

## Regression Testing

### Build Test Suite:

```
'''
Runs the test suite in parallel x number of times
'''

import os
import sys
import datetime

#Get the arguments from CLI
paramIn = sys.argv[1]
numofTimes = int(paramIn)
command = "python runTestSuite.py 10 &"

timeStart = datetime.datetime.now()
for num in range(0, numofTimes):
    os.system(command)
timeEnd = datetime.datetime.now()

print("Input parallels used for StressTest:", numofTimes)
print("Number of times runTestSuite was used: 10")
print("Time started:", timeStart)
print("Time ended:", timeEnd)
```

### Run Test Suite Code:

```
import sys
import win32com.client
import pytest
import datetime
import test_generator as tGen
import test_hashgrinder as tHash
import test_pagan as tPagan
import test_pgnreader as tPgn

emailSubject = ""
emailBody = ""

#Get the arguments from CLI
if(len(sys.argv) == 2):
    numberLoop = sys.argv[1]
```

## Regression Testing

```
    emailResultTo = None
    shoudlIEmail = False
elif(len(sys.argv) == 1):
    numberLoop = 1
    emailResultTo = None
    shoudlIEmail = False
else:
    numberLoop = sys.argv[1]
    emailResultTo = sys.argv[2]
    shoudlIEmail = True

numberLoop =int(numberLoop)

#Runs Default Pytest Module
def runPytest():
    if(pytest.main(['--resultlog=emailBody.txt'],plugins=None)):
        return True
    else:
        return False

def makeEmail(to, title, body):
    olMailItem = 0
    ol = win32com.client.Dispatch("Outlook.Application")
    msg = ol.CreateItem(olMailItem)
    msg.To = to
    msg.Subject = title
    msg.Body = body
    msg.Send()
    ol.Quit()

def sendEmail(eSubj):
    emailFile = open("emailBody.txt", "r")
    body = ""
    for line in emailFile:
        body += line

    makeEmail(emailResultTo, eSubj, body)

#Main Driver Method
def runTests():
    #For number of loops run the pytest suite
    timeStart = datetime.datetime.now()
    for tests in range(0, numberLoop):
        if(runPytest()):
```

## Regression Testing

```
        emailSubject = "FAIL"
    else:
        emailSubject = "PASS"
    timeEnd = datetime.datetime.now()

    if(shoudlIEmail):
        sendEmail(emailSubject)

    print("Time Started:", timeStart)
    print("Time Ended:", timeEnd)

#Bottom Run
runTests()
```

## Regression Testing

### Output

```
(pytest) C:\Users\v_mar\OneDrive - Seattle University\CPSC\SQ19\TestingAndDebugging\Project\TestingDebugging-Pr
oject\pagan-master\test>python runTestSuite.py 1 marcellig@seattleu.edu
===== test session starts =====
platform win32 -- Python 3.7.3, pytest-4.5.0, py-1.8.0, pluggy-0.11.0
rootdir: C:\Users\v_mar\OneDrive - Seattle University\CPSC\SQ19\TestingAndDebugging\Project\TestingDebugging-Pr
oject\pagan-master
plugins: cov-2.6.1
collected 18 items

test_generator.py . [ 5%]
test_hashgrinder.py ..... [ 61%]
test_pagan.py .. [ 72%]
test_pgnreader.py ..... [100%]

===== warnings summary =====
C:\Users\v_mar\Anaconda3\envs\pytest\lib\site-packages\_pytest\resultlog.py:39
C:\Users\v_mar\Anaconda3\envs\pytest\lib\site-packages\_pytest\resultlog.py:39: PytestDeprecationWarning: --r
esult-log is deprecated and scheduled for removal in pytest 5.0.
See https://docs.pytest.org/en/latest/deprecations.html#result-log-result-log for more information.
  _issue_warning_captured(RESET_LOG, config.hook, stacklevel=2)

-- Docs: https://docs.pytest.org/en/latest/warnings.html
===== 18 passed, 1 warnings in 2.79 seconds =====
Time Started: 2019-06-05 16:39:36.047901
Time Ended: 2019-06-05 16:39:39.047875
```

## Regression Testing

### Email Message

 Reply

 Reply all

 Forward



**PASS**

**Marcelli, Valerio** <marcelli@seattleu.edu> 

4:39 PM

To: Marcelli, Valerio

```
. test/test_generator.py::test_hashGen
. test/test_hashgrinder.py::test_hashgrinder_init_weapon_list
. test/test_hashgrinder.py::test_grind_hash_for_aspect
. test/test_hashgrinder.py::test_hex_to_rgb
. test/test_hashgrinder.py::test_grind_hash_for_colors
. test/test_hashgrinder.py::test_grind_hash_for_weapon
. test/test_hashgrinder.py::test_choose_weapon
. test/test_hashgrinder.py::test_choose_aspect
. test/test_hashgrinder.py::test_map_decision
. test/test_hashgrinder.py::test_split_sequence
. test/test_hashgrinder.py::test_diff
. test/test_pagan.py::test_ImageCreation
. test/test_pagan.py::test_ImageChange
. test/test_pgnreader.py::test_parse_pagan_file
. test/test_pgnreader.py::test_decide_optional_pixels
. test/test_pgnreader.py::test_invert_vertical
. test/test_pgnreader.py::test_enforce_vertical_symmetry
. test/test_pgnreader.py::test_diff
```