Contents

1 Basic Test Results

```
Wed 20 Mar 2024 11:35:19 IST
Process Process-60:
Traceback (most recent call last):
  File "/usr/lib/python3.9/multiprocessing/process.py", line 315, in _bootstrap
    self.run()
  File "/usr/lib/python3.9/multiprocessing/process.py", line 108, in run
    self._target(*self._args, **self._kwargs)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/autotest.py", line 74, in wrap
    res=target(*args, **kwargs)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 39, in import_runner
    code,res = peel(runners, modulename, fname, args, kwargs)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 7, in peel
    return runners[-1](modulename, fname, args, kwargs,options,runners[:-1])
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 17, in check_args
    code,res = peel(runners, modulename, fname, args, kwargs)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 7, in peel
    return runners[-1](modulename, fname, args, kwargs,options,runners[:-1])
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 12, in base_runner
    return None,func(*args, **kwargs)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 53, in is_power
    return powerer(b,b,x)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
    return powerer(base, log_mult(current, base), dest_value)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
    return powerer(base, log_mult(current, base), dest_value)
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
    return powerer(base, log_mult(current, base), dest_value)
  [Previous line repeated 977 more times]
  File "/tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 28, in log_mult
    if y > 1:
RecursionError: maximum recursion depth exceeded in comparison
Wed 20 Mar 2024 11:35:19 IST
Archive:  /tmp/bodek.__nbzvkz/intro2cs1/ex7r/dori.plg/final/submission
  inflating: src/ex7.py
9 passed tests out of 9 in test set named 'presubmit'.
result_code    presubmit    9    1
--> BEGIN TEST INFORMATION
Test name: addmult_1
Module tested: ex7
Function call: mult(3,0)
Expected return value: 0
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'addmult_1' failed.
**********************************************************************
Wrong result, input: [3, 0]:
expected: 0
actual:   3
result_code    addmult_1    wrong    1
13 passed tests out of 14 in test set named 'addmult'.
result_code    addmult    13    1
10 passed tests out of 10 in test set named 'iseven'.
result_code    iseven    10    1
--> BEGIN TEST INFORMATION
Test name: logmult_1
Module tested: ex7
Function call: log_mult(3,0)
```

```
Expected return value: 0
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'logmult_1' failed.
**********************************************************************
Wrong result, input: [3, 0]:
expected: 0
actual:   3
result_code    logmult_1    wrong    1
13 passed tests out of 14 in test set named 'logmult'.
result_code    logmult    13    1
--> BEGIN TEST INFORMATION
Test name: ispower_f0
Module tested: ex7
Function call: is_power(0,2)
Expected return value: False
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'ispower_f0' failed.
**********************************************************************
Test did not complete, exited with exitcode 1.
This probably means your code caused an exception to be raised.
result_code    ispower_f0    exception    1
23 passed tests out of 24 in test set named 'ispower'.
result_code    ispower    23    1
8 passed tests out of 8 in test set named 'reverse'.
result_code    reverse    8    1
8 passed tests out of 8 in test set named 'hanoi'.
result_code    hanoi    8    1
10 passed tests out of 10 in test set named 'ones'.
result_code    ones    10    1
--> BEGIN TEST INFORMATION
Test name: compare_t1
Module tested: ex7
Function call: compare_2d_lists([[]],[[]])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t1' failed.
**********************************************************************
result_code    compare_t1    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t4
Module tested: ex7
Function call: compare_2d_lists([[1, 2], [4, 5, 8]],[[1, 2], [4, 5, 6]])
Expected return value: False
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t4' failed.
**********************************************************************
result_code    compare_t4    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t5
Module tested: ex7
Function call: compare_2d_lists([[1, 2], [4, 5, 8]],[[1, 2], [4, 5, 8]])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
```

```
********************     The test named 'compare_t5' failed.
**********************************************************************
result_code    compare_t5    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t6
Module tested: ex7
Function call: compare_2d_lists([[1], [2], [3], [4], [5], [6], [7], [8]],[[1], [2], [3], [4], [5], [6], [7], [8]])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t6' failed.
**********************************************************************
result_code    compare_t6    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t7
Module tested: ex7
Function call: compare_2d_lists([[1], [2], [3], [4], [5], [6], [7], [9]],[[1], [2], [3], [4], [5], [6], [7], [8]])
Expected return value: False
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t7' failed.
**********************************************************************
result_code    compare_t7    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t8
Module tested: ex7
Function call: compare_2d_lists([[], [1], [2, 3], [4, 5, 6]],[[], [1], [2, 3], [4, 5, 6]])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t8' failed.
**********************************************************************
result_code    compare_t8    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t10
Module tested: ex7
Function call: compare_2d_lists([[], [1], [2, 3], [4, 5, 6], [7, 8], [9], []],[[], [1], [2, 3], [4, 5, 6], [7, 8], [9], []])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t10' failed.
**********************************************************************
result_code    compare_t10    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t11
Module tested: ex7
Function call: compare_2d_lists([[], [1], [2, 3], [4, 5, 6], [7, 8], [9, 10], []],[[], [1], [2, 3], [4, 5, 6], [7, 8], [9],
Expected return value: False
More test options: {}
--> END TEST INFORMATION
**********************************************************************
********************     There is a problem:
********************     The test named 'compare_t11' failed.
**********************************************************************
result_code    compare_t11    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t14
Module tested: ex7
Function call: compare_2d_lists([[9, 8, 7, 6, 5, 4, 3, 2, 1, 9, 8, 7, 6, 5, 4, 3, 2, 1]],[[9, 8, 7, 6, 5, 4, 3, 2, 1, 9, 8,
Expected return value: True
```

```
More test options: {}
--> END TEST INFORMATION
************************************************************************
********************     There is a problem:
********************     The test named 'compare_t14' failed.
************************************************************************
result_code    compare_t14    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t15
Module tested: ex7
Function call: compare_2d_lists([[9, 8, 7, 6, 5, 4, 3, 2, 0, 9, 8, 7, 6, 5, 4, 3, 2, 1]],[[9, 8, 7, 6, 5, 4, 3, 2, 1, 9, 8,
Expected return value: False
More test options: {}
--> END TEST INFORMATION
************************************************************************
********************     There is a problem:
********************     The test named 'compare_t15' failed.
************************************************************************
result_code    compare_t15    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t16
Module tested: ex7
Function call: compare_2d_lists([[4, 5, 8], [1, 2]],[[4, 5, 8], [1, 2]])
Expected return value: True
More test options: {}
--> END TEST INFORMATION
************************************************************************
********************     There is a problem:
********************     The test named 'compare_t16' failed.
************************************************************************
result_code    compare_t16    modified    1
--> BEGIN TEST INFORMATION
Test name: compare_t17
Module tested: ex7
Function call: compare_2d_lists([[4, 5, 6], [1, 2]],[[4, 5, 8], [1, 2]])
Expected return value: False
More test options: {}
--> END TEST INFORMATION
************************************************************************
********************     There is a problem:
********************     The test named 'compare_t17' failed.
************************************************************************
result_code    compare_t17    modified    1
6 passed tests out of 18 in test set named 'compare'.
result_code    compare    6    1
20 passed tests out of 20 in test set named 'magic'.
result_code    magic    20    1
Running mypy
Success: no issues found in 2 source files
Finished running mypy

TESTING COMPLETED
```

2 ex7.py

```
1  ######################################################################
2  # FILE : ex7.py
3  # WRITER : Dori_Peleg , dori.plg , 207685306
4  # EXERCISE : intro2cs ex7 2024
5  # DESCRIPTION: Let's recourse
6  # STUDENTS I DISCUSSED THE EXERCISE WITH:
7  # WEB PAGES I USED:
8  # NOTES: Recursion is a must, see excercise description
9  ######################################################################
10
11 from typing import *
12 from ex7_helper import *
13
14 def mult(x: N, y: int) -> N:
15     """multiplies recursively"""
16     if y > 1:
17         return add(mult(x,subtract_1(y)),x)
18     return x
19
20 def is_even(n: int) -> bool:
21     """checks if a non-neg integer is even"""
22     if n>0:
23         return not is_even(subtract_1(n))
24     return True
25
26 def log_mult(x: N, y: int) -> N:
27     """multiplies efficiantly - O(log(y))"""
28     if y > 1:
29         if is_odd(y):
30             return add(add(
31                 log_mult(x, divide_by_2(y)),
32                 log_mult(x, divide_by_2(y))),
33                   x)
34         else:
35             return add(
36                 log_mult(x,divide_by_2(y)),
37                 log_mult(x,divide_by_2(y)))
38     return x
39
40 def powerer(base: int, current: int, dest_value: int) -> bool:
41     """brings up to power until it can check equality to the dest_value
42     O(log(b)*log(x))"""
43     if current < dest_value:
44         return powerer(base, log_mult(current, base), dest_value)
45     return current == dest_value
46
47 def is_power(b: int,x: int) -> bool:
48     """Not recursive, opens a recursive function"""
49     if x in (b, 1):  # x equals base or base^0
50         return True
51     if b == 1:  # base is 1 but x is not
52         return False
53     return powerer(b,b,x)
54
55 def reverse(s: str) -> str:
56     """flips the order of letters in a string"""
57     if len(s)>0:
58         return append_to_end(reverse(s[1:]),s[0])  # cheating?
59     return ""
```

```
60
61  def play_hanoi(hanoi: Any, n: int, src: Any, dest: Any, temp: Any) -> None:
62      """the algorithm for the hanoi towers"""
63      if n>0:
64          if n > 1:
65              play_hanoi(hanoi,n-1,src,temp,dest)
66              hanoi.move(src,dest)
67              play_hanoi(hanoi,n-1,temp, dest,src)
68          else:
69              hanoi.move(src, dest)
70
71  def digs_in_number(num: int,dig: int) -> int:
72      """helper function to find the amount of times a certain digit apears in a number"""
73      if num > 0:
74          cur = 0
75          if num % 10 == dig:
76              cur = 1
77          return digs_in_number(num//10,dig) + cur
78      return 0
79
80  def number_of_ones(n: int) -> int:
81      """counts how many ones are between 0 and n"""
82      if n==1:
83          return 1
84      return digs_in_number(n,1) + number_of_ones(n-1)
85
86  def compare_1d_lists(l1: List[int], l2: List[int]) -> bool:
87      """compares the values and place of two 1D lists"""
88      if len(l1) == len(l2):
89          if len(l1) > 0 and len(l2) > 0:
90              return l1.pop(-1) == l2.pop(-1) and compare_1d_lists(l1,l2)
91          return True
92      return False
93
94  def compare_2d_lists(l1: List[List[int]], l2: List[List[int]]) -> bool:
95      """compares the values and place of two 2D lists"""
96      if len(l1) == len(l2):
97          if len(l1) > 0 and len(l2) > 0:
98              return compare_1d_lists(l1.pop(-1),l2.pop(-1)) and compare_2d_lists(l1,l2)
99          return True
100     return False
101
102 def magic_list(n: int) -> List[Any]:
103     """preforms the magic of a list containing empty lists containing empty lists"""
104     if n>0:
105         return magic_list(n-1) + [magic_list(n-1)]
106     return []
107
108
109
110 if __name__ == "__main__":
111     pass
```