**Python Clock**

**Description:**

TimeBud is an analogue clock program written using Python. It provides an analogue clock, digital clock and current date based on the system time. It also includes a stop watch and an alarm. Other miscellaneous features include a moving face and sound.

**Problem (Program Specification):**

**Abstraction:**

Main Function:

An analogue clock requires three hands. The **second** hand, **minute** hand, and **hour** hand. The hands would require a graphical object to represent them. Furthermore, a window is required for the hands to be present. The hands also require some measurement to indicate where they should point to. The measurement is also needed to produce a display to the user to indicate what time it is so another object is needed to indicate the number the hands are pointing to.

Additional Functions:

Some users prefer reading the time in digital format, so a **digital clock** should also be included. When users look at the time, sometimes they also want to see the date. Therefore, the **date** should also be included in the clock. Users may also want to time things, so a **stop watch** should also be included. Users would also want the clock to ring at a certain time, so an **alarm** is also useful.

Miscellaneous Features:

Users may want to know when a minute has ended, so the program should **make a sound every minute**. To make the program less dull, the background consists a face that is **animated every minute**. Moreover, the face moves every second when the stop watch is running. The face should also move when the alarm is running.

**Program Design:**

Modularization:

Main

analogue\_clock(): Draw and update hour, minute and seconds hand. Calls the alarm module when needed. Calls the Stop Watch module when needed.

drawTimeBud(): The character in the background.

draw\_clock\_circles(): The circles represent each number on the clock.

testCurrentTime

random-colors (): Generates random colors for the circles.

testCurrentTime

testCurrentTime

draw\_hour\_hand()

alarm(): Setup an alarm and rings at the time specified.

draw\_seconds\_hand\_and\_digital\_clock\_and\_date()

draw\_minute\_hand()

testCurrentTime,firstPress,temp,loop

drawButtons()

stop\_watch(testCurrentTime,firstPress,temp,loop) : Creates the green hand for the Stop Watch. Changes 'myImage', the face', everytime a second goes by.

Draws and updates digital date.

Draws and updates digital time.

Create Stop Watch button.

Create Alarm Button

**Functions:**

random\_colors(): Generates random colors.

Input: None

Returns: A random rgb color.

draw\_clock\_circles(): Creates the circles that represent the numbers on an analogue clock.

Input: None

Output: The circles on the clock.

Returns: None

draw\_seconds\_hand\_and\_digital\_clock\_and\_date(): Creates and updates the seconds hand as well as create and update a digital clock everytime the function is called.

Input: testCurrentTime

Output: Second hand, digital date and digital time.

Returns: None

draw\_minute\_hand(): Creates and updates the minute hand everytime the function is called.

Input: testCurrentTime

Output: The minute hand

Returns: None

draw\_hour\_hand(): Creates and updates the hour hand everytime the function is called.

Input: testCurrentTime

Output: The hour hand.

Returns: None

minute\_sound(): Plays a sound every minute.

Input: None

Output: A sound every minute.

Returns: None

drawTimeBud(): Draws the background character 'TimeBud'. It includes its shirt and his face in the background.

Input: None

Output: The character in the background.

Returns: None

stop\_watch(): Creates the green hand for the Stop Watch. Changes 'myImage', the face', everytime a second goes by.

Input: testCurrentTime, firstPress, temp, loop

Output: A visual stop watch

alarm(): Setup an alarm and rings at the time specified.

Input: None

Output: Moving face when it is the time set by the user is reached.

Returns: None

analogue\_clock(): Draw and update hour, minute and seconds hand. Calls the alarm module when needed. Calls the Stop Watch module when needed.

Input: None

Output: The analogue clock.

Returns: None

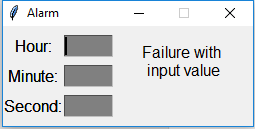
**Performance and Functionality:**

Python is a high programming language meaning that it takes time for the compiler to convert it into an object code. The code I have written has a total of 10 functions that are used in the main program. There are several functionalities stated before: The analogue clock, stop watch, alarm including a moving face using an image. These functionalities may reduce the performance, but on the other side, the program has more features. The program might be slower at times, but it’s functionality should still work.

**Program Maintenance:**

The main() function calls on the analogue\_clock() function. The analogue clock function calls the rest of the functions. Therefore, any additional features the are used to mod the program should be in the form of a function and it should be called in the analogue\_clock() function. The main() functions only calls anaologue\_clock().

**User Interface:**

Alarm input

Alarm Button

Stop Watch button

Digital Time

Digital Date

Validation (Exception Handling)

Validation for alarm:

1. Presence Check. All fields must be entered.
2. Value Error. Time must be in the correct format.
3. Type Error. Input must be an integer.
4. Name Error. Input must be in the correct format.