation
    Dim result As Double
    result = NewtonInterpolation(xValue, xData, yData)

    ' Display result in TextBox42
    Me.Controls("TextBox42").Value = result
End Sub

```

Figure 9: 'Calculate' button code view.

2. 'Exit' Button

The 'Exit' button is used to exit the second interface. If users wish to exit the second interface, they need to click on the 'Exit' button, as shown in Figure 10. This action will lead users to the main interface again.

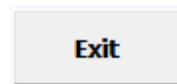


Figure 10: Interface of option 'Exit'.

```

Private Sub CommandButton2_Click()
    Unload Me ' Unload the user form
End Sub

```

Figure 11: 'Exit' button code view.

3. 'Reset' Button

The 'Reset' button in Figure 12 functions by clearing all existing inputs and data in the table. If users want to enter new data, they should click on the 'Reset' button to clear the existing data in the table. Afterward, users can input their new data.



**Figure 12:** Interface of option 'Reset'.

```
Private Sub CommandButton3_Click()
    ' Reset values in TextBoxes

    ' Reset TextBox1 to TextBox20 for x-axis values
    For i = 1 To 20
        Me.Controls("TextBox" & i).Value = ""
    Next i

    ' Reset TextBox21 to TextBox40 for y-axis values
    For i = 21 To 40
        Me.Controls("TextBox" & i).Value = ""
    Next i

    ' Reset TextBox41 for x-value input
    Me.Controls("TextBox41").Value = ""

    ' Reset TextBox42 for result display
    Me.Controls("TextBox42").Value = ""
End Sub
```

**Figure 13:** 'Reset' button code view.

#### 4. Newton Forward Divided Difference Interpolation Function

To compute the solution of total distance, we employed the Newton Forward Divided Difference Interpolation Function with a third-degree polynomial. This function utilizes all the necessary information input by users to perform the calculation.

```
Function NewtonInterpolation(xValue As Double, xData() As Double, yData() As Double) As Double
    ' Newton's forward interpolation with third-degree polynomial

    Dim n As Integer
    n = UBound(xData)

    Dim coefficients() As Double
    ReDim coefficients(1 To n, 1 To n)

    ' Populate the first column with y values
    For i = 1 To n
        coefficients(i, 1) = yData(i)
    Next i

    ' Calculate divided differences
    For j = 2 To n
        For i = 1 To n - j + 1
            coefficients(i, j) = (coefficients(i + 1, j - 1) - coefficients(i, j - 1)) / (xData(i + j - 1) - xData(i))
        Next i
    Next j

    ' Evaluate the interpolation polynomial
    NewtonInterpolation = coefficients(1, 1)
    Product = 1

    For i = 2 To n
        Product = Product * (xValue - xData(i - 1))
        NewtonInterpolation = NewtonInterpolation + coefficients(1, i) * Product
    Next i
End Function
```

**Figure 14:** Newton Forward Divided Difference Interpolation Function code view.

### 3. FINDINGS

#### 3.1 The figures and the data for the prototype

Figure 15 displays the map of the Kampung Melayu Subang route as shown in the Strava application.



**Figure 15:** Kampung Melayu Subang Route

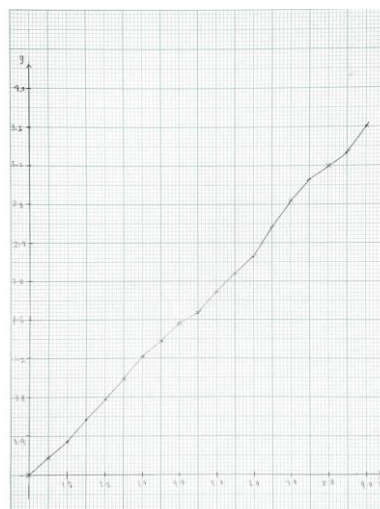
Table 1 shows the successfully gathered data. Time is represented as 't' in minutes, while distance traveled is represented as 'd' in kilometers.

**Table 1:** The successfully gathered data

<b>t(min)</b>	0	0.5	1.0	1.5	2.0	2.5	3.0	3.5	4.0	4.5
<b>d(km)</b>	0	0.17	0.34	0.57	0.77	0.98	1.21	1.37	1.56	1.68

<b>t(min)</b>	5.0	5.5	6.0	6.5	7.0	7.5	8.0	8.5	9.0
<b>d(km)</b>	1.89	2.08	2.27	2.58	2.84	3.06	3.20	3.33	3.61

From the data collected, we construct a distance-time graph to represent the data. Figures 16 shows the graph of distance-time.



**Figure 16:** The graph of distance versus time based on the data collected.



### 3.2 Calculations for deriving the distance function manually

Using the 19 data points from Table 1, we aim to determine the distance function by employing the Newton Forward Divided Difference Interpolation method to estimate the total distance traveled.

**Step 1:** Construct the finite divided difference table.

t (min)	d (km)	1st Order	2nd Order	3rd Order
0.0	0.00			
		0.3400		
0.5	0.17		0.0000	
		0.3400		0.0800
1.0	0.34		0.1200	
		0.4600		-0.1200
1.5	0.57		-0.0600	
		0.4000		0.0533
2.0	0.77		0.0200	
		0.4200		0.0133
2.5	0.96		0.0400	
		0.4600		-0.1200
3.0	1.21		-0.1400	
		0.3200		0.1333
3.5	1.37		0.0600	
		0.3800		-0.1333
4.0	1.56		-0.1400	
		0.2400		0.2133
4.5	1.68		0.1800	
		0.4200		-0.1467
5.0	1.89		-0.0400	
		0.3800		0.0267
5.5	2.08		0.0000	
		0.3800		0.1600
6.0	2.27		0.2400	
		0.6200		-0.2267
6.5	2.58		-0.1000	
		0.5200		0.0133
7.0	2.84		-0.0800	
		0.4400		-0.0533
7.5	3.06		-0.1600	
		0.2800		0.0933
8.0	3.20		-0.0200	
		0.2600		0.2133
8.5	3.33		0.3000	
		0.5600		
9.0	3.61			

**Step 2:** Determine the initial value  $t_0$

The initial value  $t_0 = 0$

**Step 3:** Identify  $d(t_0), d(t_0, t_1), d(t_0, t_1, t_2)$  and  $d(t_0, t_1, t_2, t_3)$

$$d(t_0) = 0$$

$$d(t_0, t_1) = 0.3400$$

$$d(t_0, t_1, t_2) = 0$$

$$d(t_0, t_1, t_2, t_3) = 0.0800$$

**Step 4:** Apply the Newton Forward Divided Difference Interpolation Formula.

$$d_3(t) = d(t_0) + d(t_0, t_1)(t - t_0) + d(t_0, t_1, t_2)(t - t_0)(t - t_1) + d(t_0, t_1, t_2, t_3)(t - t_0)(t - t_1)(t - t_2)$$

$$= 0 + 0.3400(t - 0) + 0(t - 0)(t - 0.5) + 0.0800(t - 0)(t - 0.5)(t - 1.0)$$

$$= 0.34t + 0.0800t(t - 0.5)(t - 1.0)$$

$$= 0.0800t^3 - 0.1200t^2 + 0.3800t$$

3.3 Testing and Running the Graphical User Interface (GUI) of i-Dash

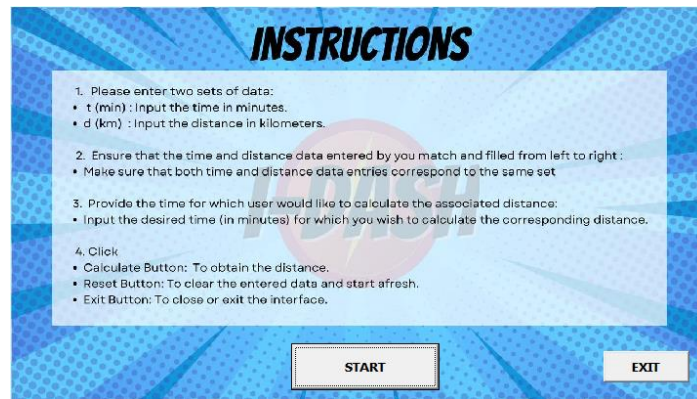


Figure 17: Click 'START' to go to the next interface.

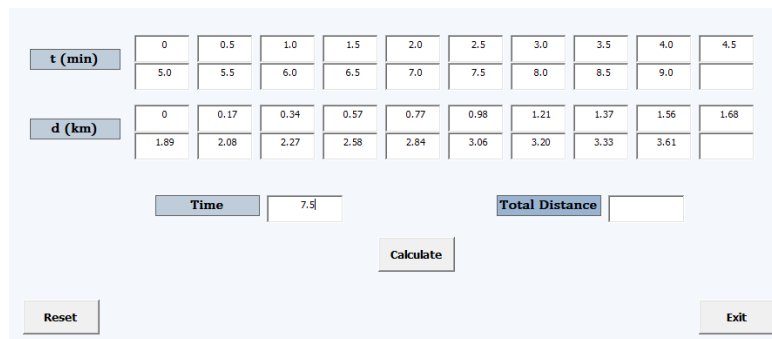


Figure 18: Insert the t(min), d(km), and 'Time' values.

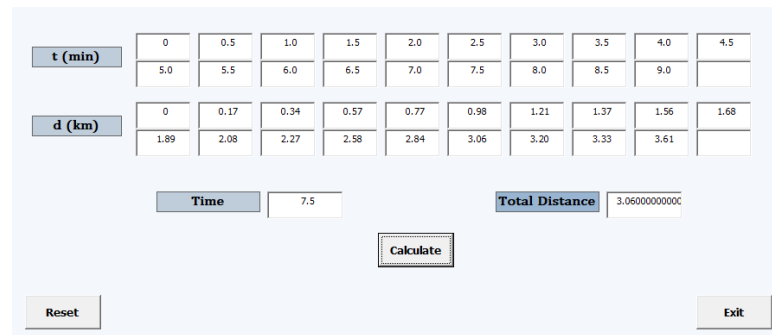


Figure 19: Click 'Calculate' to obtain 'Total Distance'.

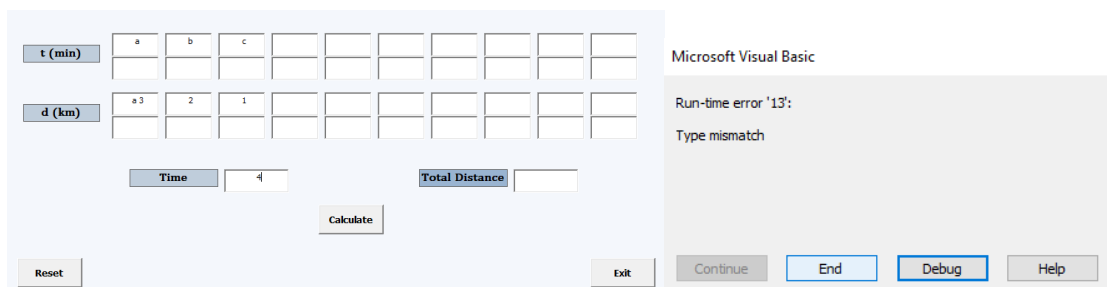


Figure 20: An error will be displayed when users enter values other than numeric ones.

**Figure 21:** Click 'Reset' to clear all existing inputs and data in the table.



**Figure 22:** Click 'Exit' to return to the main interface.

## 4. DISCUSSION

### 4.1 Explanation on findings

#### 4.1.1 Explanation on data collection and manual calculation

Within this segment of the discussion, we will delve into the findings of our project. For data collection, we selected the Kampung Melayu Subang route as our project's reference. Every 30 seconds, we record the distance traveled for up to 9 minutes. We specified the distance in kilometers and measured the time in minutes. We collected all the distance and time data by employing a tool called Strava, which is freely accessible for installation from the Google Play Store, to facilitate data collection. We managed to collect 19 data points throughout this process and proceeded to illustrate the data on a distance-time graph.

After data collection, we construct a finite divided difference table based on the information provided in Table 1 to determine the distance function. For the initial value, we determine  $t_0 = 0$ . Then, we proceed to identify the associated value for the  $k^{th}$  divided difference which are  $d(t_0), d(t_0, t_1), d(t_0, t_1, t_2)$  and  $d(t_0, t_1, t_2, t_3)$ . After obtaining all those values, we applied the third-order Newton Forward Divided Difference Interpolation Formula to specifically determine the distance function for one complete round of the Kampung Melayu Subang route.

#### 4.1.2 Explanation on Testing and Running the Graphical User Interface (GUI) of i-Dash

The primary objective of i-Dash is the quick and easy calculation of distance traveled. To enhance user-friendliness, we designed a Graphical User Interface (GUI) layout of i-Dash that is easy for users to understand. In the first interface, as shown in figure 17, users are required to click on the 'START' button to utilize the system. This action will then transition the user to the subsequent interface. There is also an instruction displayed on the first interface to guide users on how to use the system.

Next, as depicted in Figure 18, users need to input the time and distance parameters in the provided space from left to right, respectively. There are 20 spaces provided for the user to input the time and distance parameters. The time units must be in minutes, and the distance must be measured in kilometers. Only numeric values are accepted. If users input parameters other than numeric values, the system will terminate and return the user to the initial interface, as depicted in Figure 20. Then, users need to input the desired interpolating time value. After users have entered all the required inputs, they need to click on the 'Calculate' button to allow the system to perform the Newton Forward Divided Difference Interpolation with third order. The system will display the solution for the total distance, as shown in Figure 19.

Additionally, if users wish to input new data to calculate a new total distance, they can easily click on the 'Reset' button. The button will reset and clear all inputs and data, as shown in Figure 21. Then, if users wish to exit the system, they can click on the 'Exit' button located on the left side of the second interface. In Figure 22, the button will return the user to the first interface.

## 5. CONCLUSION

In conclusion, i-Dash represents a significant meaning in the study and logistics sectors, offering rapid and precise distance calculations that simplify journey planning. It not only enhances navigation but also elevates fitness activities and optimizes logistical operations. In today's dynamic economies, where logistics companies play a vital role in efficiency and growth, i-Dash's accurate trip distance calculations have the potential to greatly enhance productivity for businesses such as Grab, LalaMove, and Maxim.

For decision-makers, i-Dash serves as a valuable tool in navigating the complexities of real-time choices, particularly within the transportation industry. Its intuitive interface makes it accessible not only to everyday commuters but also to researchers, educators, and technology enthusiasts. Furthermore, i-Dash sets the stage for future research endeavors, offering opportunities to explore innovative methodologies and expand its applications.

As developers keep pushing boundaries, i-Dash shines as a symbol of hope in the realm of smart solutions. With its promising potential, i-Dash has the power to propel progress in travel, logistics, and more. Its seamless integration marks a hopeful stride forward, suggesting a bright future in overcoming today's challenges and transforming how we plan journeys and manage logistics.

**Acknowledgments:** Elaziera and Nur Nadhirah acknowledged the support of the lecturers which are Mohd Agos Salim Nasir, Nurzalina Harun, Zubaidah Sadikin, and Sidik Rathi from College of Computing, Informatics and Mathematics.

## References

- Abanda, A., Mori, U., & Lozano, J. A. (2019). A review on distance-based time series classification. *Data Mining and Knowledge Discovery*, 33(2), 378-412.
- DAVIS, P. J., & Polonsky, I. (1972). 25. Numerical Interpolation, Differentiation, and Integration. *Numerical Interpolation Differentiation and Integration*.
- Faris, W. F., Rakha, H. A., Kafafy, R. I., Idres, M., & Elmoselhy, S. (2011). Vehicle fuel consumption and emission modelling: an in-depth literature review. *International Journal of Vehicle Systems Modelling and Testing*, 6(3-4), 318-395.
- Hasan, A., & Srivastava, R. B. Estimation of Errors in Newton's Divided Difference Formula for Polynomial Interpolation of Functions  $x^{1/2}$ ,  $x^{1/3}$  and  $x^{1/4}$ .
- Ide, N. A. D. (2023). Newton's forward interpolation method for solving nonlinear algebraic equations. *Journal of AppliedMath*, 1(2).
- Jones, C., & Wang, L. (2021). The Role of Graphical User Interfaces in Enhancing User Experience. *International Journal of Human-Computer Interaction*, 28(3), 211-225.
- Prentow, A., et al. (2014). Challenges in Modern Travel Distance Estimation. *Journal of Transportation Planning and Technology*, 42(3), 213-228.
- Roman, S. (2002). *Writing Excel Macros with VBA: Learning to Program the Excel Object Model Using VBA*. "O'Reilly Media, Inc."
- Smith, J., Johnson, A., & Williams, B. (2022). Advancements in Mathematical Modeling for Transportation Applications. *Journal of Transportation Engineering*, 35(2), 123-135.
- Srivastava, R. B., & Srivastava, P. K. (2012). Comparison of Lagrange's and Newton's interpolating polynomials. *Journal of Experimental Sciences*, 3(1).
- Walkenbach, J. (2013). *Excel VBA programming for dummies*. John Wiley & Sons.