# Test – Softwareentwicklung 4. JG (2.12.2022)

Name: Johannes Werner

|  |  |  |
| --- | --- | --- |
| **Beispiel** | **Mögliche Punkte** | **Punkte** |
| **Theorie** | 14 |  |
| **Tageslänge** | 7 |  |
| **MutatedVowel** | 5 |  |
| Gesamt | **26** |  |

## Tageslänge (7 Punkte)

Gegeben ist die Klasse Sun.

Beantworte mit Hilfe eines Programms die Frage, wann die Tageslänge wieder größer oder gleich 9 Stunden ist. Erzeugen Sie ein Objekt der Klasse Sun. Für den Standort Wieselburg, verwenden sie die Argumente 48.12754 und 15.12402 für die Parameter latitude und longitude des Konstruktors. Ermitteln Sie mit Hilfe der Methode SunriseAndSunset, den jeweiligen Sonnenaufgang und Sonnenuntergang durch probierenden der zukünftigen Tage, bis eine Tageslänge größer oder gleich 9 Stunden gefunden wurde. Geben Sie das Datum (ohne Uhrzeit) auf der Konsole aus.

internal class Sun

{

private double latitude;

private double longitude;

public Sun(double latitude, double longitude)

{

this.latitude = latitude;

this.longitude = longitude;

}

public void SunriseAndSunset(int day, int month, int year, out DateTime sunrise, out DateTime sunset)

{

double pi = Math.PI;

double dr = pi / 180;

double rd = 1 / dr;

double lat = this.latitude;

double @long = this.longitude;

double h = 0; // timezone UTC

DateTime now = DateTime.Now;

int m = month;

int d = day;

lat = dr \* lat;

int n = (Int32)(275 \* m / 9) - 2 \* (Int32)((m + 9) / 12) + d - 30;

double l0 = 4.8771 + .0172 \* (n + .5 - @long / 360);

double c = .03342 \* Math.Sin(l0 + 1.345);

double c2 = rd \* (Math.Atan(Math.Tan(l0 + c)) - Math.Atan(.9175 \* Math.Tan(l0 + c)) - c);

double sd = .3978 \* Math.Sin(l0 + c);

double cd = Math.Sqrt(1 - sd \* sd);

double sc = (sd \* Math.Sin(lat) + .0145) / (Math.Cos(lat) \* cd);

if (Math.Abs(sc) <= 1)

{

// calculate sunrise

double c3 = rd \* Math.Atan(sc / Math.Sqrt(1 - sc \* sc));

double r1 = 6 - h - (@long + c2 + c3) / 15;

int hr = (Int32)(r1);

int mr = (Int32)((r1 - hr) \* 60);

sunrise = new DateTime(year, month, day, hr, mr, 0);

// calculate sunset

double s1 = 18 - h - (@long + c2 - c3) / 15;

int hs = (Int32)(s1);

int ms = (Int32)((s1 - hs) \* 60);

sunset = new DateTime(year, month, day, hs, ms, 0);

}

else

{

if (sc > 1)

{

// sun is up all day ...

// Set Sunset to be in the future ...

sunset = new DateTime(now.Year + 1, now.Month, now.Day, now.Hour, now.Minute, now.Second);

// Set Sunrise to be in the past ...

sunrise = new DateTime(now.Year - 1, now.Month, now.Day, now.Hour, now.Minute - 1, now.Second);

}

else

{

// sun is down all day ...

// Set Sunrise and Sunset to be in the future ...

sunrise = new DateTime(now.Year + 1, now.Month, now.Day, now.Hour, now.Minute, now.Second);

sunset = new DateTime(now.Year + 1, now.Month, now.Day, now.Hour, now.Minute, now.Second);

}

}

}

}

## Code:

In Program.cs

using Tageslaenge\_1\_1;

Sun sun = new Sun(48.12754, 15.12402);

DateTime date = DateTime.Now.Subtract(new TimeSpan(24, 0 , 0));

DateTime sunrise;

DateTime sunset;

do

{

date = date.AddDays(1);

sun.SunriseAndSunset(date.Day, date.Month, date.Year, out sunrise, out sunset);

} while (sunset.Subtract(sunrise).Hours < 9);

Console.WriteLine("Ab den " +date.ToShortDateString() +" sind die Tage länger als 9 Stunden.");

# Ergebniss:



## MutatedVowel (5 Punkte)

Gegeben ist die Klasse ConsoleInput und MutatedVowelEventArgs.

Rufen Sie die Methode ReadInput der Klasse ConsoleInput in einer Endlosschleife auf. Melden Sie sich zuvor für das event MutatedVowelInput an. Wenn ein Umlaut eingegeben wurde, soll dies auf der Konsole ausgegeben werden (siehe Screenshot).

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

internal class ConsoleInput

{

public event EventHandler<MutatedVowelEventArgs> MutatedVowelInput;

public string ReadInput(string description)

{

Console.Write(description);

string result = Console.ReadLine();

if(result.ToLower().Contains("ä") )

{

if(MutatedVowelInput != null)

{

MutatedVowelInput(this, new MutatedVowelEventArgs('ä'));

}

}

if(result.ToLower().Contains("ö") )

{

if (MutatedVowelInput != null)

{

MutatedVowelInput(this, new MutatedVowelEventArgs('ö'));

}

}

if (result.Contains("ü"))

{

if (MutatedVowelInput != null)

{

MutatedVowelInput(this, new MutatedVowelEventArgs('ü'));

}

}

return result;

}

}

internal class MutatedVowelEventArgs : EventArgs

{

public char MutatedVowel { get; }

public MutatedVowelEventArgs(char vowel):base()

{

this.MutatedVowel = vowel;

}

}

# Code:

In Program.cs

using MutatedVowel\_1\_2;

ConsoleInput consoleInput = new ConsoleInput();

consoleInput.MutatedVowelInput += informAboutVowel;

while (true)

{

consoleInput.ReadInput("Bitte Text eingeben: ");

}

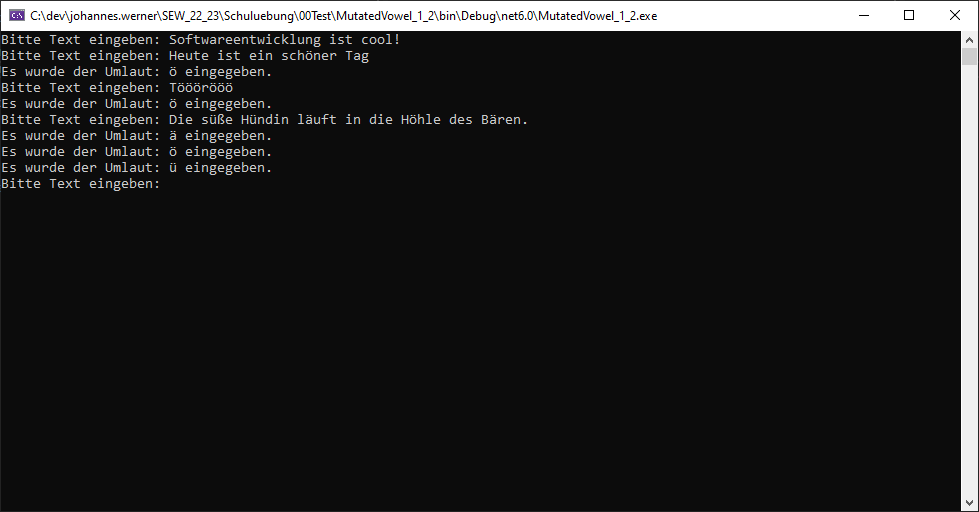
static void informAboutVowel(object? sender, MutatedVowelEventArgs e)

{

Console.WriteLine("Es wurde der Umlaut: " + e.MutatedVowel + " eingegeben.");

}

# Ergebnis:

****

**Geben Sie Ihre Programme (Zip-Archiv des gesamten Projektordners) in Eduvidual ab.**