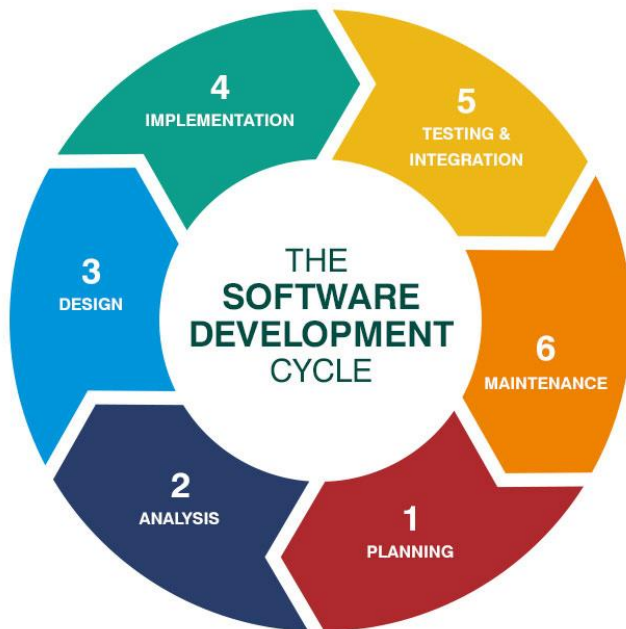


Testing

Testing is an important part of the software development process but often one process which is often over looked (and squeezed) in terms of project time

Software Development Life Cycle



See this link for testing methodologies

<http://softwaretestingfundamentals.com/software-testing-methods/>

The testing you are going to carry out testing known as **Black-Box testing**, see this link <http://softwaretestingfundamentals.com/black-box-testing/>

To do this 'test' you will need to following Python files:

- **testing_of_afunction2_sv2.py**
- **afunction2.pyc**

This problem involves testing an algorithm the 'program' takes an integer input, called x and carries out a calculation using x the evaluation [the result] is returned to the calling program.

You will need to import the library is shown below

```
import afunction2 #this imports the file afunction2.pyc
```

For a given input [x] you would call the function `.calc(x)`

And the function will return a given value

```
afunction2.calc(x)
```

Task: Write a program that determines the maxima of this function. For maxima see <http://www.mathsisfun.com/algebra/functions-maxima-minima.html>

You should achieve this without any knowledge of the function `[.calc(x)]` is implemented in the code library.

You will only use the output values returned by the function `.calc()` for the given inputs to determine its maxima.

Hints

- Begin by using a queue or stack keep track of what input `[x]` values you want to try and what the `[y]` output values are
- Determine the search space (values of `x`) that you want to explore (e.g., from -300 to 300).
- Also decide on the granularity at which you want to explore the search space (e.g., in increments of 20).
- Select values from the search space with the chosen granularity to feed as inputs to the simulator.
- Create a Work Queue/Stack task to run the simulator with a given input value.