

Program obliczający
pierwiastki równania
kwadratowego.

Private Sub cmdOblicz_Click()

```
Dim a, b, c, delta, x1, x2
a = Val( txta.Text )
b = Val( txtb.Text )
c = Val( txtc.Text )
delta = b * b - 4 * a * c
lblDelta.Caption = "delta=" & Str( delta )

If delta >= 0 Then
    x1 = ( -b - Sqr( delta ) ) / ( 2 * a )
    x2 = ( -b + Sqr( delta ) ) / ( 2 * a )
    lblx1.Caption = "X1=" & Str( x1 )
    lblx2.Caption = "X2=" & Str( x2 )
Else
    lblx1.Caption = "Brak rozwiązania."
End If
End Sub
```

Private Sub cmdReset_Click()

```
txta.Text = ""
txtb.Text = ""
txtc.Text = ""
lblDelta = ""
lblx1 = ""
lblx2 = ""
txta.SetFocus
End Sub
```

Private Sub cmdKoniec_Click()

```
End
End Sub
```

Private Sub txta_KeyPress(KeyAscii As Integer)

```
If KeyAscii = 13 Then txtb.SetFocus
End Sub
```

Private Sub txtc_KeyPress(KeyAscii As Integer)

```
If KeyAscii = 13 Then cmdOblicz_Click
End Sub
```

'Wykonaj po kliknięciu w przycisk

'Deklarowanie zmiennych typu Variant i zerowanie ich wartości

'Zamień tekst na liczbę (VAL) i przypisz ją zmiennej a

'Przypisz zmiennej wartość i oblicz ją

'Wyświetl napis delta= i wartość zmiennej delta na etykiecie jako tekst (Str)

'Jeżeli delta dodatnia lub =0

'to oblicz pierwiastki

'i wyświetl na etykiecie X1= i wynik jako tekst (Str)

'W przeciwnym wypadku tzn. gdy delta ujemna

'Wyświetl napis

'Koniec analizy wartości delty

'Wykonaj po kliknięciu w przycisk

'Usuń zawartość pól tekstowych

'i etykiet

'W polu tekstowym txta ustaw fokus

'Koniec procedury

'Wykonaj po kliknięciu w przycisk

'Zakończ pracę

'Koniec procedury

'Po ENTER przejdź do pola txtb

'Po ENTER wywołaj zdarzenie cmdOblicz_Click

Dim – instrukcja

Deklaruje zmienną i przydziela pamięć.

Zmienne zadeklarowane za pomocą instrukcji Dim na poziomie modułu są dostępne we wszystkich procedurach wewnątrz modułu. Na **poziomie procedury**, zmienne dostępne są tylko **wewnątrz procedury**.

Np.: Dim Liczba_zatrudnionych As Integer

Jeżeli nie zostanie określony typ danych lub typ obiektu, jako domyślny przyjmowany jest typ Variant.

Podczas inicjowania zmiennych zmienna numeryczna przyjmuje wartość 0, ciąg o zmiennej długości przyjmuje wartość ciągu o długości zerowej (""), a ciąg o stałej długości jest wypełniany zerami. Zmienne typu Variant są inicjowane wartością Empty. Każdy element zmiennej typu zdefiniowanego przez użytkownika jest inicjowany tak, jakby był oddzielną zmienną.

Uwaga Jeżeli instrukcja Dim występuje w procedurze, zwykle umieszczana jest na początku procedury. Aby zmienne przenosiły swoją wartość do innych procedur należy zadeklarować je w sekcji **GENERAL**.

Val – funkcja

Zwraca wartości liczb tworzących ciąg w postaci wartości numerycznej odpowiedniego typu.

Funkcja Val przerywa odczyt ciągu przy pierwszym znaku, który nie jest fragmentem liczby. Symbole i znaki, które często stanowią element wartości numerycznych, na przykład znak dolara czy przecinek, nie są rozpoznawane przez funkcję Val. Funkcja Val rozpoznaje jednak symbol podstawy &O (dla systemu ósemkowego) i &H (dla systemu szesnastkowego). Spacje, tabulatory i znaki wysuwu wiersza są pomijane.

Str – funkcja

Zwraca wartość typu Variant (String) reprezentującą liczbę. Składnia:

Str (liczba)

Obowiązkowy argument liczba jest wartością typu Long zawierającą dowolne poprawne wyrażenie numeryczne.

Podczas zamiany liczby na ciąg znaków na znak liczby rezerwowana jest zawsze spacja. Jeżeli liczba jest dodatnia, zwracany ciąg zawiera na początku spację, co oznacza znak plus.

Funkcja Format służy do zamiany wartości numerycznych, które mają być sformatowane jako daty, godziny lub wartości pieniężne. W odróżnieniu od funkcji Str, funkcja Format nie umieszcza na początku spacji przeznaczonej na znak liczby.

If...Then...Else – instrukcja

Powoduje warunkowe wykonanie grupy instrukcji, zależnie od wartości wyrażenia. Składnia:

If warunek Then instrukcja	np.: If x >= 0 Then y = Sqr (x)
If warunek Then [instrukcje] [instrukcje] Else [instrukcje]	If delta >= 0 Then x1 = (-b - Sqr(delta)) / (2 * a) x2 = (-b + Sqr(delta)) / (2 * a) Else Label1 = " Brak rozwiązania ,, End If

SetFocus — metoda

Przenosi fokus do określonego formantu lub formularza.

Składnia: **object.SetFocus**

Obiekt musi być obiektem **Form**, obiektem MDIForm lub **formantem**, który może otrzymać **fokus**. Po wywołaniu metody **SetFocus** wszelkie dane wejściowe wprowadzone przez użytkownika są kierowane do określonego formularza lub formantu.

Fokus można przenieść tylko do **widocznego** formularza lub formantu. Ponieważ formularze i formanty na formularzu nie są widoczne, dopóki nie zostanie zakończone zdarzenie **Load** formularza, nie można użyć metody **SetFocus** do przeniesienia fokusu na ładowany formularz w jego własnym zdarzeniu **Load**, chyba że najpierw zostanie użyta metoda Show, aby pokazać formularz przed zakończeniem procedury zdarzenia Form_Load.

Nie można również przenieść **fokusu** na formularz lub formant, jeśli właściwość **Enabled** jest ustawiona na wartość **False**. Jeśli właściwość Enabled została ustawiona na wartość False w czasie projektowania, trzeba najpierw ustawić ją na wartość True, zanim dany obiekt będzie mógł otrzymać fokus za pomocą metody SetFocus.

Sqr – funkcja

Zwraca wartość typu Double równą pierwiastkowi kwadratowemu z liczby.

Np.: a = Sqr (16) wynik: a = 4