



**Innovative Teaching Approaches
in development of Software Designed
Instrumentation and its application
in real-time systems**

Podstawy Projektowania Przyrządów Wirtualnych **Wykład 3: struktury programistyczne, obsługa błędów.**

Co-funded by the
Erasmus+ Programme
of the European Union



Powtórka z wykładu 2

1/10 Która z akcji mechanicznych wykonana na kontrolce Boolean spowoduje zmianę stanu z False na True od momentu kliknięcia na kontrolkę aż do momentu, w którym LabVIEW będzie odczytywać wartość.

- a) Switch Until Released
- b) Switch When Released
- c) Latch Until Released
- d) Latch When Released

Powtórka z wykładu 2

1/10 Która z akcji mechanicznych wykonana na kontrolce Boolean spowoduje zmianę stanu z False na True od momentu kliknięcia na kontrolkę aż do momentu, w którym LabVIEW będzie odczytywać wartość.

- a) Switch Until Released
- b) Switch When Released
- c) Latch Until Released
- d) **Latch When Released**

Powtórka z wykładu 22

2/10 Która z funkcji zamienia stałą zmiennoprzecinkową na łańcuch znaków:

- a) Format value
- b) Scan value
- c) Number to string
- d) String to number

Powtórka z wykładu 2

2/10 Która z funkcji zamienia stałą zmiennoprzecinkową na łańcuch znaków:

- a) **Format value**
- b) Scan value
- c) Number to string
- d) String to number

Powtórka z wykładu 2

3/10 Które tryby wyświetlania tekstu są dostępne dla łańcuchów znaków na Front Panel?

- a) \Codes
- b) Password
- c) Hex
- d) Wszystkie powyższe

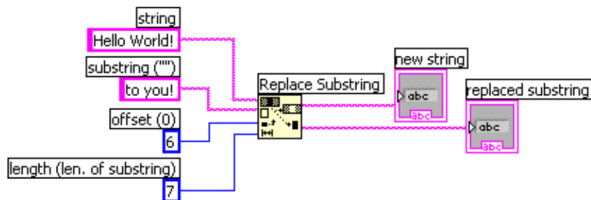
Powtórka z wykładu 2

3/10 Które tryby wyświetlania tekstu są dostępne dla łańcuchów znaków na Front Panel?

- a) \Codes
- b) Password
- c) Hex
- d) **Wszystkie powyższe**

Powtórka z wykładu 2

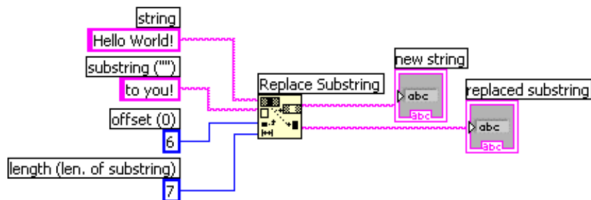
4/10 Jaki wynik będzie wyświetlony w zmiennej new string po uruchomieniu programu?



- a) Hello to you!
- b) Hello Wto you!
- c) Hello to you!!
- d) Helloto you!

Powtórka z wykładu 2

4/10 Jaki wynik będzie wyświetlony w zmiennej new string po uruchomieniu programu?



- a) **Hello to you!**
- b) Hello Wto you!
- c) Hello to you!!
- d) Helloto you!

Powtórka z wykładu 2

5/10 Którego elementu nie można stworzyć:

- a) Klaster tablic
- b) Tablica klastrów
- c) Tablica tablic
- d) Klaster klastrów

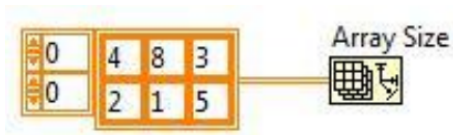
Powtórka z wykładu 2

5/10 Którego elementu nie można stworzyć:

- a) Klaster tablic
- b) Tablica klastrów
- c) **Tablica tablic**
- d) Klaster klastrów

Powtórka z wykładu 2

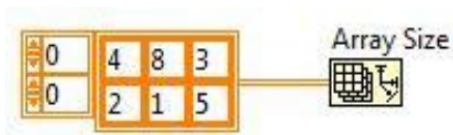
6/10 Jaką wartość zwróci funkcja **Array Size** z poniższego rysunku:



- a) tablica jednowymiarowa: {2,3}
- b) tablica jednowymiarowa: {3,2}
- c) 2
- d) Żadna z powyższych

Powtórka z wykładu 2

6/10 Jaką wartość zwróci funkcja **Array Size** z poniższego rysunku:



- a) **tablica jednowymiarowa: {2,3}**
- b) tablica jednowymiarowa: {3,2}
- c) 2
- d) Żadna z powyższych

Powtórka z wykładu 2

7/10 Które z poniższych zdań nie jest prawdziwe?

- a) Indeksowanie jest używane aby dostać się do wybranego elementu w tablicy
- b) Tablica jest indeksowana od 1 do n
- c) Dwuwymiarowe tablice posiadają oddzielne indeksowanie dla kolumn i wierszy
- d) Tablica jest indeksowana od 0 do n-1

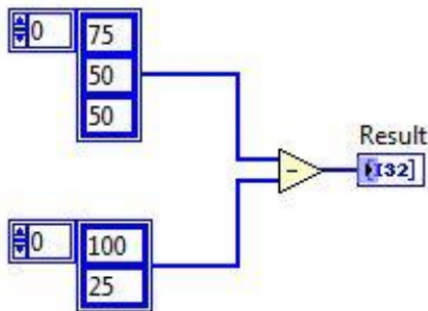
Powtórka z wykładu 2

7/10 Które z poniższych zdań nie jest prawdziwe?

- a) Indeksowanie jest używane aby dostać się do wybranego elementu w tablicy
- b) **Tablica jest indeksowana od 1 do n**
- c) Dwuwymiarowe tablice posiadają oddzielne indeksowanie dla kolumn i wierszy
- d) Tablica jest indeksowana od 0 do n-1

Powtórka z wykładu 2

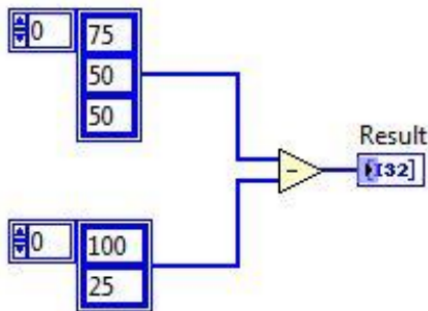
8/10 Jaka wartość zostanie wyświetlona w zmiennej Result?



- a) tablica jednowymiarowa $\{-25, 25, 50\}$
- b) tablica jednowymiarowa: $\{-25, 25\}$
- c) tablica dwuwymiarowa: $\{\{-25, -50, -50\}, \{50, 25, 25\}\}$
- d) tablica jednowymiarowa: $\{75, -50, 25\}$

Powtórka z wykładu 2

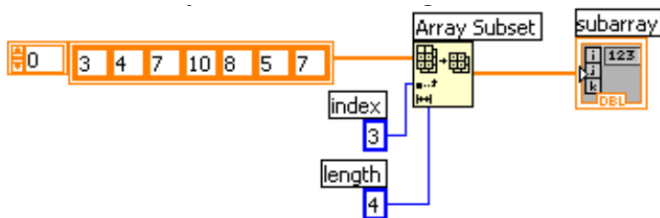
8/10 Jaka wartość zostanie wyświetlona w zmiennej Result?



- a) tablica jednowymiarowa $\{-25, 25, 50\}$
- b) **tablica jednowymiarowa: $\{-25, 25\}$**
- c) tablica dwuwymiarowa: $\{\{-25, -50, -50\}, \{50, 25, 25\}\}$
- d) tablica jednowymiarowa: $\{75, -50, 25\}$

Powtórka z wykładu 2

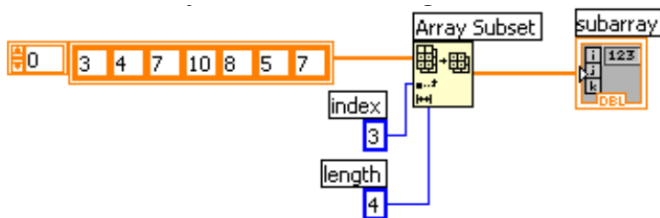
9/10 Jaki wynik będzie wyświetlony w zmiennej subarray?



- a) tablica jednowymiarowa: {8,5,7}
- b) tablica jednowymiarowa: {10,8,5}
- c) tablica jednowymiarowa: {10,8,5,7}
- d) tablica jednowymiarowa: {7,10,8,5}

Powtórka z wykładu 2

9/10 Jaki wynik będzie wyświetlony w zmiennej subarray?



- a) tablica jednowymiarowa: {8,5,7}
- b) tablica jednowymiarowa: {10,8,5}
- c) **tablica jednowymiarowa: {10,8,5,7}**
- d) tablica jednowymiarowa: {7,10,8,5}

Powtórka z wykładu 2

10/10 Jaka jest funkcja klastra?

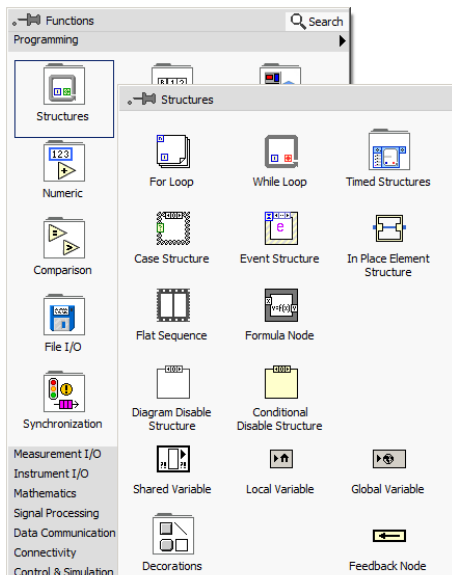
- a) Grupowanie różnych typów danych w logiczną strukturę.
- b) Prezentowanie danych na Front Panel przy użyciu wykresu.
- c) Separowanie danych różnego typu na Front Panel.

Powtórka z wykładu 2

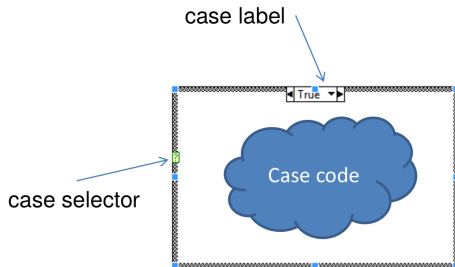
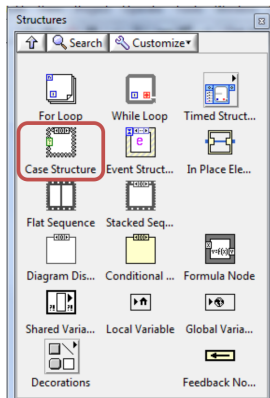
10/10 Jaka jest funkcja klastra?

- a) **Grupowanie różnych typów danych w logiczną strukturę.**
- b) Prezentowanie danych na Front Panel przy użyciu wykresu.
- c) Separowanie danych różnego typu na Front Panel.

Struktury programistyczne

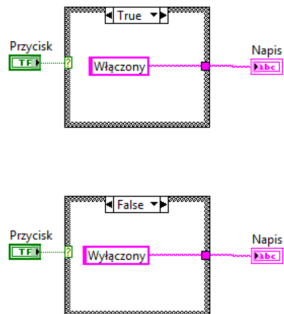
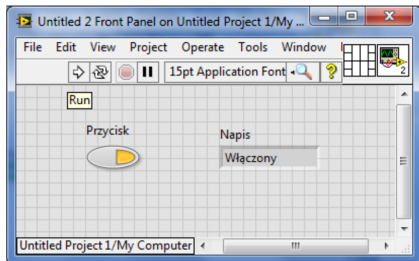


Struktura Case

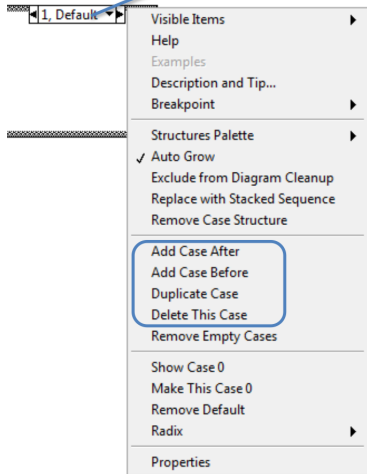


if($X == A$) wykonaj przypadek dla A
if($X == B$) wykonaj przypadek dla B
if($X == C$) wykonaj przypadek dla C

Struktura Case



Struktura Case

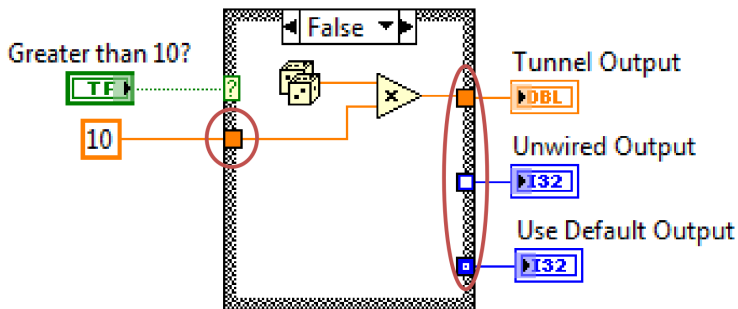


Możliwe wartości:

- ..5, 6, 7..10, 12, 13, 14
- True, False
- Default
- „Red”, „Green”

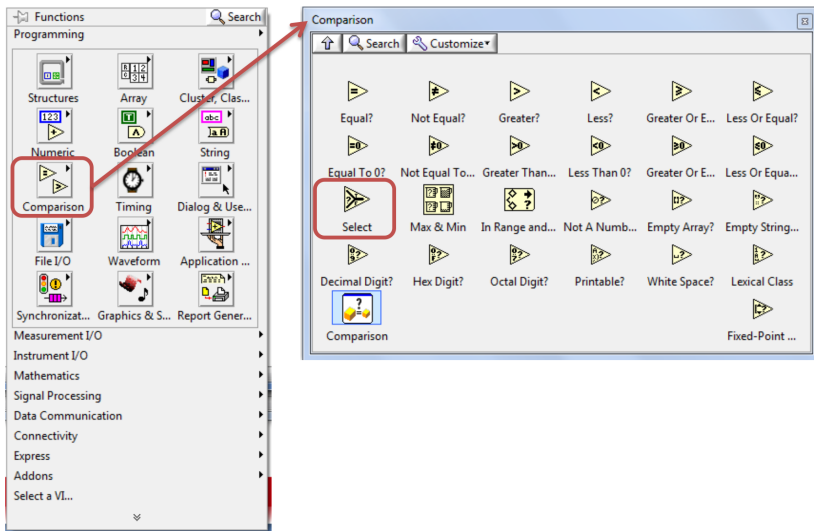
Tunele

Tunele to wejścia/wyjścia drutów ze struktur

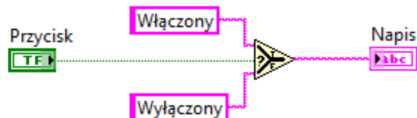
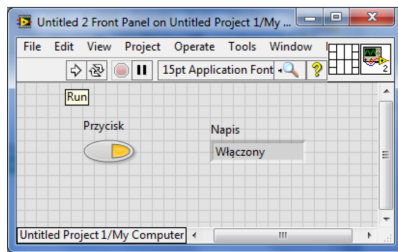


W przypadku struktury warunkowej Case wyjście musi być podłączone dla każdego przypadku. W razie nie podłączenia drutu możliwe jest wykorzystanie wartości domyślnej danego typu danych (**use default if unwired**).

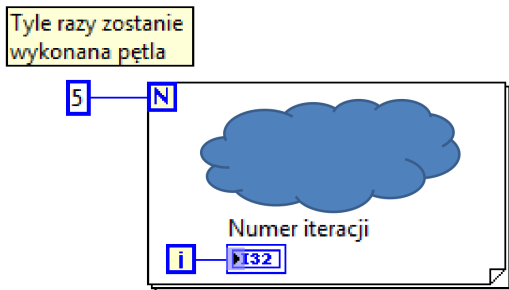
Funkcja select



Funkcja select



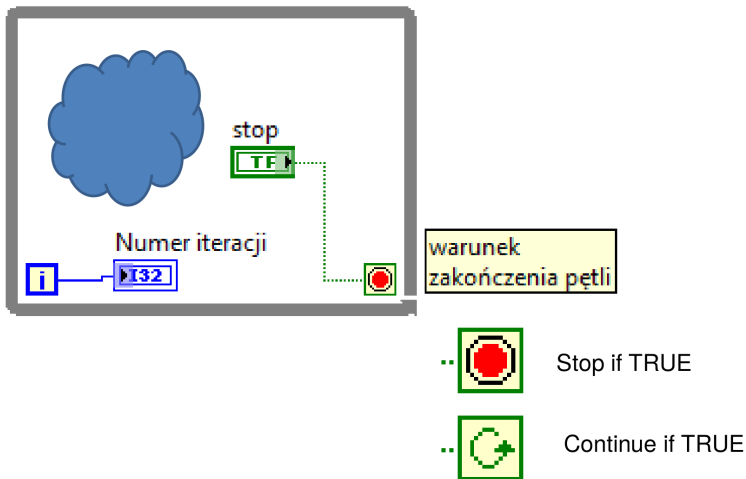
Pętla For



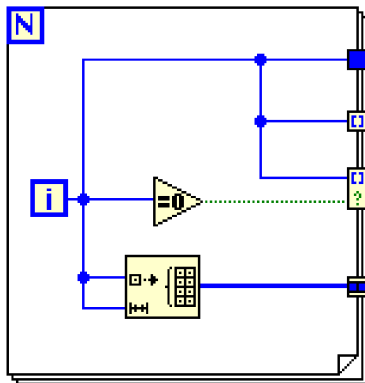
Pętle iterują się od 0!

Po uruchomieniu powyższego kodu, otrzymamy liczby od 0 do 4 w indykatorze: Numer iteracji.

Pętla While



Tunele



Last value

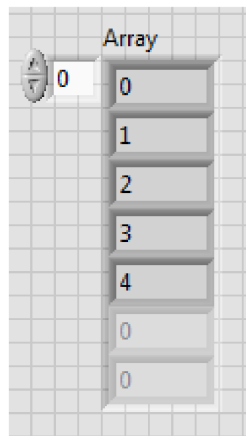
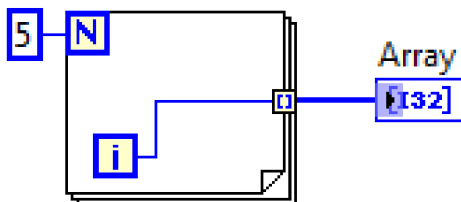
Indexing

Conditional

Concatenating

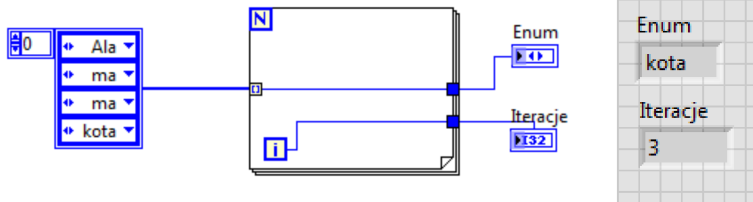
W pętli for domyślnie tunele wstawiane są w trybie indexing a w pętli while w trybie last value

Indeksowanie



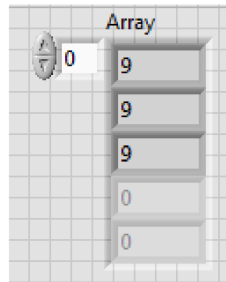
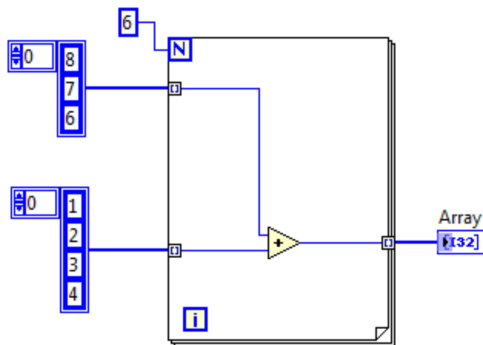
Indeksowanie

W przypadku wejściowych tuneli indeksujących pętla wykona się tyle razy ile jest elementów w tablicy.

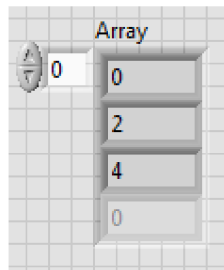
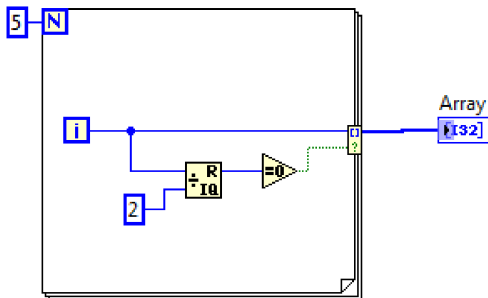


Indeksowanie

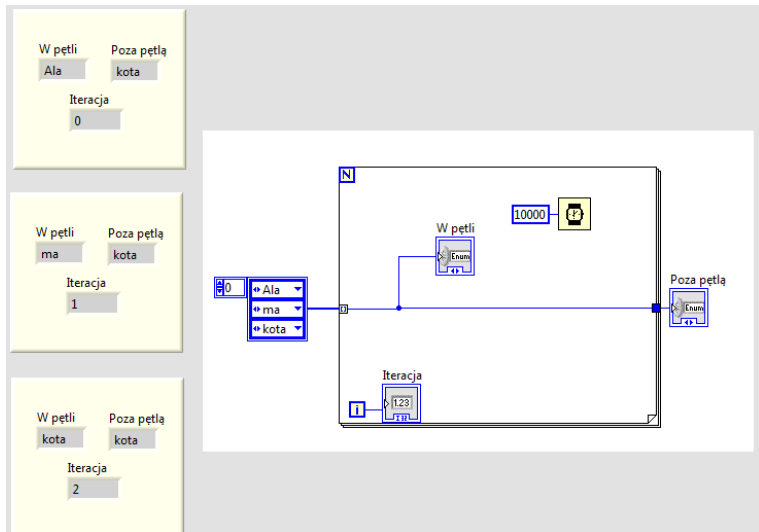
W przypadku kilku warunków na liczbę iteracji zostanie wybrany najmniejszy.



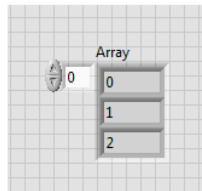
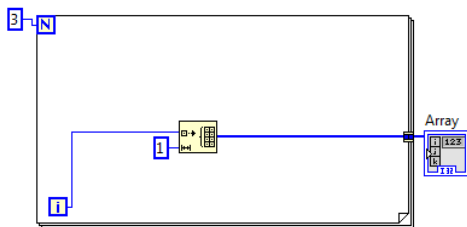
Indeksowanie warunkowe



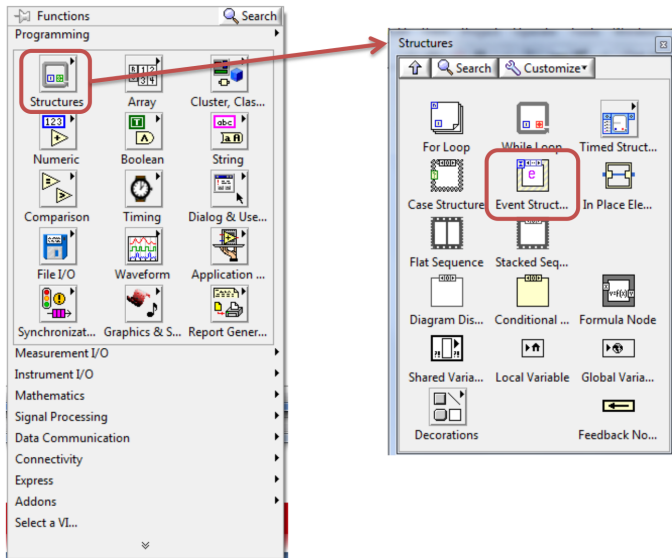
Indeksowanie



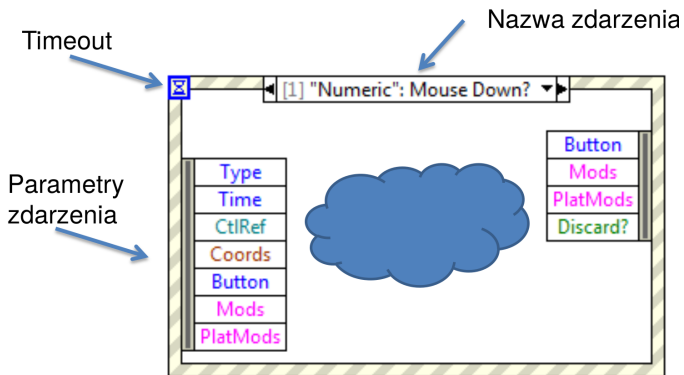
Concatenating



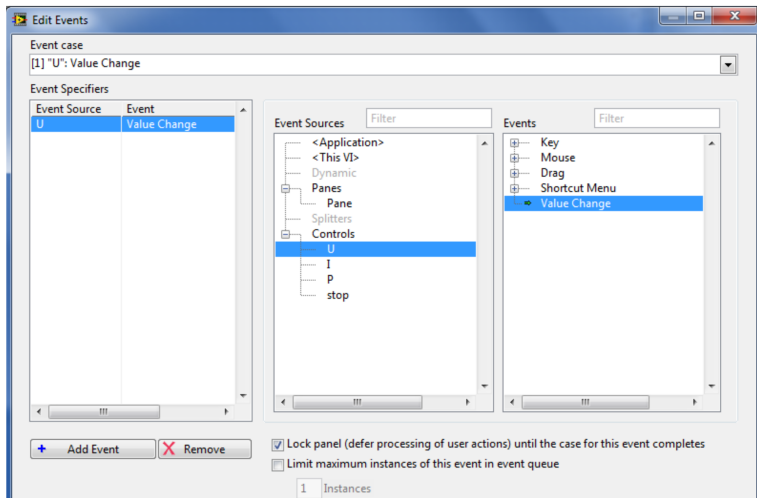
Struktura zdarzeniowa



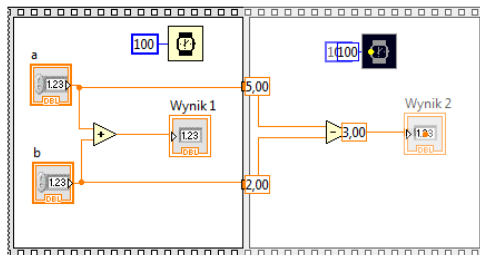
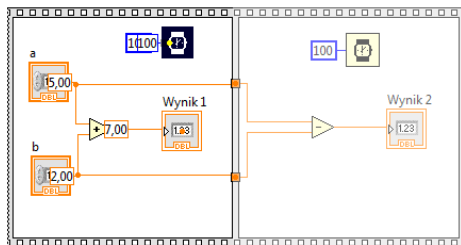
Struktura zdarzeniowa



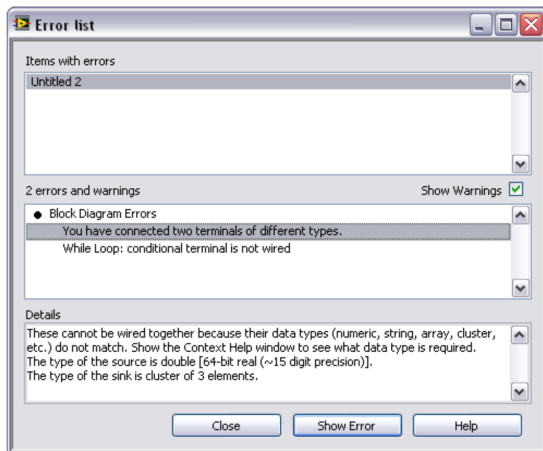
Struktura zdarzeniowa



Flat Sequence



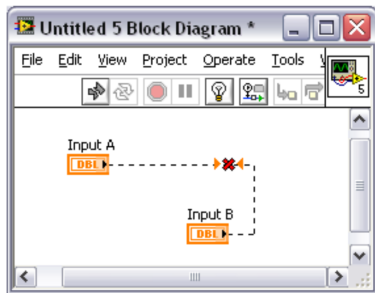
Poprawianie błędów w VI



Złamana strzałka "run" - VI nie może być uruchomiony z powodu błędów.

Przyczyny błędów

- Podłączono ze sobą 2 kontrolki zamiast kontrolka-indykator.
- Podłączono 2 elementy różnych typów (np. numeric z boolean).
- Niektóre funkcje mają niepodłączone wymagane wejścia.
- Podprogram (subVI) nie działa.



Odpluskwanie/Debugowanie

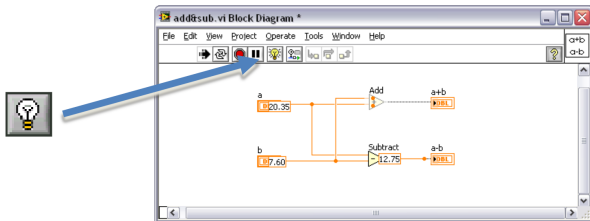
Co robić jeśli VI działa, ale nie tak jak powinno (np. produkuje błędne wyniki obliczeń)

- Execution Highlighting
- Single-Stepping
- Probes
- Breakpoints



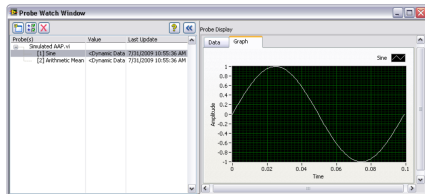
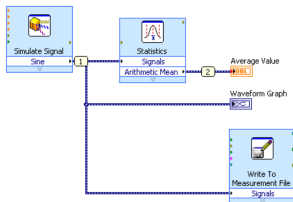
Execution Highlight

- Włączenie lampki powoduje pokazanie jak Vi się wykonuje.
- Lampka powoduje spowolnienie wykonywania się programu - wyłącz ją po znalezieniu błędu.



Probe

- Probe służy do obserwacji wartości w drutach w trakcie działania programu
- Włącz funkcję „Retain Wire Values” żeby użyć narzędzia Probe po zakończeniu działania programu.



Single Stepping i Breakpoint

- Kiedy program podczas wykonywania natrafi na Breakpoint, jego działanie zatrzyma się.
- Można wtedy użyć narzędzia Probe żeby sprawdzić wartości danych albo wykonać Single-Stepping aby prześledzić działanie fragmentu programu
- Przejście po kolei przez wszystkie funkcje w VI.



Niespodziewany wynik

Jeśli uzyskasz niespodziewany wynik w operacjach matematycznych oznacza to, że:

- ∞ (nieskończoność) - wykonano dzielenie jakiejś liczby przez zero.
- NaN (Not a number - to nie liczba) - wykonano niedozwoloną operację np. liczyłeś pierwiastek z liczby ujemnej.

Obsługa błędów

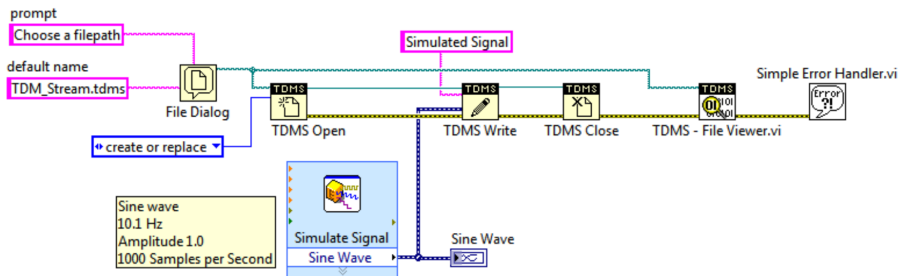
1 Automatyczna obsługa błędów:

- Domyślnie LabVIEW używa automatycznej obsługi błędów.
- Składa się z trzech etapów: zawieszeniu wykonywania się programu, podświetlenie subVI lub funkcji przy której wykonywaniu pojawił się błąd, wyświetleniu listy błędów.
- Akceptowalna dla szybkich prototypowych programów.
- Nie rekomendowana dla profesjonalnych aplikacji.

2 Manualna obsługa błędów:

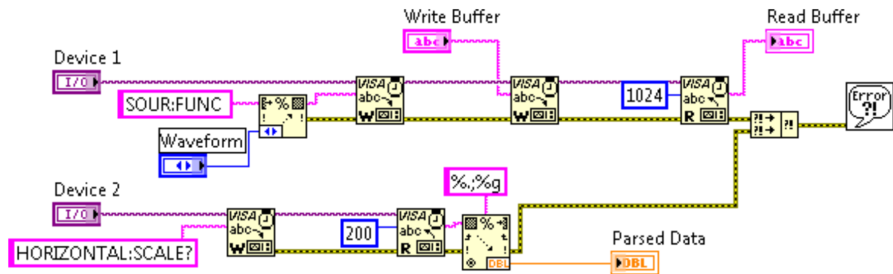
- Zapewnia kontrole kiedy dialog boxes ma się pojawić.
- Błędy są propagowane poprzez połączenie klastra error out z klastrem error in.

Manualna obsługa błędów



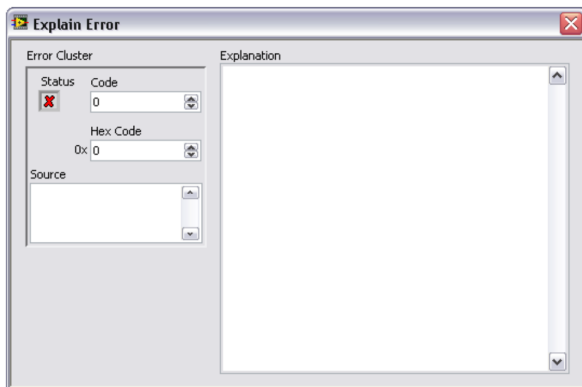
Merge errors function

Merge Errors Function zwraca pierwszy błąd. Jeśli żaden błąd nie zostanie znaleziony to funkcja zwraca pierwszy warning. Merge Errors Function nie łączy błędów.



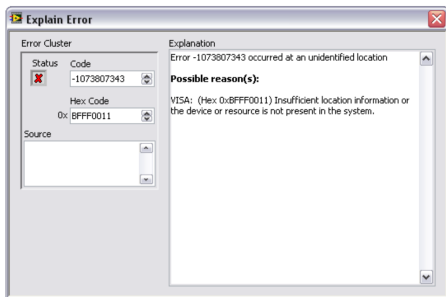
Error Clusters

- Używaj kontrolki lub indykatory error cluster w celu stworzenia informacji o błędzie wejściowym i wyjściowym w subVI.
- Error in/out cluster zawiera następujące informacje: status, kod błędu, źródło.



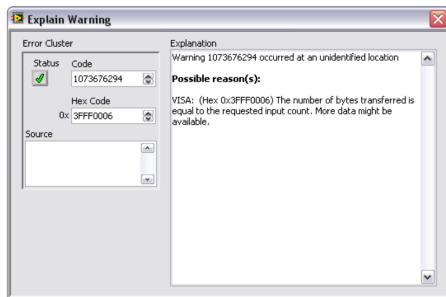
Błędy i ostrzeżenia

Status = TRUE



Error

Status = FALSE
Code = Non-zero



Warning



Dziękuję za uwagę!

Wsparcie Komisji Europejskiej dla produkcji tej publikacji nie stanowi poparcia dla treści, które odzwierciedlają jedynie poglądy autorów, a Komisja nie może zostać pociągnięta do odpowiedzialności za jakiegokolwiek wykorzystanie informacji w niej zawartych.

Wykład został opracowany w oparciu o materiały: "LabVIEW Core 1 Course Manual", "LabVIEW Core 2 Course Manual", pierwotną wersję wykładu: mgr. inż. Marcina Biedy oraz przykładowe egzaminy CLAD opublikowane na stronie www.ni.com.