

Introdução à programação G

G é uma linguagem de programação de fluxo de dados de alto nível desenvolvida para aplicações que são

- Interativas
- Executadas em paralelo
- Multicore

Este programa é um diagrama de blocos editado em uma janela de programação G.

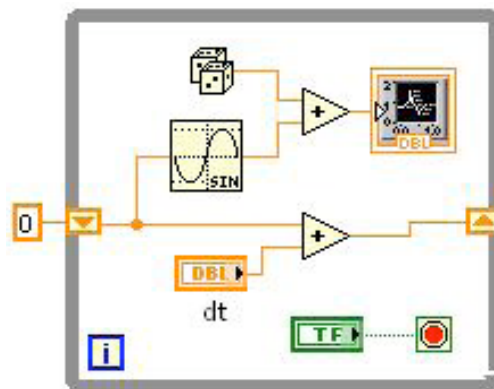


Figura 1. Diagrama de Blocos G

Os dados de entrada e resultados são manipulados e mostrados na janela GUI (G User Interface).

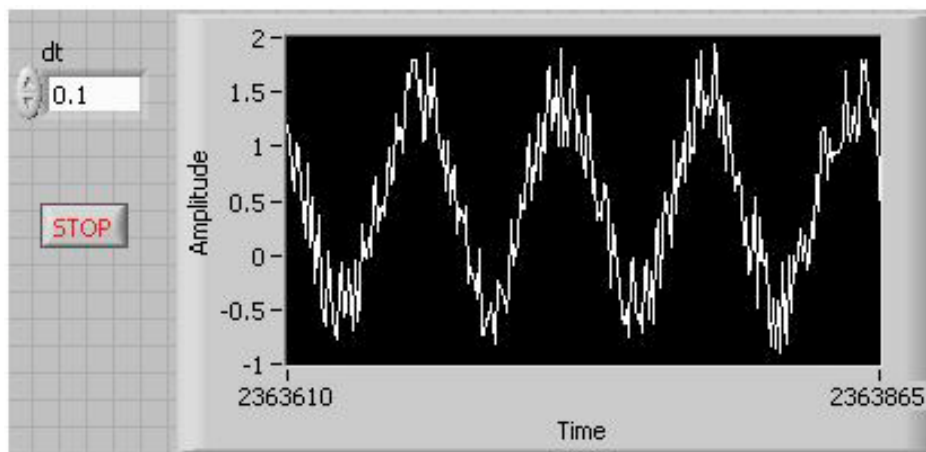


Figura 2. Interface de Usuário

Olá Mundo!

O primeiro programa mostra o texto “Hello graphical interactive parallel multicore world” na janela GUI. Clique com o botão direito do mouse na janela de programação G e selecione o menu **String Constant Functions>>Programming>>String**

Arraste e solte o **String Constant** na janela de programação G como mostrado na figura 3.

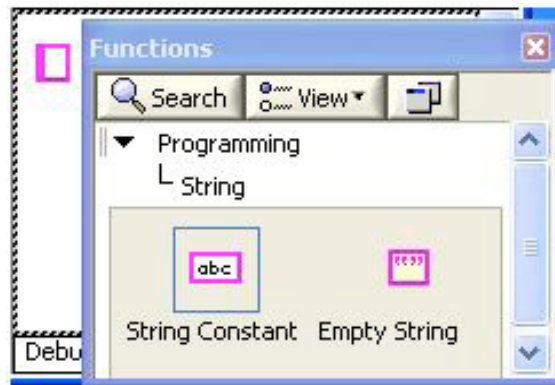


Figura 3. Constante de String

Digite “Hello graphical interactive parallel multicore world.” na **String Constant**

Hello graphical interactive
parallel multicore world.

Figura 4. Constante de String "Hello ... world"

Clique com o botão direito do mouse na janela GUI e selecione um **String Indicator** do menu **Controls>>Modern&>StringPath**

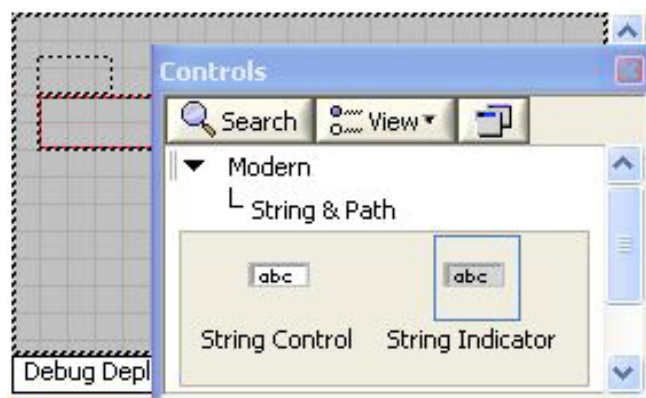


Figura 5. Indicador de String

Solte-o na janela GUI

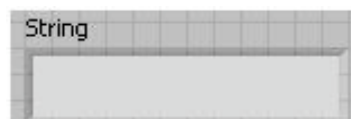


Figura 6. Saída de String

Volte à janela de programação. Veja o terminal de string correspondente ao **String Indicator** na janela GUI. Na medida em que você se aproxima da constante de sting pela direita, o terminal de fiação é destacado e o ponteiro se transforma em um carretel de fio.

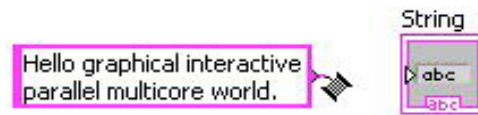


Figura 7. Ligando o Diagrama de Linguagem G

Clique no terminal **“Hello graphical interactive parallel multicore world”** e em seguida clique no terminal triangular **String Indicator** para ligar os terminais. Salve seu programa como **Hello,World.vi**.

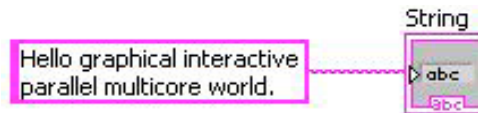


Figura 8. Diagrama de Blocos “ligado”

Volte à janela GUI. Clique no botão run.

Você completou e executou seu primeiro programa em G.

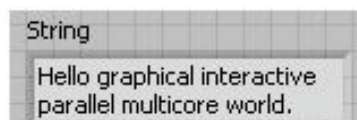


Figura 9. Programa Hello, World Executado

Expressões Aritiméticas

O próximo programa converte graus de Farenheit para Celsius.

Na janela de programação G, selecione as funções de subtração, multiplicação e divisão do menu **Functions>>Mathematics>>Numeric**.

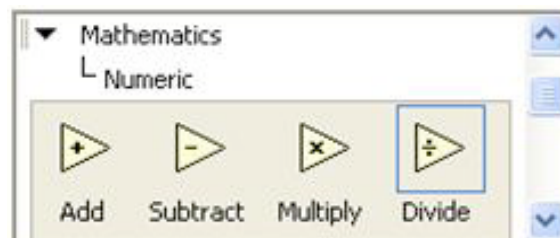


Figura 10. Operações Numéricas

Ligue as funções como mostrado na figura 11



Figure 11. Subtração, Multiplicação, e Divisão

Clique com o botão direito do mouse no terminal no lado superior esquerdo da função de subtração e selecione **Create>>Control** do menu pop-up.

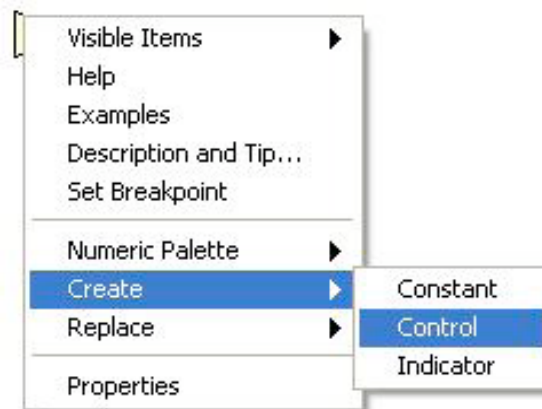


Figura 12. Crie um Controle

Renomei **x** como **Fahrenheit** e ligue o terminal como mostrado na figura 13.

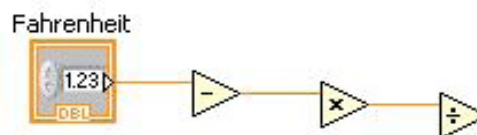


Figura 13. Entrada do Controle Fahrenheit

Clique com o botão direito do mouse no terminal no lado inferior esquerdo da função de subtração e selecione **Create>>Constant** e digite **32.0**.

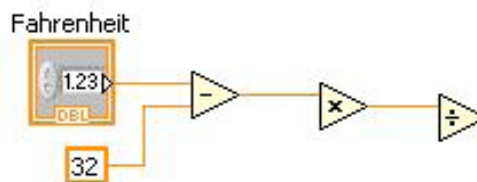


Figura 14. Constante do Numérica para Fahrenheit

Repita o processo para gerar constantes numéricas para as funções de multiplicação e divisão com os valores de **5.0** e **9.0**, respectivamente.

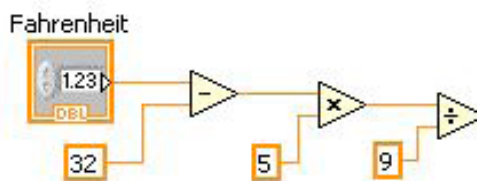
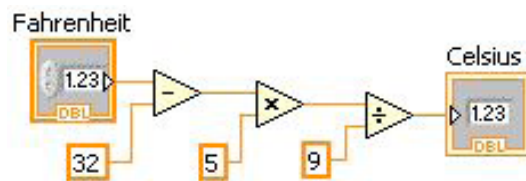


Figura 15. Constantes Numéricas de Fahrenheit

Para completar o programa, clique com o botão direito do mouse no terminal direito da função de

divisão e selecione **Create>>Indicator**. Renomei o x/y para **Celsius**. O diagrama final é mostrado na figura 16.



F

Figura 16. Diagrama Fahrenheit para Celsius

Volte para a janela GUI e execute o programa. Salve o programa como **Celsius.vi**. Experimente com vários valores de Fahrenheit para ver seu valor correspondente em Celsius. Você criou um calculador de Fahrenheit para Celsius

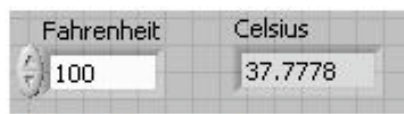


Figura 17. Programa Fahrenheit para Celsius

Funções

Clique em um espaço em branco e arraste para selecionar o diagrama todo

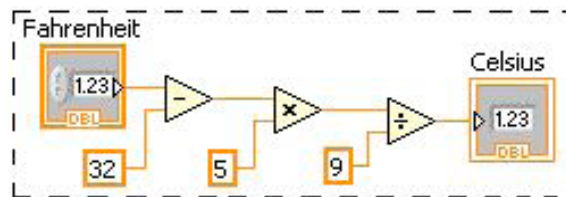


Figura 18. Diagrama de Blocos Selecionado

O diagrama selecionado está destacado como mostrado na Figura 19.

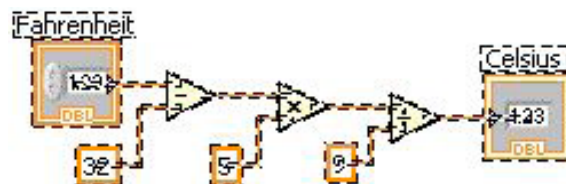


Figura 19. Diagrama de Blocos Selecionado

Do menu **Edit**, selecione **Create SubVI** para criar uma função G. O diagrama resultante é mostrado em **Error! Reference source not found.**

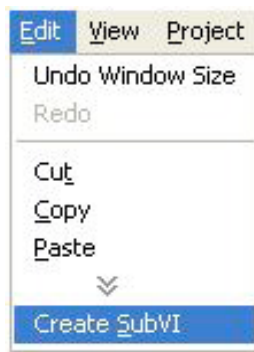


Figura 20 Criando uma Função

Do menu **File** menu, selecione **Save All** e salve a função **Untitled** como **Fahrenheit to Celsius.vi**.



Figura 21. Diagrama com Função

Abra o Fahrenheit to Celsius.vi dando dois cliques no ícone. **Clique com o botão direito do mouse no editor de ícone (canto superior direito) e selecione Edit Icon...**

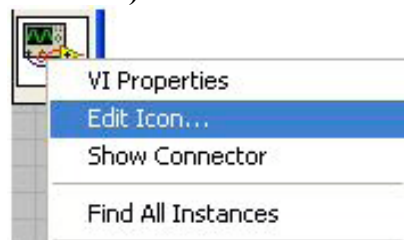


Figura 22. Edite o Ícone

Isto abre o Editor de ícones. Modifique o ícone

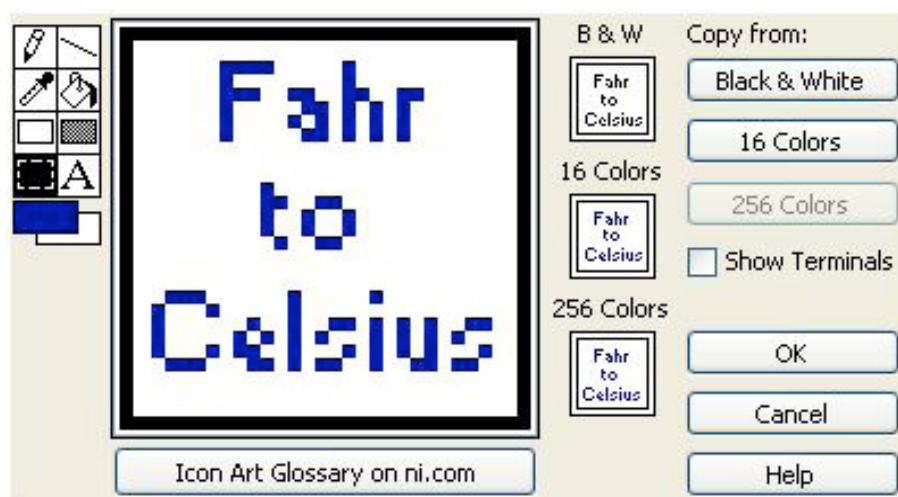


Figura 23. Editor de Ícone

Depois de editar ícone, as funções do ícone são mostradas no canto superior direito da janela GUI. Salve a função, coloque vários valores de entrada e execute a função. Salve a função novamente.

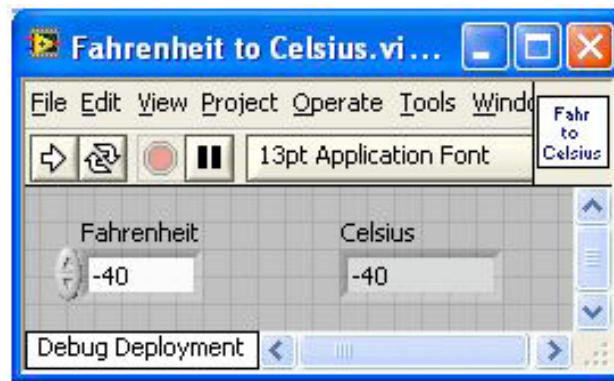


Figura 24. Ícone Editado

Feche a função **Fahrenheit to Celsius** e retorne à janela de programação do **Celsius**. O diagrama Celsius mostra o ícone Fahrenheit to Celsius atualizado.

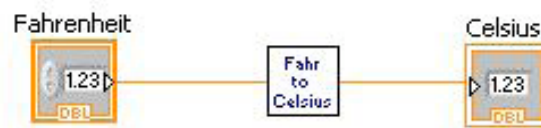


Figura 25. Chamado de Função

Seleção de Case

Este programa determina se o ano é ou não bissexto. Um ano bissexto é divisível por 4 mas não por 100, exceto quando divisível por 400. Um número x é divisível por um número y se o resto da divisão x/y é igual a zero.

Por exemplo:

1900 não é um ano bissexto porque ele é divisível por 100.

1970 não é um ano bissexto porque ele não é divisível por 4

1980 é um ano bissexto porque ele é divisível por 4 mas não por 100

2000 é um ano bissexto porque é divisível por 400.

Crie um novo programa G e clique com o botão direito do mouse na janela de programação. Vá para o menu **Functions>>Programming>>Numeric**

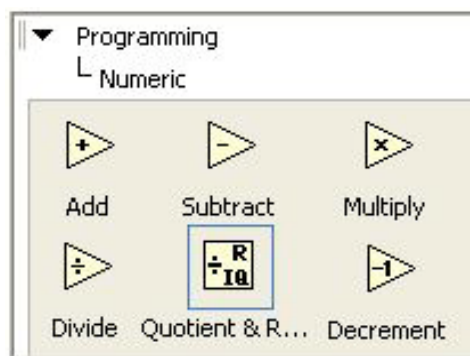


Figura 26. Função Quotient&Remainder

Selecione três cópias da função **Quotient & Remainder** e três constantes numéricas. Digite 4, 100, e 400 nas constantes numéricas e ligue estas constantes ao terminal inferior (correspondente ao dividendo) da função **Quotient & Remainder**.

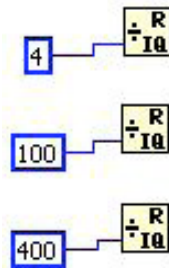


Figura 27. Constantes Numéricas ano Bissexto

Do menu **Functions >> Programming >> Comparison** selecione duas cópias da função **Equal to Zero** e uma cópia da função **Not Equal to Zero**.

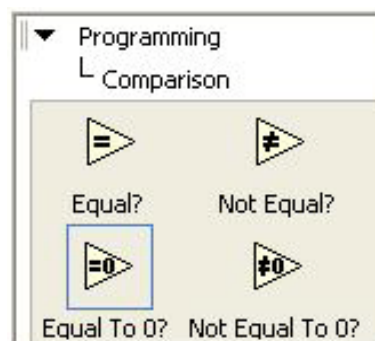


Figura 28. Funções de Comparação

Organize as operações da comparações como mostrado na figura 29.

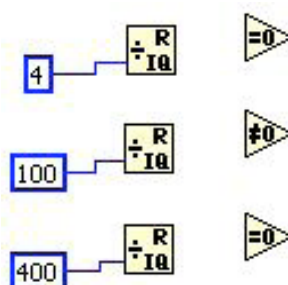


Figura 29. Comparação Q&R

Do menu **Functions >> Programming >> Boolean**, selecione as operações **AND** e **OR**.

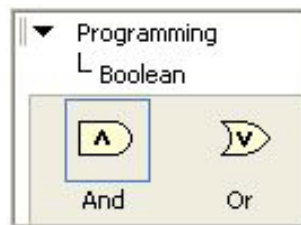


Figura 30. Operadores Booleanos

Coloque os operadores booleanos como mostrado na Figura 31.

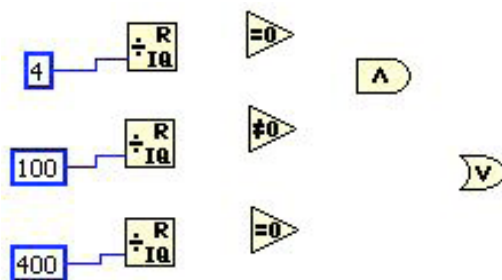


Figura 31. Comparação Q&R e Funções Booleanas

Do menu **Functions>>Programming>>Structures** clique na estrutura **Case**.

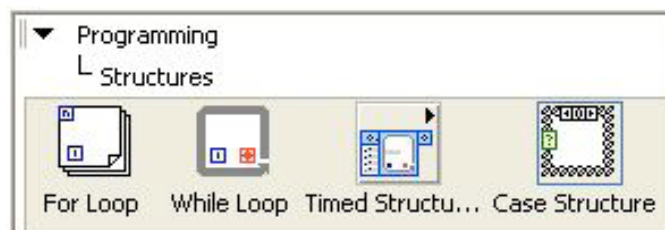


Figura 32. Estrutura Case

Clique e arraste na janela de programação para criar uma estrutura **Case**

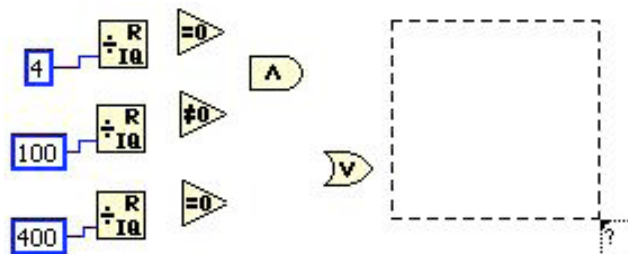


Figura 33. Criando uma Estrutura Case

A opção True está indicada no topo da estrutura case

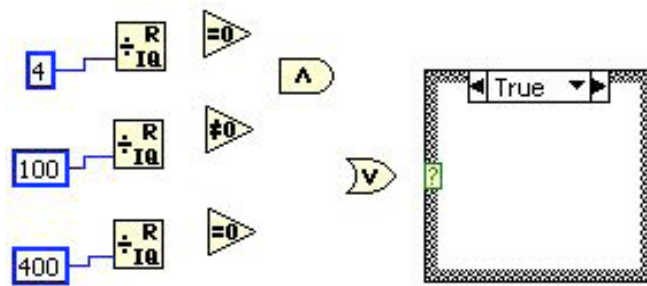


Figura 34. Estrutura Case

Coloque uma constante de string e digite “Is a Leap Year.”

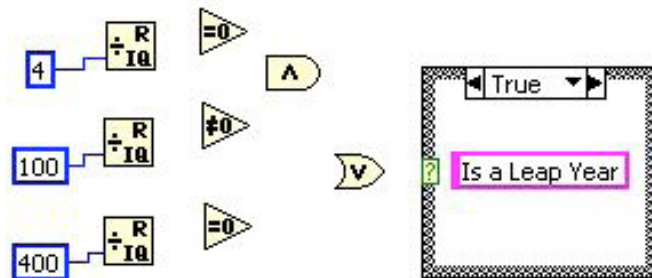


Figura 35. Edição do Caso True

Clique na seta para baixo ao lado do **True** e selecione a opção **False**

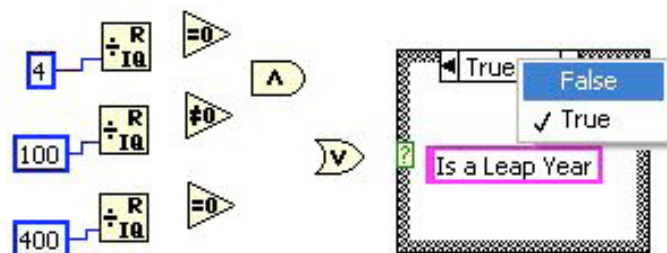


Figura 36. Selecionado o caso False

Coloque outra constante de string e digite “Is not a Leap Year.”

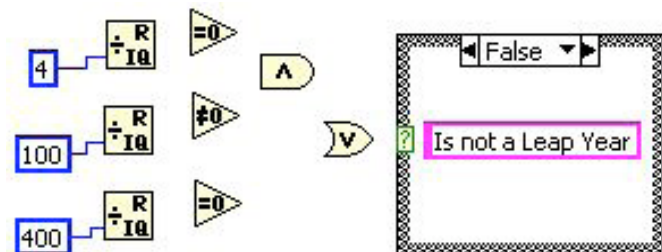


Figura 37. Editando o caso False

Vá à janela GUI e coloque uma entrada numérica e um string de saída. Renomei a entrada numérica para **Year** e o string de saída para **Message**.

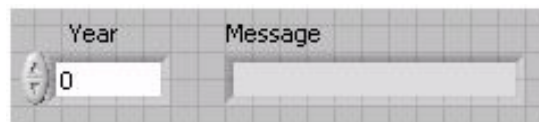


Figura 38. GUI Ano Bissexto

Clique com o botão direito do mouse em **Year** e selecione **Representation>>I32** no menu numérico.

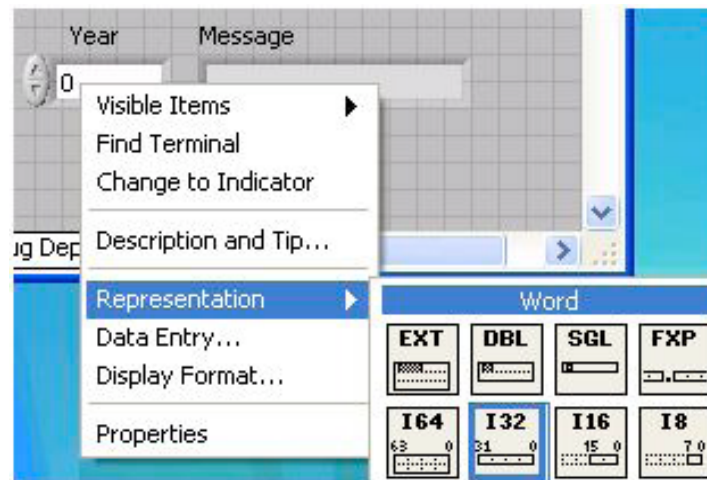


Figura 39. Inteiro de 32 Bits

Arrume os terminais **Year** e **Message** na janela de programação G como mostrado na figura 40

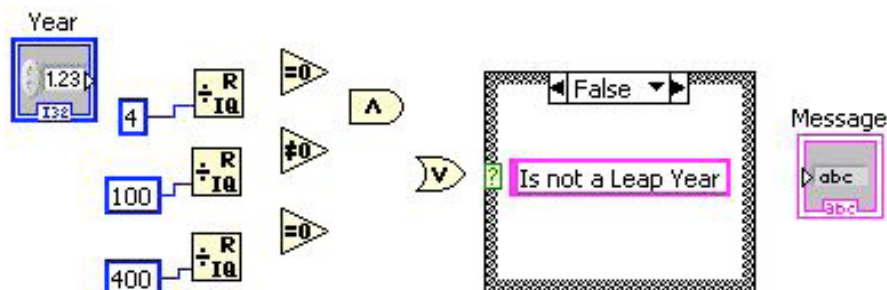


Figura 40. Diagrama sem a fiação

O operador **OR** é ligado ao “?” da estrutura case e a constante de string é ligado à **Message**

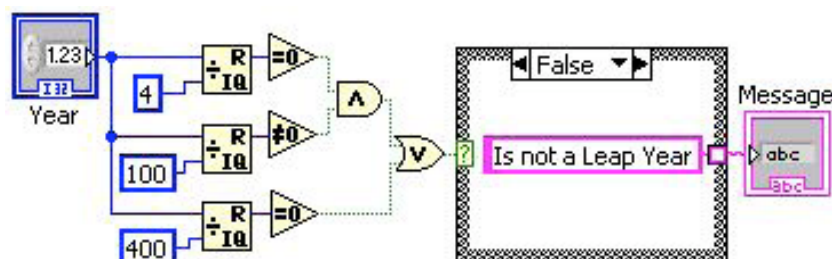


Figura 41. Caso “False”

Selecione a opção **True** e ligue a constante “Is a Leap Year” ao terminal de saída da estrutura **Case**.

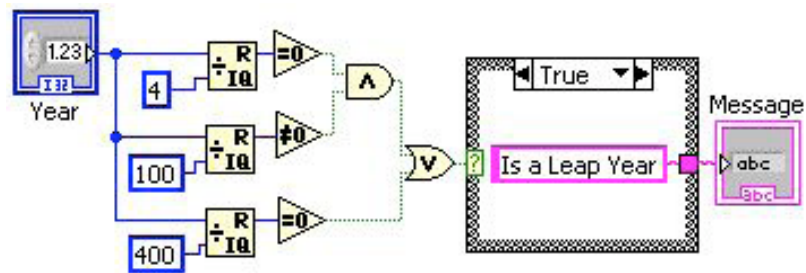


Figura 42. Caso “True”

Salve o programa como **Leap Year.vi**, digite valores em **Year**, e execute o programa para determinar se o valor de **Year** é ou não um ano bissexto.

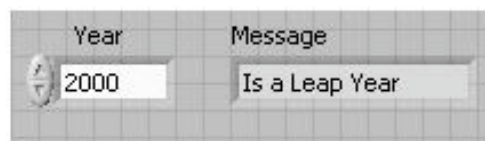


Figura 43. Programa Leap Year

Arrays

Clique com o botão direito do mouse na janela GUI e selecione **Array** no menu **Controls>>Modern>>Array, Matrix&Cluster**, coloque um array na janela GUI.

A estrutura array consiste em um índice ou offset de elementos (porção esquerda da estrutura) e os elementos do array (porção direita da estrutura). Quando a estrutura array é colocada na janela GUI, o tipo de dados do array é indefinido (indicado pela parte acinzentada do array).

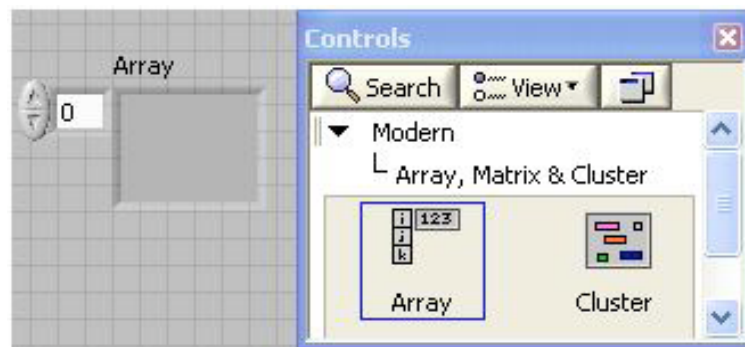


Figura 44. Arrays

Para definir o tipo de dados do array, arraste e solte dados dentro da estrutura. Por exemplo, para criar um array de números de entrada, coloque um **Numeric Control** dentro do array.

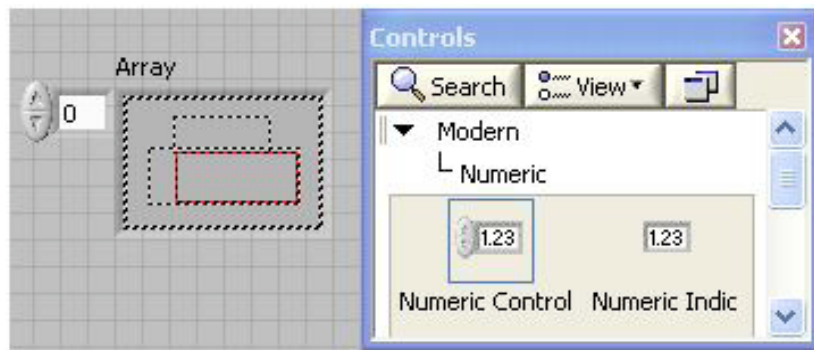


Figura 45. Criando um Array Numérico

Neste momento, o array é **Vazio ou Nulo** pois os elementos do array não foram definidos ainda. Isto é indicado pela cor acinzentada do controle numérico dentro da estrutura do array.

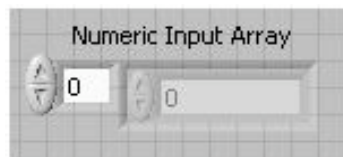


Figura 46. Array Numérico Vazio

Defina elementos de um array de entrada selecionando o offset e digitando valores. Por exemplo, em um offset =4 digite o valor 0.0. Isto define o array numérico como **{0, 0, 0,0, 0}**.

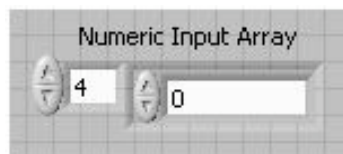


Figura 47. Definindo um array de elementos numéricos

Um array de saída é criado de forma análoga a um array de entrada com a exceção de que colocamos estruturas de saída no array.

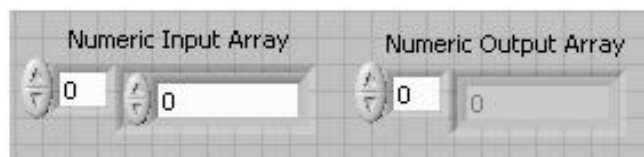


Figura 48. Criando um array de saídas numéricas

Loop For

Este programa converte um array de valores de Farenheit em Celsius. Crie um array de entrada e um de saída e nomeei-os de **Fahrenheit** e **Celsius**, respectivamente. No array **Fahrenheit**, digite os valores 0, 20, 40, 60, 80, 100, 120, 140, 160, 180, e 200 nos offsets 0 a 10 como mostrado na figura 49

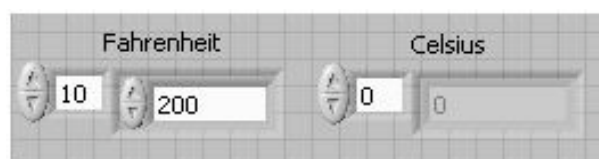


Figura 49. Arrays de Entradas e Saídas numéricas

Clique com o botão da direita do mouse na janela de programação G, navegue para **Programming>>Structures**, e clique no **For Loop**.

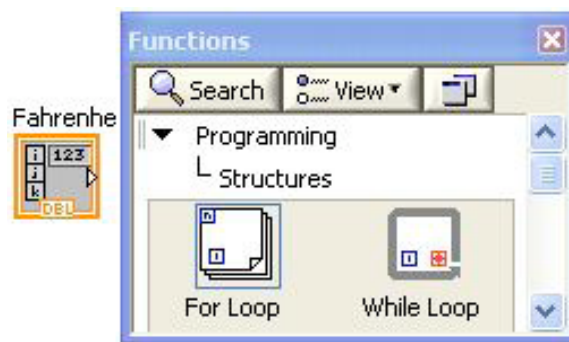


Figura 50. Estrutura do Loop For

Clique e arraste para criar um loop for como mostrado nas figuras 51 e 52

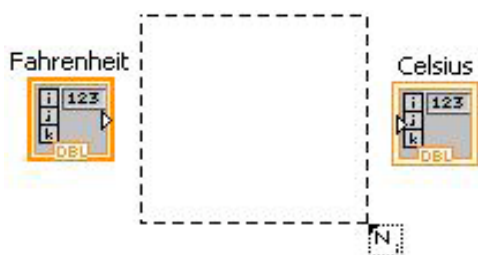


Figura 51. Criando um Loop For

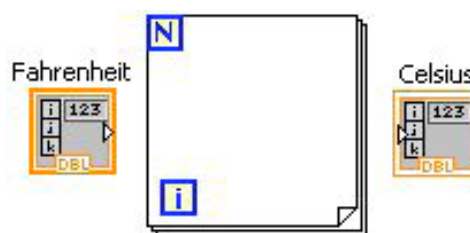


Figura 52. Loop For

Clique com o botão da direita do mouse no loop for e selecione **Select a VI...** do menu que aparece. Encontre o **Fahrenheit to Celsius.vi** e clique em **OK**. Coloque esta função dentro do Loop **For**.

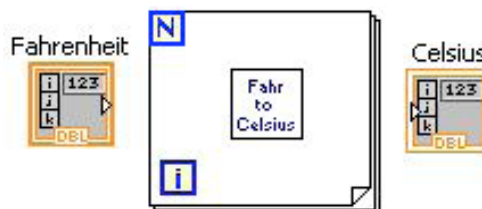


Figura 53. Figura dentro do Diagram

Para completar o programa, ligue o array de entrada **Fahrenheit** à função **Fahrenheit to Celsius** e ligue o terminal de saída da função ao array de saída **Celsius**

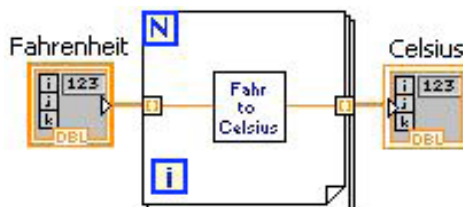


Figura 54. Função Ligada dentro do Diagrama

Este programa usa o loop For para selecionar cada elemento dentro do array de Fahrenheit, converter este valor para Celsius e salvar o resultado no array Celsius. Salve o programa como **Fahrenheit to Celsius For Loop.vi** e execute o programa



Figura 55. Arrays de Farenheite para Celsius

O array de saída **Celsius** contém os seguintes valores: **Celsius** = {-17.7778, -6.6667, 4.44444, 15.5556, 26.6667, 37.7778, 48.8889, 60, 71.1111, 82.2222 e 93.3333}

Loop While

O próximo programa gera valores de Fahrenheit e converte estes para Celsius até que uma condição de parada seja satisfeita em um **loop while**. Na janela de programação G, selecione a estrutura do **loop while** clicando nela no menu **Functions>>Programming>>Structures**.

Clique e arraste para criar uma estrutura de um **loop while**

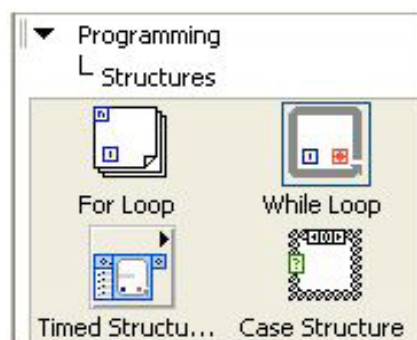


Figura 56. Estrutura do Loop While

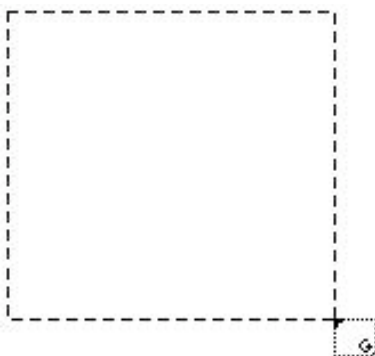


Figura 57. Criando um Loop While

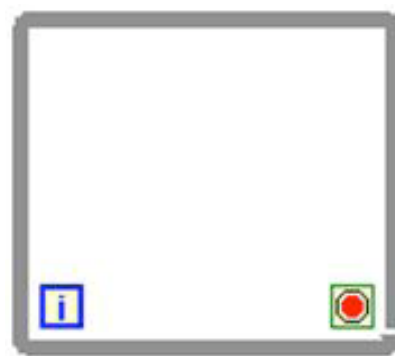


Figura 58. Loop While

Na janela GUI, crie dois arrays numéricos de saída. Nomeei os **Fahrenheit** e **Celsius**

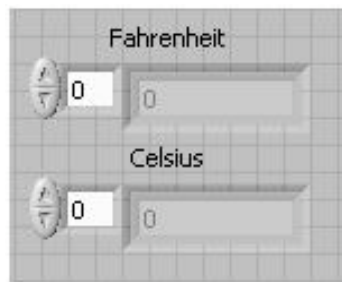


Figura 59. Array Numérico de Saída

Reordene o diagrama como mostrado na figura 60



Figura 60. Diagrama do Loop While

No menu **Functions**, selecione a função **Multiply** e duas constantes numéricas. Digite **20.0** e **300.0** para o valor das constantes numéricas. Selecione o **Fahrenheit to Celsius.vi** e coloque-o dentro do loop while. Reordene o diagrama para se parecer como o da figura 61

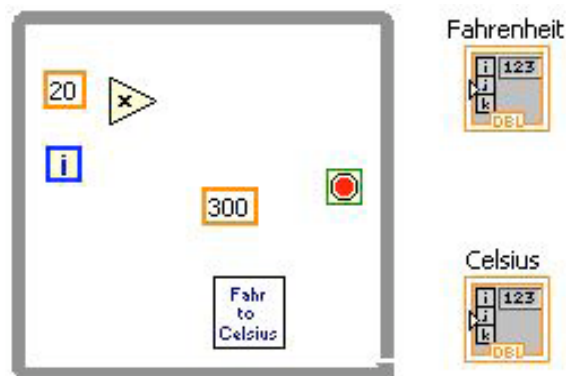


Figura 61. Gerando Valores de Fahrenheit

Do menu **Functions>>Programming>>Comparison** selecione o operador **Greater or Equal**.

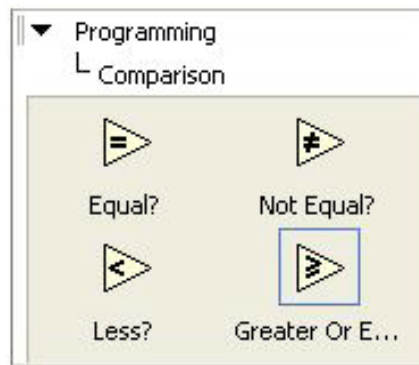


Figura 62. Função Maior ou Igual

Ligue os componentes do **loop while** como mostrados na figura 63

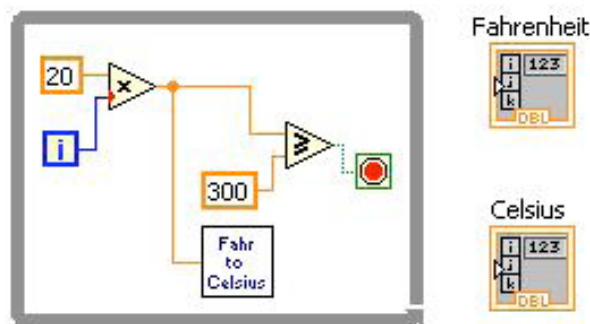


Figura 63. Gerando Valores de Fahrenheit e Condição de Parada

Ligue o saída da operação **Multiply** ao array **Fahrenheit** e saída à função **Fahrenheit to Celsius** ao array **Celsius**. Veja que as conexões entre o loop while e os arrays estão quebradas (veja figura 64)

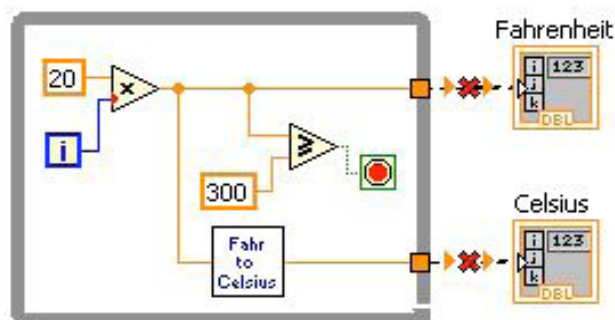


Figura 64. Fios Quebrados

Para reparar a conexão quebrada, passe o mouse por cima do **loop tunnel**

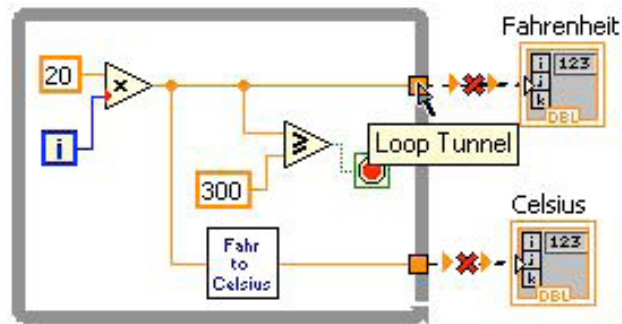


Figura 65. Loop Tunnel

Clique com o botão direito do mouse no **loop tunnel** e selecione **Enable Indexing** no menu que aparece.

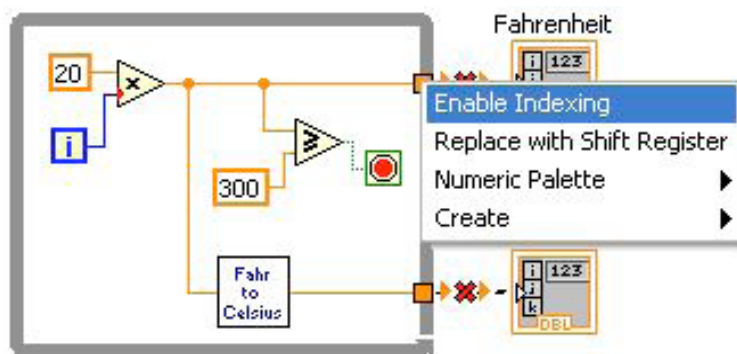


Figura 66 Habilitando o Loop Indexing

Isto permite que valores se acumulem e que seus resultados sejam armazenados em um array.

Repita o processo para o array **Celsius**

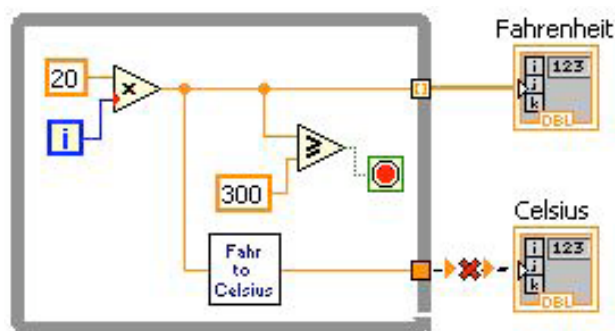


Figura 67. Fio Quebrado consertado

Cada interação do **loop while** gera $i \cdot 20$ como valor de Fahrenheit e converte este valor para Celsius. O **loop while** para quando o valor para Fahrenheit for maior ou igual a 300. Os arrays resultantes são armazenados nos arrays numéricos de saída **Fahrenheit** e **Celsius**.

Salve o programa como **Fahrenheit to Celsius While Loop.vi** e execute-o. O programa gera os seguintes resultados:

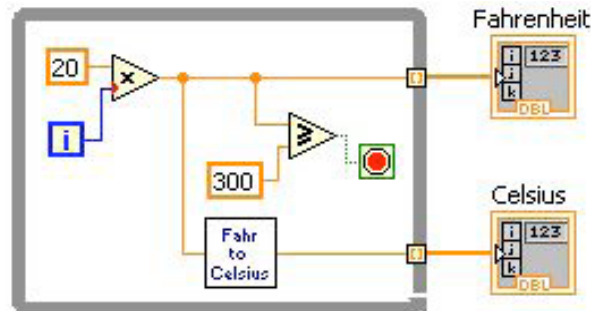


Figura 68 Loop While Fahrenheit para Celsius

Fahrenheit = {0, 20, 40, 60, 80, 100, 120, 140, 160, 180, 200, 240, 260, 280, 300}

Celsius = {-17.7778, -6.6667, 4.44444, 15.5556, 26.6667, 37.7778, 48.8889, 60, 71.1111, 82.2222, 93.3333, 104.444, 115.556, 126.667, 137.778, 148.889}

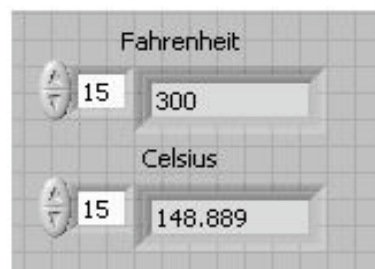


Figura 69. Arrays Fahrenheit e Celsius

Gráficos

Usando o último programa como exemplo, veja os resultados acrescentando gráficos à janela GUI. Clique com o botão direito do mouse na janela GUI. Selecione **XY Graph** do menu **Controls>>Modern>>Graph**.



Figura 70. Seleção de Gráfico XY

Coloque o **XY Graph** na janela GUI. Clique duas vezes no eixo x e y para renomear **Time** para **Fahrenheit** e **Amplitude** para **Celsius**

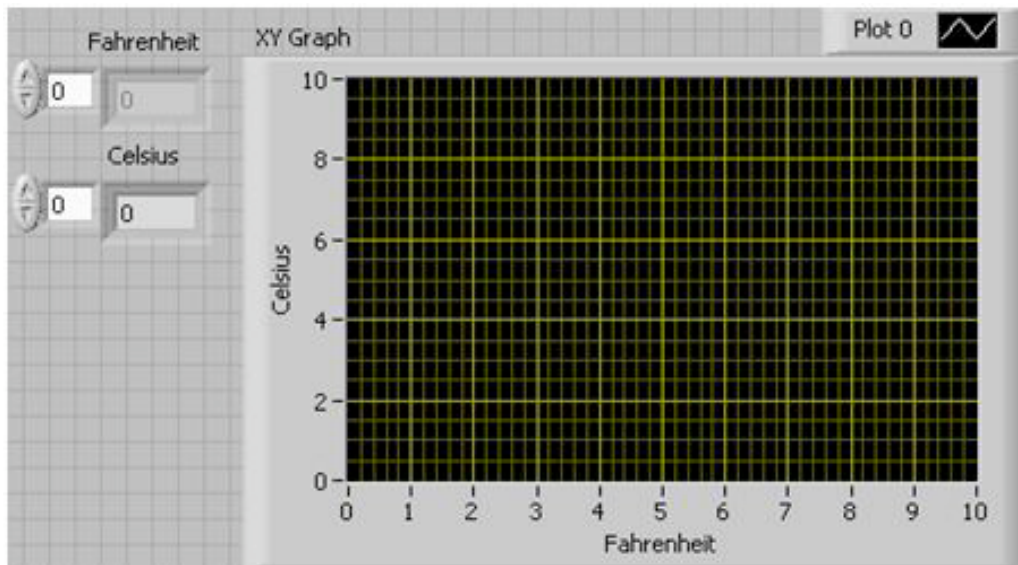


Figura 71. Gráfico XY na Janela GUI

A janela de programação G contém, agora, um terminal para o gráfico XY

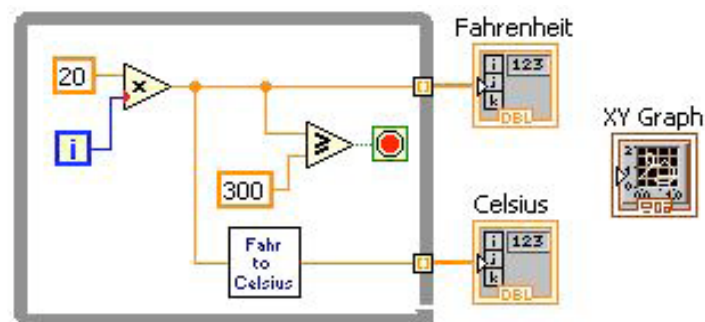


Figura 72. Terminal do Gráfico XY no Diagrama

Selecione **Bundle** do menu **Functions>>Programming>>Cluster, Class&Variant**

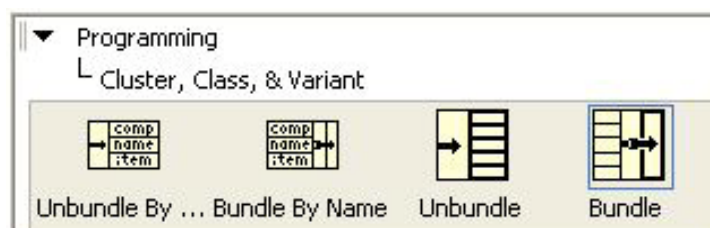


Figura 73. Operador Bundle

Coloque o Bundle no diagrama como mostrado na figura 74

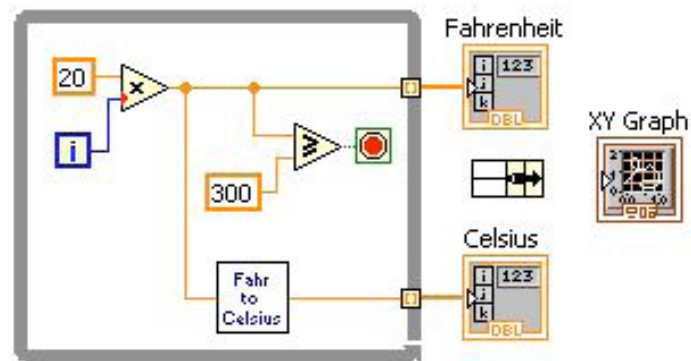


Figura 74. Bundle para Gráfico XY

Ligue os resultados do **Fahrenheit** e **Celsius** aos terminas de entrada do **Bundle** e ligue a saída ao **XY Graph**.

Salve o programa e execute-o. O gráfico resultante esta na figura 76

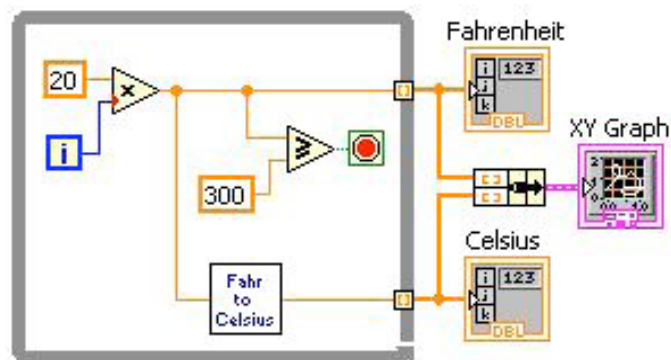


Figura 75. Gráfico XY Ligado

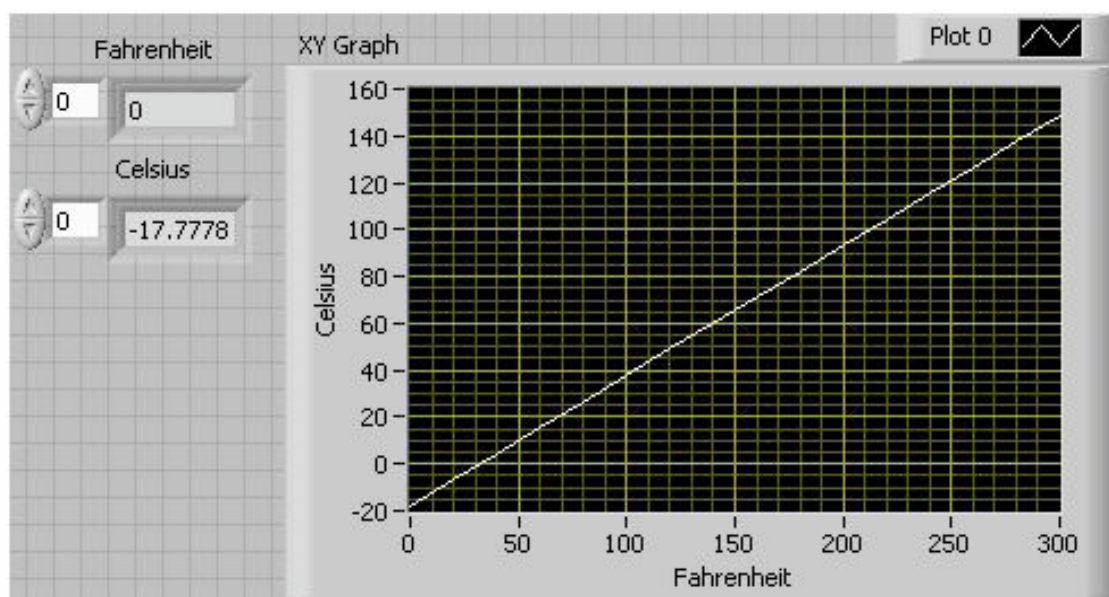


Figura 76. Resultados Gráfico XY

Interatividade

Este programa mostra como usar a linguagem G para desenvolver programas interativos. Crie o seguinte programa em G e ligue os componentes como na figura 77

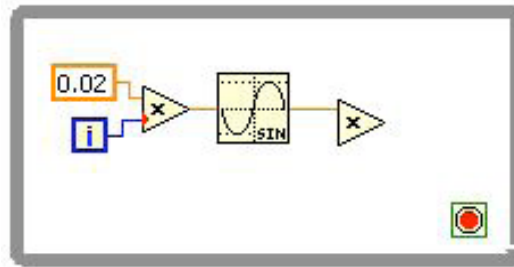


Figura 77. Criando Programas Interativos

Na janela GUI, selecione um slide vertical com do menu **Functions>>Modern>>Numeric**. Selecione um **Waveform Chart** do menu **Functions>>Modern>>Graph**.

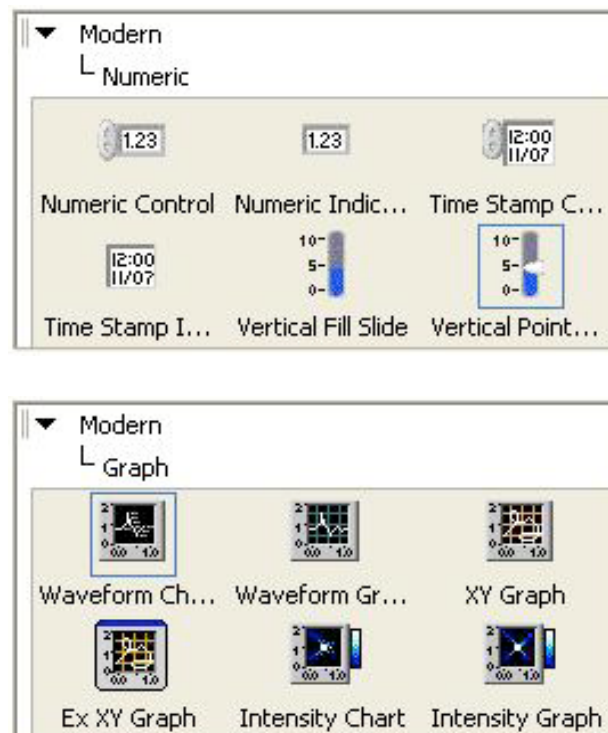


Figura 78. Slide Vertical com ponteiro e Waveform Graph

Renomei o slide como **Amplitude** e o gráfico como **Sine Wave**. Reordene o GUI para se parecer como o da figura 79

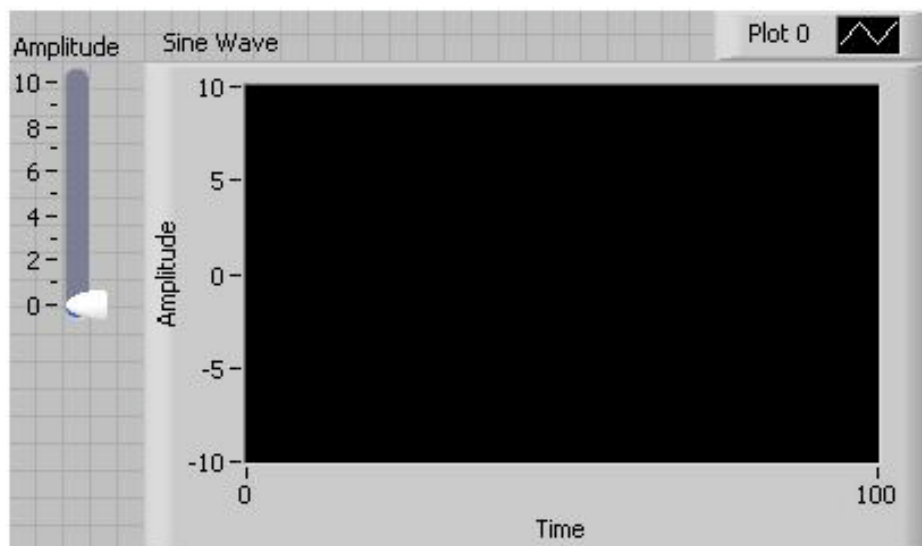


Figura 79. Slide e Waveform Char na janela GUI

Clique com o botão direito do mouse em **Sine Wave** e selecione **Properties** no menu pop up.

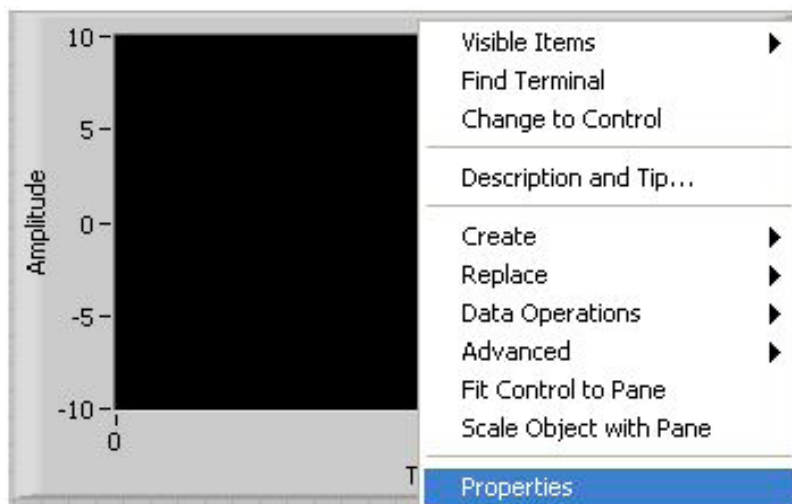


Figura 80. Selecionado as Propriedades do Gráfico

Selecione a aba **Scales** e altere o valor do **Maximum** para 1023. O **Sine Wave** mostrará 1024 samples

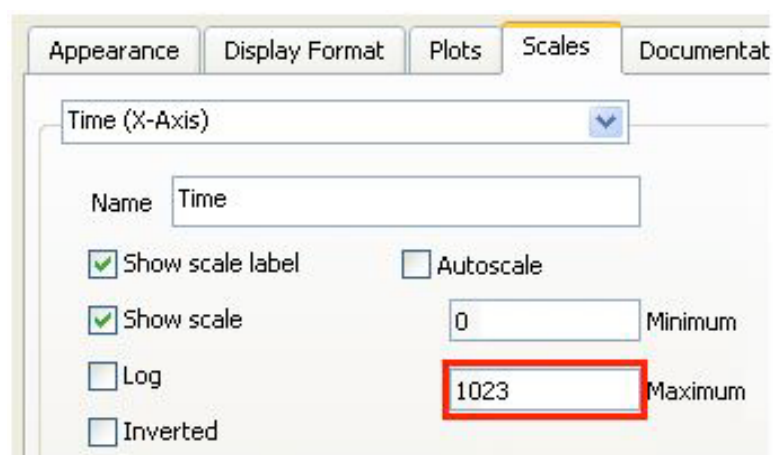


Figura 81. Máximo do Eixo X

Clique na seta para baixo localizada à direita do **Time (X-Axis)** e selecione **Amplitude (Y-Axis)**.

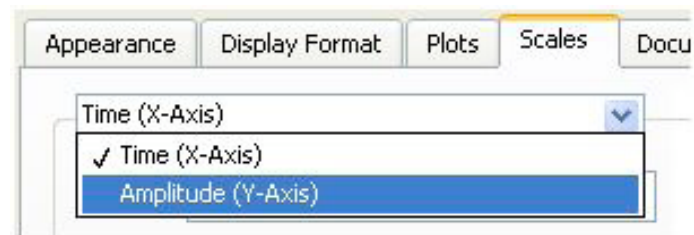


Figura 82. Selecionando o eixo Y

Des-selecione o **Autoscale** e mude os valores de **Minimum** e **Maximum** para **-10** e **10**. Clique em **OK**.

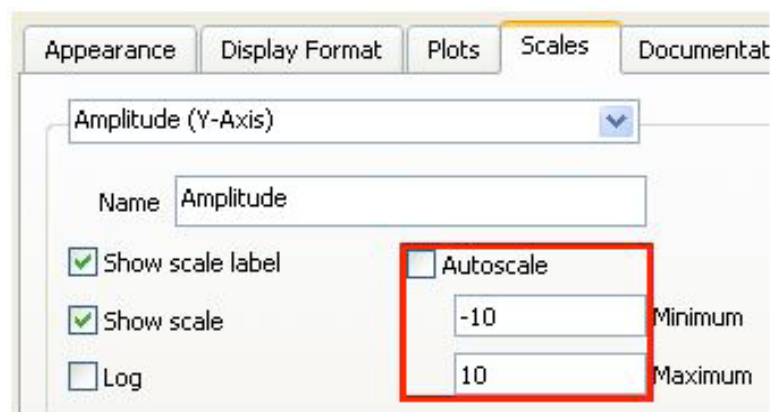


Figura 83. Desativando o Autoscale

Na janela de programação, reordene os terminais **Amplitude** e **Sine Wave** e termine o programa como mostrado na figura 84.

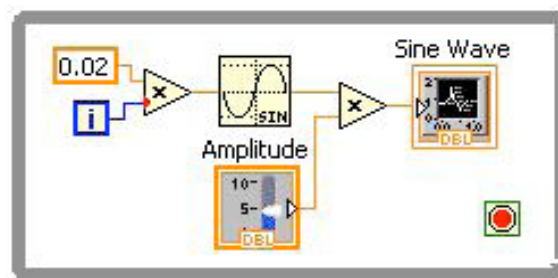


Figura 84. Diagrama Interativo de uma onda senoidal

Passe o ponteiro do mouse por cima do **Loop Condition**.

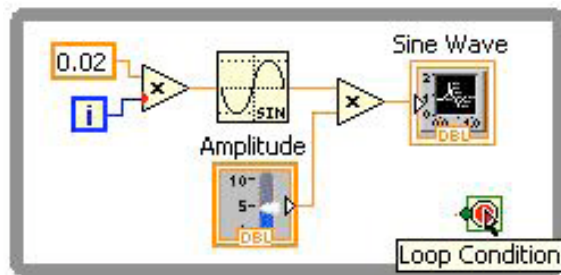


Figura 85. Condição do Loop

Clique com o botão direito do mouse no **Loop Condition** e selecione **Create Control** no menu pop-up

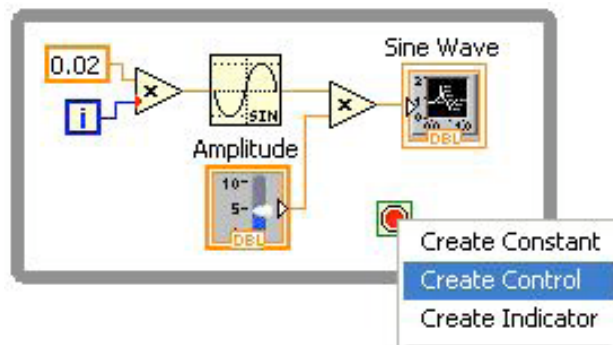


Figura 86. Criar um Controle de Loop

Um terminal de parada é criado...

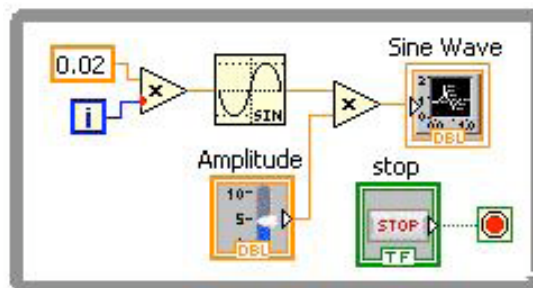


Figura 87. Programa Interativo

... com um botão Booleano **stop** correspondente na janela GUI. Salve o programa como **Interactivity.vi**.

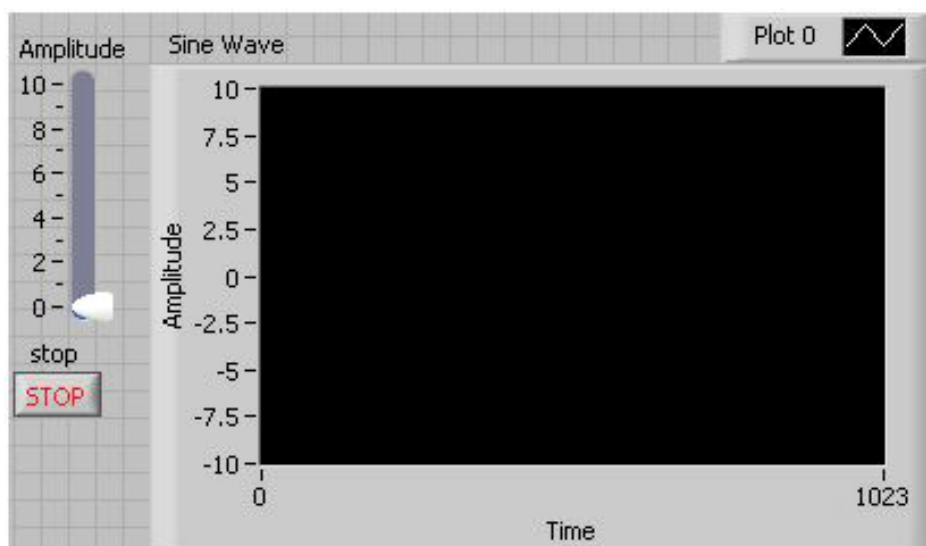


Figura 88. Programa Interativo

Execute o programa

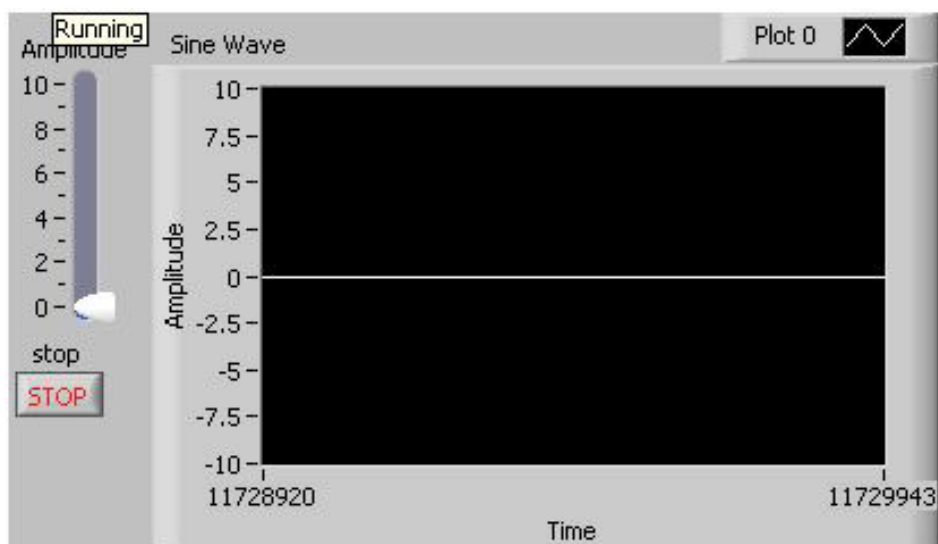


Figura 89. Programa Interativo

Enquanto o programa estiver sendo executado, altere o valor da **Amplitude** e veja o gráfico se atualizar para refletir as mudanças interativas.

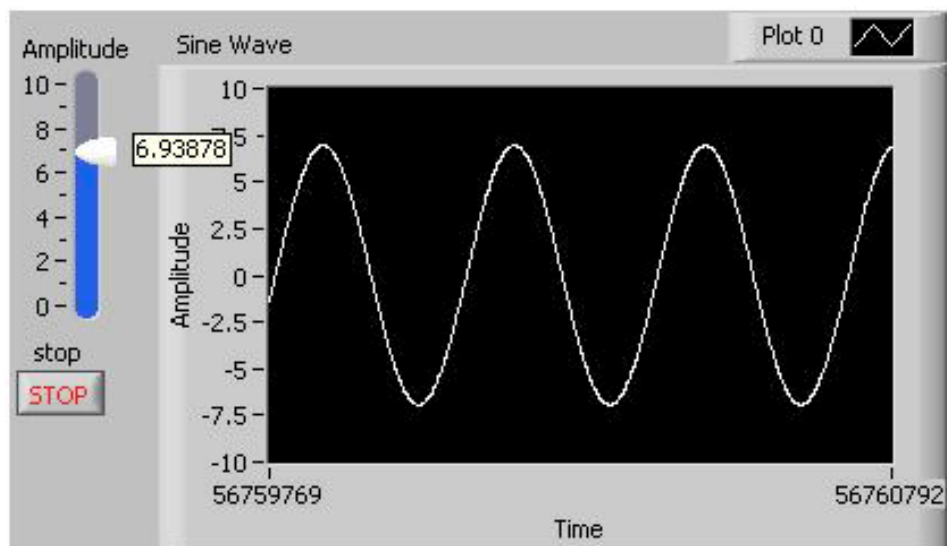


Figura 90. Programa Interativo

Para finalizar o programa, simplesmente clique no botão **stop**.

Parabéns! Você criou e executou seu primeiro programa interativo com linguagem.

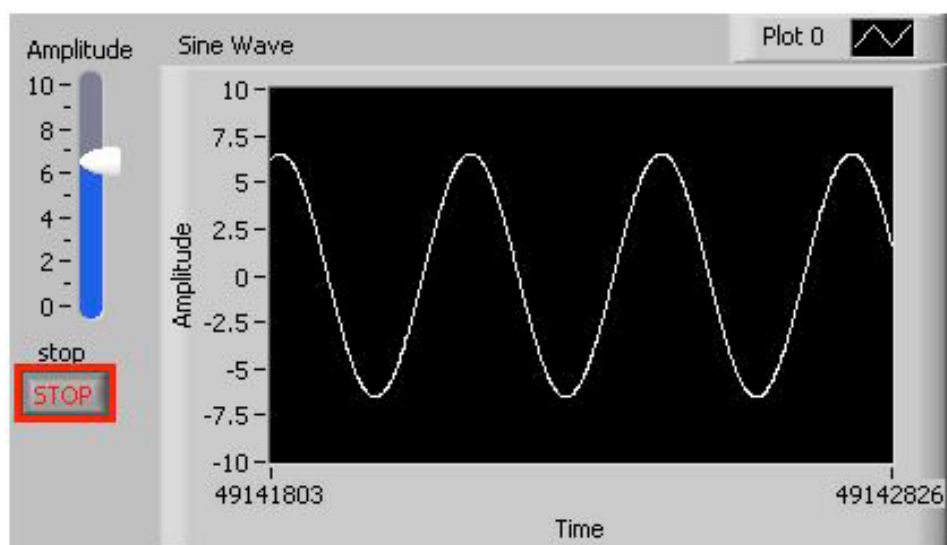


Figura 91. Programa Interativo

Programação Paralela

Salve uma cópia do **Interactivity.vi** como **Parallel Programming.vi**. Selecione o loop while como mostrado na figura 92

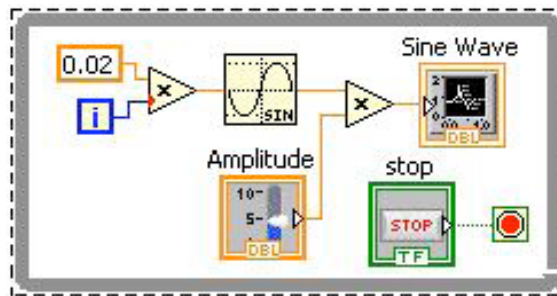


Figura 92. Selecione o Diagrama para programação paralela

No menu, selecione **Edit>>Copy**

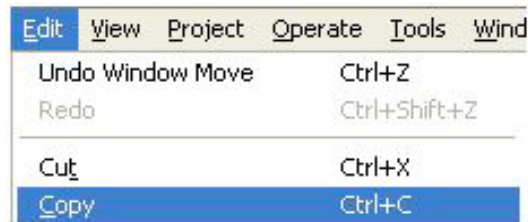


Figura 93. Copie Diagrama Selecionado

Crie uma cópia do loop while e seu conteúdo selecionando **Edit>>Paste**. Organize seu diagrama como o diagrama da figura 94.

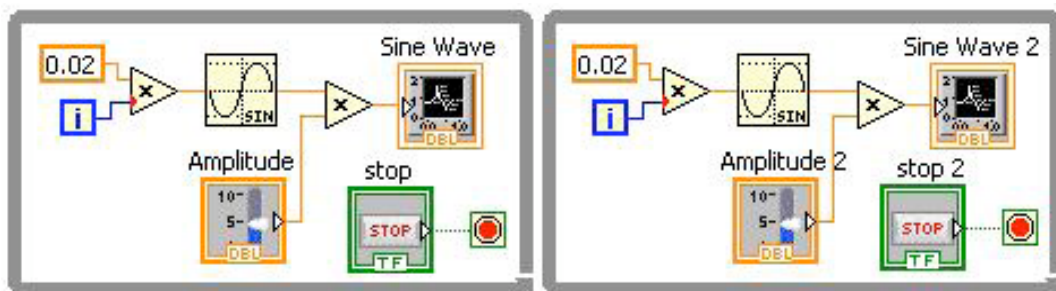


Figura 94. Cole o Diagrama

Vá à janela GUI e organize os controles de entrada e saída como mostrado na figura 95.

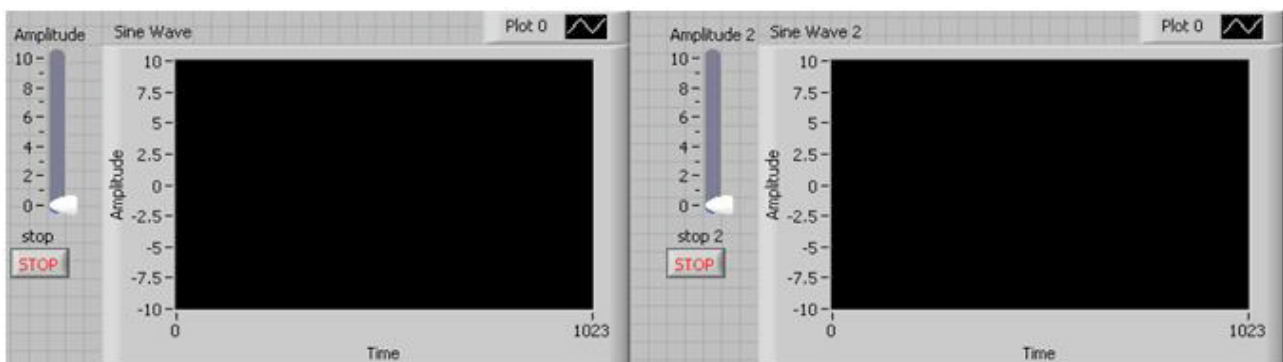


Figura 95. Programa Paralelo

Parabéns! Você criou seu primeiro programa interativo paralelo usando a linguagem G. Salve este programa, execute-o, e interaja com ele. Para finalizar este programa, clique nos botões **stop** e **stop**

2.

Programação Multicore

Salve uma cópia do **Parallel Programming.vi** como **Multicore Programming.vi**. Se você estiver usando um computador com tecnologia multicore, parabéns! Você acaba de criar seu primeiro programa paralelo interativo e multicore com linguagem G.

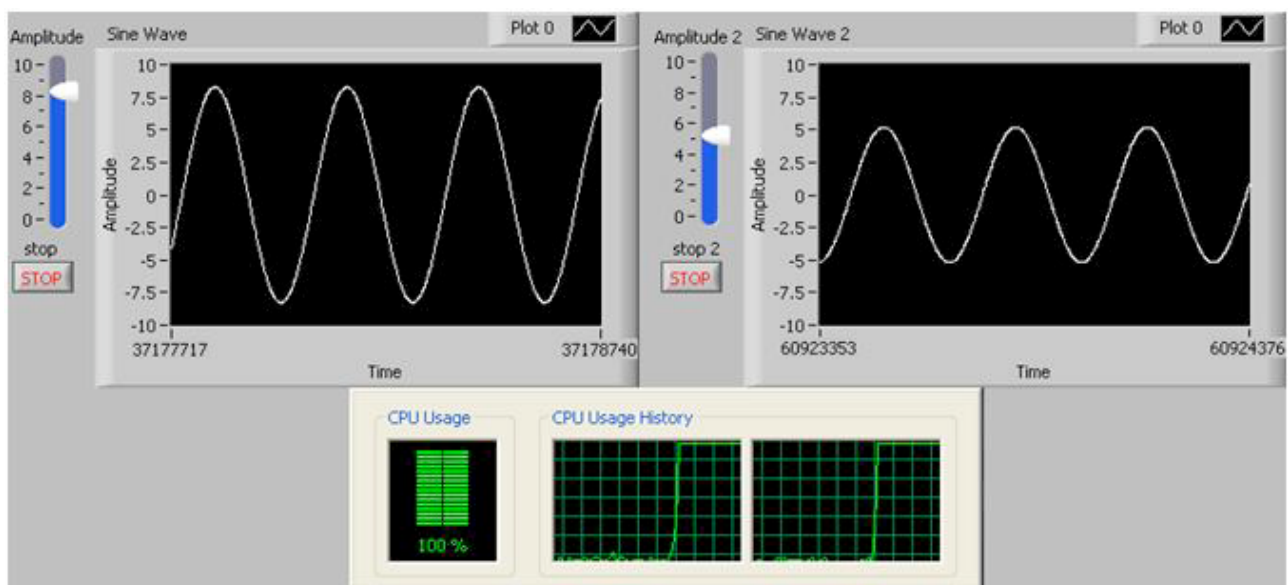


Figura 97. Programa Interativo Multicore

Polimorfismo

Este programa mostra as propriedades polimórficas da linguagem G. Repare que as operações **Subtract** e **Multiply** permitem que arrays sejam ligadas ao programa em linguagem G.

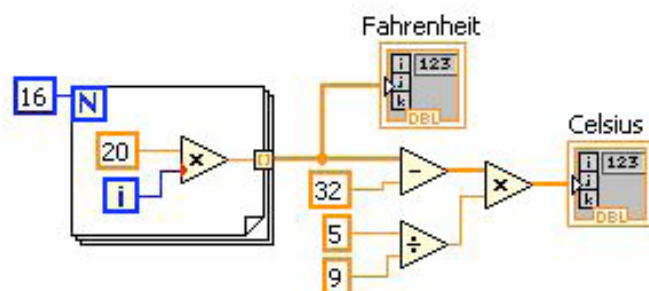


Figura 98. Diagrama Polimórfico

Links Relacionados

[Scientific Computing with Graphical System Design](#)

[Scientific Computing with NI LabVIEW](#)

[Multicore Programming](#)

[Visualization and Imaging](#)

Introduction to Curve Fitting
Statistical Analysis
Scripting Languages and NI LabVIEW