LISTINFO

The following is a detailed explanation of how LISTINFO works. The program was designed to be useful in calculating Riemann Sums (Left, Right, Midpoint, Trapezoidal, and Simpsons rules) using information from tables, in particular where the subintervals are not necessarily of equal length. The program also has a feature of using the Intermediate Value Theorem, and Mean Value Theorem, or both in conjunction to determine values of the function, its derivative, and the sign of the second derivative.

Before using the program:
Note that the program runs SetupEditor and ClearAllLists to make sure that the program can use the lists. If the user does not want to lose information in the lists the user has an option to exit and not have any information in lists cleared.

x	x_1	x_2	x_3	$\cdots$	x_{n-2}	x_{n-1}	x_n
$f(x)$	y_1	y_2	y_3	$\cdots$	y_{n-2}	y_{n-1}	y_n

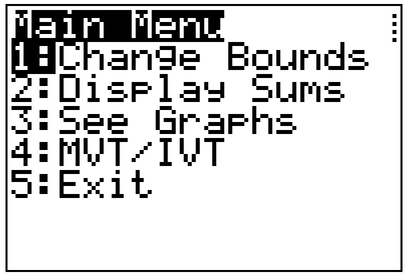
The user is prompted to input the number of points that will be used to calculate the sums. The user does not have to enter the points in any order, as the program stores the *x*-coordinates in L1 and *y*-coordinates in L2 and then executes a SortA(L1,L2) so that the points are automatically listed in their correct order before the calculations of the sums are executed. The program also checks to see that the value for the number of points is an integer greater than 1. The program checks when entering each *x*-coorindate whether there is already a point with the same *x*-coordinate. If so, it notifies the user and asks for a different *x*-coordinate input value.

The Left Sum, Right Sum, and Trapezoidal Sums are calculated using the points entered by the user in consecutive order, even if the subintervals defined by the points are not of equal length.

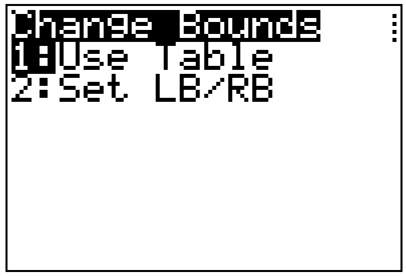
In calculating the Midpoint & Simpsons Sums, the program checks to see if the sequence of *x*-coordinates

$$\underbrace{x_1, x_2, x_3}, \ldots, \underbrace{x_{2n-1}, x_{2n}, x_{2n+1}}, \ldots, \underbrace{x_{n-2}, x_{n-1}, x_n}$$

is first a list of odd length, and then checks to see that the even indexed *x*-coordinates are the midpoint of the interval defined by the previous and subsequent terms. If so, the program will use the intervals (as illustrated by the grouping above) to calculate the Midpoint or Simpsons Sum. If the list is not of odd length, or any evenly indexed *x*-coordinate is not the midpoint of the interval defined by the previous and subsequent terms, the program will not calculate the Midpoint Sum or Simpsons Sum and notify the user.



Change Bounds:



You can choose to use the entire table, or use an entry in the table for the left bound and another for the right bound. Once the bounds are changed, the subsequent calculations will only be used using the entries between the new left and right bounds. The bounds must be *x*-coordinates of the entries in the table, and the left bound must be less than the right bond

Display Sums:

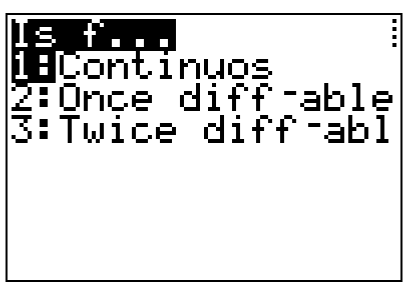
Will display the numerical values of each of the sums. The sums will be calculated based on the left bound and right bound. When the program is first run, the left and right bounds are the first and last entries in the table respectively. Once the bounds have been changed, the sums displayed correspond to the new left and right bounds.

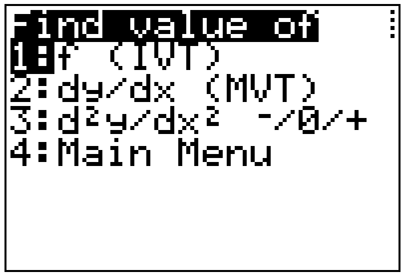
See Graphs:

Will display the graphs of each sum, along with the values of each sum, based on the left bound and right bound.

IVT/MVT:

The user must identify if the function is continuous, differentiable, or twice differentiable on the given interval.





***f* (IVT):**

If the function is continuous, the user can input a value of k and see if IVT can be used to demonstrate if it can be demonstrated that $f(x)=k$ on the given interval. If so, the program will return all the possible pairs of entries in the table between the left and right bounds where IVT can be used to demonstrate $f(x)=k$. The pair of points is displayed as a matrix $\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}$

Since $f(x)$ is continuous, and $\min(y_1,y_2)<k<\max(y_1,y_2)$, by IVT there exists a c such that $x_1<c<x_2$ and $f(c)=k$.

dy/dx (MVT [+IVT]):

If the function is differentiable or twice differentiable the user can see if $f'(x)=k$. This is done in two ways, if the function is once differentiable, the program searches through all possible pairs of points between the left and right bounds such that MVT can be applied with those two points to show that $f'(x)=k$. If such a pair of points exists, it is displayed in a matrix $\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix}$

Since $f(x)$ is differentiable, by MVT there exists a c, $x_1<c<x_2$, such that $f'(c)=\dfrac{y_2-y_1}{x_2-x_1}=k$.

If the desired value of $\dfrac{dy}{dx}=0$, then IVT can be used in conjunction with Rolle's theorem to show that $f'(x)=0$. The following types of points will appear on the screen:

$$\text{Case 1:}\ \begin{matrix} \{x_1,y_1\} \\ \begin{bmatrix} x_2 & y_2 \\ x_3 & y_3 \end{bmatrix} \end{matrix} \qquad \text{Case 2:}\ \begin{matrix} \begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix} \\ \{x_3,y_3\} \end{matrix}$$

Where $x_1<x_2<x_3$ in both cases.

Case 1: Since $f(x)$ is differentiable, $f(x)$ is continuous. By IVT there exists a c_1 such that $x_2<c_1<x_3$ such that $f(c_1)=y_1$. Since $f(x)$ is differentiable, by Rolle's Theorem, there exists a c_2 such that $x_1<c_2<c_1$ and $f'(c_2)=0$.

Case 2: Since $f(x)$ is differentiable, $f(x)$ is continuous. By IVT there exists a c_1 such that $x_1 < c_1 < x_2$ such that $f(c_1) = y_3$. Since $f(x)$ is differentiable, by Rolle's Theorem, there exists a c_2 such that $c_1 < c_2 < x_3$ and $f'(c_2) = 0$.

If the function is twice differentiable, MVT can be used in conjunction with IVT to demonstrate that $f'(x) = k$. If this is the case, the following four points are displayed in two matrices

$$\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix} \begin{bmatrix} x_3 & y_3 \\ x_4 & y_4 \end{bmatrix} \text{ where } x_1 < x_2 \le x_3 < x_4$$

Since $f(x)$ is differentiable, there exists a c_1, $x_1 < c_1 < x_2$, such that $f'(c_1) = \frac{y_2 - y_1}{x_2 - x_1} = m_1$

Since $f(x)$ is differentiable, there exists a c_2, $x_3 < c_2 < x_4$, such that $f'(c_2) = \frac{y_2 - y_1}{x_2 - x_1} = m_2$

Since $x_1 < x_2 \le x_3 < x_4$, we know that $c_1 < c_2$. Let $\min(m_1, m_2) < k < \max(m_1, m_2)$

Since $f(x)$ is twice differentiable, we know that $f'(x)$ is continuous. By IVT there exists a d, where $c_1 < d < c_2$ such that $f'(d) = k$.

d2y/dx2 = -/0/+ (MVTx2)

If the function is twice differentiable MVT can be applied twice to $f(x)$ to determine if $f''(x)$ is positive, negative, or zero in the given interval. This requires that the left and right bounds have at least one entry in the table in between them. If this is not the case, the user must change the left and right bounds so that there is one entry in the table between the left and right bound.

If the function is twice differentiable and there is at least one entry in the table between the left and right bounds, then the following four points are displayed in two matrices:

$$\begin{bmatrix} x_1 & y_1 \\ x_2 & y_2 \end{bmatrix} \begin{bmatrix} x_3 & y_3 \\ x_4 & y_4 \end{bmatrix} \text{ where } x_1 < x_2 \le x_3 < x_4$$

Since $f(x)$ is differentiable, there exists a c_1, where $x_1 < c_1 < x_2$, such that $f'(c_1) = \frac{y_2 - y_1}{x_2 - x_1} = m_1$

Since $f(x)$ is differentiable, there exists a c_2, where $x_3 < c_2 < x_4$, such that $f'(c_2) = \frac{y_2 - y_1}{x_2 - x_1} = m_2$

Since $x_1 < x_2 \le x_3 < x_4$, we know that $c_1 < c_2$.

Since $f(x)$ is twice differentiable, $f'(x)$ is differentiable. We can use MVT with $f'(x)$ and the points (c_1, m_1) and (c_2, m_2) on the graph of $f'(x)$ from above. By MVT there exists a d such that $c_1 < d < c_2$ and $f''(d) = \frac{m_2 - m_1}{c_2 - c_1} \leftrightarrow \frac{(- / 0 / +)}{(+)}$

If you notice any issues or have suggestions for improvements to the program, feel free to e-mail me at:

Marco Rainaldi marcorainaldi@gmail.com