Assignments

Homework 4

Objective

Use Python to create a clock program with a Windows interface(provided).

Problem domain

Programming teams work on different features, the Windows graphical user interface team has completed the front end. Your task is to use the modulus(%) operator to calculate the seconds, minutes, and hours given the total seconds since 1970. Afterward, attach your code to existing Python Windows program to display an analog clock.

https://www.epoch101.com/
To get the hours from the seconds since 1970 take into account the fact that 3600 is on hour. Using this with integer division, you can divide seconds since 1970 by 3600 to get the hour. If you get the current seconds since 1970, you can calculate the current hour as of right now in GMT time.

To get the minutes use integer division to divide seconds since 1970 by 60 and get the remainder of minutes divided by 60. (Use the % operator).

To get the seconds, calculate the seconds remainder by dividing seconds since 1970 by 3600. Get the remainder of seconds remainder by dividing by 60. (Use the % operator)

Prerequisites

Install Python
GitHub account and repository
Install and configure Visual Studio Code

Write Code

In Visual Studio Code, find the /src/homework/e_functions folder.

a. In the value_return.py folder, write the value return functions:
   get_hour takes an epoch_seconds parameter
   get_minutes takes an epoch_seconds parameter
   get_seconds takes an epoch_seconds parameter

   Use the instructions from above to calculate the return values.
b. Create a value return function time_from_utc that accepts two parameters utc_offset and utc_zero.
   Sum utc_offset and utc_zero.
   Return the remainder of the sum divided by 24.
c. Write the Tests for the functions (see next section)

Write Unit Test

In Visual Studio Code, find the /tests/homework/e_functions folder.

a. In the file test_functions.py add the following code:
   import unittest
   from src.homework.e_functions.value_return import get_hour

```
from src.homework.e_functions.value_return import get_minutes
from src.homework.e_functions.value_return import get_seconds
from src.homework.e_functions.value_return import time_from_utc


class Test_Config(unittest.TestCase):
```

b. After the line that begins with class write a test case function test_get_hour
Test that get_hour with parameter value 3800 returns the value 1.
Test that get_hour with parameter value 3600 returns the value 1.

c. Test case function test_get_minutes
Test that get_minutes with parameter value 3800 returns the value 3.
Test that get_minutes with parameter value 3600 returns the value 0.

d. Test case function test_get_seconds
Test that get_seconds with parameter value 3800 returns the value 20.
Test that get_seconds with parameter value 3600 returns the value 0.

e. Test case function test_utc_time_to_eastern_standard_time
Test that time_from_utc with parameters -5 and 20 returns 15

f. Test case function test_utc_time_to_central_standard_time
Test that time_from_utc with parameters -6 and 20 returns 14

g. Test case function test_utc_time_to_mountain_standard_time
Test that time_from_utc with parameters -7 and 20 returns 13

h. Test case function test_utc_time_to_pacific_standard_time
Test that time_from_utc with parameters -8 and 20 returns 12

Run the Unit Tests

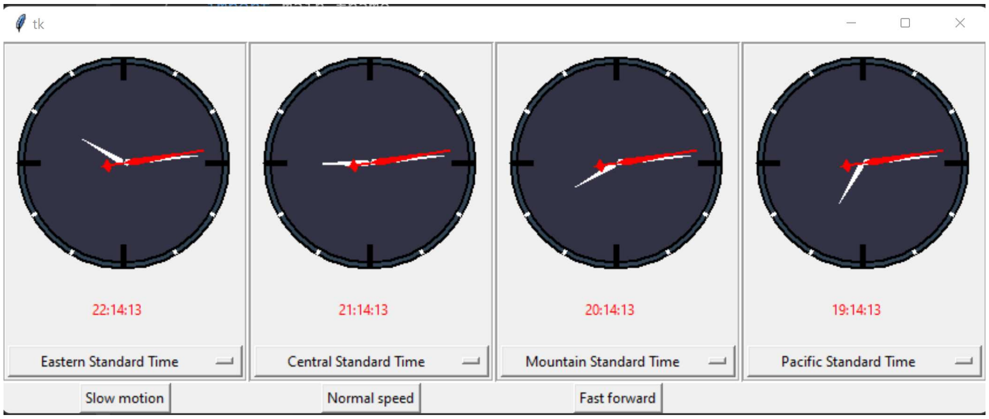
In Visual Studio Code, find the /tests/homework/e_functions folder

a. From the source code root folder, find the run_tests.py file.
Replace: `from tests.homework.d_repetition import test_repetition`
with
`from tests.homework.e_functions import tests_functions`
Verify that line has the following statement :
Replace tests_repetion with tests_functions.
`suite = unittest.TestLoader().loadTestsFromModule(tests_functions)`
b. Click on the play button to run the test case.
c. Make sure the test results return ok for the test cases (Fix the code if it fails).

Create and Run the Main Program

In Visual Studio Code, find the /src/homework/e_functions folder find the main.py file.
write code to create display a Windows analog clock.

1. Download clock.py, clock_app.py, and main_frame.py (Find them at the end of the page).
2. Copy the files to /src/homework/e_functions.
3. Run the program from the clock_app.py file.



Upload the Changes to GitHub

a. In Visual Studio Code, click on the Source Control icon .
b. Select only the files pertaining to this assignment.
c. Click on the + to stage the changes.
d. Click on the check mark to commit the changes.
e. From the menu select the …, from the menu select Push.

Submit the Assignment for Grading in Blackboard

Make sure to add your GitHub user name to the Comment edit box.

DOWNLOAD FILES

clock.py
clock_app.py
main_frame.py

Assignment 3

Draw the stack memory diagram for the following code:

```
def main():
    val = 0

main()
```