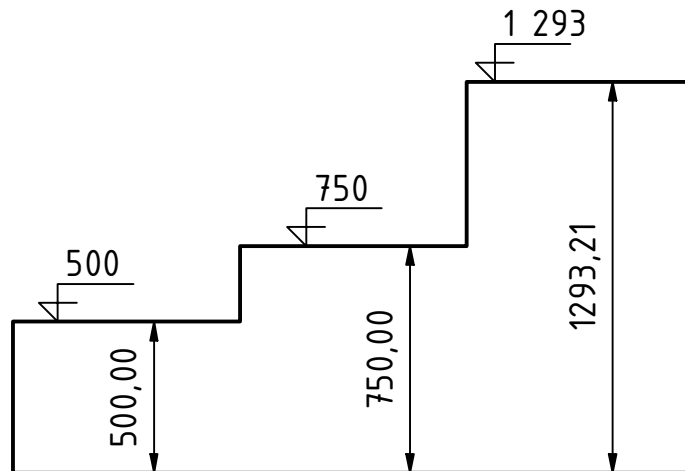


## Příklad použití výškových značek ( 1 : 25 )



### Použití

- Do výkresového pohledu umístěte značky "HeightMark" tak, aby byly připojené ke geometrii
- Během umísťování není potřeba zadávat žádnou hodnotu do výzvy
- Pro aktualizaci hodnot spusťte pravidlo "SetHeightMarks"

### Nastavení

- Pravidlo je vhodné uložit jako externí a načíst
- V pravidle je možné změnit nastavení
  - \* Jméno značky
  - \* Převod jednotek
  - \* Výšku počátku modelu
  - \* Směr výšky
  - \* Formát čísla
- Značku je možné upravit/překreslit, ale musí obsahovat právě jedno zadání s výzvou
- Značku je možné přejmenovat, ale potom je potřeba upravit v pravidle název značky

```

Public Sub Main()
    Dim drawing As DrawingDocument = ThisDoc.Document
    SetHeightSymbols(drawing.ActiveSheet)
End Sub

'*****
'*** Nastaveni / Settings ***

Dim symbolName As String = "HeightMark"
Dim stringFormat As String = "N0" 'Bez desetinných míst / No decimal places
Dim zeroLevel As Double = 0 'Vyska pocatku modelu / Model origin height
Dim fromCentimetersToUnits = 10 '[cm]->[mm] konstanta pro prevod jednotek / Units conversion constant
Dim heightDirection As HeightDirectionEnum = HeightDirectionEnum.Z ' [X, Y, Z] Smer vysky / Height direction

'Dalsi informace o parametru stringFormat
'For more information of stringFormat parameter see
' https://docs.microsoft.com/en-us/dotnet/standard/base-types/custom-numeric-format-strings
' https://docs.microsoft.com/en-us/dotnet/standard/base-types/standard-numeric-format-strings

'*****

Private Function ConvertHeight(height As Double) As Double
    Return height * fromCentimetersToUnits + zeroLevel
End Function

Sub SetHeightSymbols(sheet As Sheet)
    For Each symbol As SketchedSymbol In sheet.SketchedSymbols
        If symbol.Name = symbolName Then
            SetText(symbol)
        End If
    Next
End Sub

Private Sub SetText(symbol As SketchedSymbol)
    Dim textDef As TextBox = symbol.Definition.Sketch.TextBoxes(1)
    Dim height As Double = GetHeight(symbol)
    height = ConvertHeight(height)
    Dim value As String = height.ToString(stringFormat)
    symbol.SetPromptResultText(textDef, value)
End Sub

Private Function GetHeight(symbol As SketchedSymbol) As Double
    Dim position = symbol.Position
    For Each node As LeaderNode In symbol.Leader.AllNodes
        If node.AttachedEntity Is Nothing Then
            Continue For
        End If
        Try
            Dim gi As GeometryIntent = node.AttachedEntity
            Dim drwCurve As DrawingCurve = gi.Geometry
            Dim drawingView As DrawingView = drwCurve.Parent
            Dim line As Line = drawingView.SheetToModelSpace(position)
            Dim direction As UnitVector = line.Direction
            Dim rootPoint As Point = line.RootPoint

            Dim directionZ As Double = GetHeightCoord(direction)
            If Math.Abs(directionZ) < 0.001 Then
                Return GetHeightCoord(rootPoint)
            End If
        Catch
            Continue For
        End Try
    Next

    Return Double.NaN
End Function

Private Function GetHeightCoord(direction As UnitVector) As Double
    Select Case heightDirection
        Case HeightDirectionEnum.X
            Return direction.X
        Case HeightDirectionEnum.Y
            Return direction.Y
        Case HeightDirectionEnum.Z
            Return direction.Z
        Case Else
            Throw New ArgumentOutOfRangeException()
    End Select
End Function

Private Function GetHeightCoord(point As Point) As Double
    Select Case heightDirection
        Case HeightDirectionEnum.X
            Return point.X
        Case HeightDirectionEnum.Y
            Return point.Y
        Case HeightDirectionEnum.Z
            Return point.Z
        Case Else
            Throw New ArgumentOutOfRangeException()
    End Select
End Function

Enum HeightDirectionEnum
    X
    Y
    Z
End Enum

```