

Contents

1	Basic Test Results	2
2	ex7.py	6

1 Basic Test Results

```
1 Wed 20 Mar 2024 11:35:19 IST
2 Process Process-60:
3 Traceback (most recent call last):
4   File "/usr/lib/python3.9/multiprocessing/process.py", line 315, in _bootstrap
5     self.run()
6   File "/usr/lib/python3.9/multiprocessing/process.py", line 108, in run
7     self._target(*self._args, **self._kwargs)
8   File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/autotest.py", line 74, in wrap
9     res=target(*args, **kwargs)
10  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 39, in import_runner
11    code,res = peel(runners, modulename, fname, args, kwargs)
12  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 7, in peel
13    return runners[-1](modulename, fname, args, kwargs,options,runners[:-1])
14  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 17, in check_args
15    code,res = peel(runners, modulename, fname, args, kwargs)
16  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 7, in peel
17    return runners[-1](modulename, fname, args, kwargs,options,runners[:-1])
18  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/lib/testrunners.py", line 12, in base_runner
19    return None,func(*args, **kwargs)
20  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 53, in is_power
21    return powerer(b,b,x)
22  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
23    return powerer(base, log_mult(current, base), dest_value)
24  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
25    return powerer(base, log_mult(current, base), dest_value)
26  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 44, in powerer
27    return powerer(base, log_mult(current, base), dest_value)
28  [Previous line repeated 977 more times]
29  File "/tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/testdir/src/ex7.py", line 28, in log_mult
30    if y > 1:
31  RecursionError: maximum recursion depth exceeded in comparison
32 Wed 20 Mar 2024 11:35:19 IST
33 Archive: /tmp/bodek.__nbzvkvz/intro2cs1/ex7r/dori.plg/final/submission
34 inflating: src/ex7.py
35 9 passed tests out of 9 in test set named 'presubmit'.
36 result_code presubmit 9 1
37 --> BEGIN TEST INFORMATION
38 Test name: addmult_1
39 Module tested: ex7
40 Function call: mult(3,0)
41 Expected return value: 0
42 More test options: {}
43 --> END TEST INFORMATION
44 *****
45 ***** There is a problem:
46 ***** The test named 'addmult_1' failed.
47 *****
48 Wrong result, input: [3, 0]:
49 expected: 0
50 actual: 3
51 result_code addmult_1 wrong 1
52 13 passed tests out of 14 in test set named 'addmult'.
53 result_code addmult 13 1
54 10 passed tests out of 10 in test set named 'iseven'.
55 result_code iseven 10 1
56 --> BEGIN TEST INFORMATION
57 Test name: logmult_1
58 Module tested: ex7
59 Function call: log_mult(3,0)
```

```

60 Expected return value: 0
61 More test options: {}
62 --> END TEST INFORMATION
63 *****
64 ***** There is a problem:
65 ***** The test named 'logmult_1' failed.
66 *****
67 Wrong result, input: [3, 0]:
68 expected: 0
69 actual: 3
70 result_code logmult_1 wrong 1
71 13 passed tests out of 14 in test set named 'logmult'.
72 result_code logmult 13 1
73 --> BEGIN TEST INFORMATION
74 Test name: ispower_f0
75 Module tested: ex7
76 Function call: is_power(0,2)
77 Expected return value: False
78 More test options: {}
79 --> END TEST INFORMATION
80 *****
81 ***** There is a problem:
82 ***** The test named 'ispower_f0' failed.
83 *****
84 Test did not complete, exited with exitcode 1.
85 This probably means your code caused an exception to be raised.
86 result_code ispower_f0 exception 1
87 23 passed tests out of 24 in test set named 'ispower'.
88 result_code ispower 23 1
89 8 passed tests out of 8 in test set named 'reverse'.
90 result_code reverse 8 1
91 8 passed tests out of 8 in test set named 'hanoi'.
92 result_code hanoi 8 1
93 10 passed tests out of 10 in test set named 'ones'.
94 result_code ones 10 1
95 --> BEGIN TEST INFORMATION
96 Test name: compare_t1
97 Module tested: ex7
98 Function call: compare_2d_lists([[[]],[[]])
99 Expected return value: True
100 More test options: {}
101 --> END TEST INFORMATION
102 *****
103 ***** There is a problem:
104 ***** The test named 'compare_t1' failed.
105 *****
106 result_code compare_t1 modified 1
107 --> BEGIN TEST INFORMATION
108 Test name: compare_t4
109 Module tested: ex7
110 Function call: compare_2d_lists([[1, 2], [4, 5, 8]], [[1, 2], [4, 5, 6]])
111 Expected return value: False
112 More test options: {}
113 --> END TEST INFORMATION
114 *****
115 ***** There is a problem:
116 ***** The test named 'compare_t4' failed.
117 *****
118 result_code compare_t4 modified 1
119 --> BEGIN TEST INFORMATION
120 Test name: compare_t5
121 Module tested: ex7
122 Function call: compare_2d_lists([[1, 2], [4, 5, 8]], [[1, 2], [4, 5, 8]])
123 Expected return value: True
124 More test options: {}
125 --> END TEST INFORMATION
126 *****
127 ***** There is a problem:

```

```

128 ***** The test named 'compare_t5' failed.
129 *****
130 result_code    compare_t5    modified    1
131 --> BEGIN TEST INFORMATION
132 Test name: compare_t6
133 Module tested: ex7
134 Function call: compare_2d_lists([[1], [2], [3], [4], [5], [6], [7], [8]], [[1], [2], [3], [4], [5], [6], [7], [8]])
135 Expected return value: True
136 More test options: {}
137 --> END TEST INFORMATION
138 *****
139 ***** There is a problem:
140 ***** The test named 'compare_t6' failed.
141 *****
142 result_code    compare_t6    modified    1
143 --> BEGIN TEST INFORMATION
144 Test name: compare_t7
145 Module tested: ex7
146 Function call: compare_2d_lists([[1], [2], [3], [4], [5], [6], [7], [9]], [[1], [2], [3], [4], [5], [6], [7], [8]])
147 Expected return value: False
148 More test options: {}
149 --> END TEST INFORMATION
150 *****
151 ***** There is a problem:
152 ***** The test named 'compare_t7' failed.
153 *****
154 result_code    compare_t7    modified    1
155 --> BEGIN TEST INFORMATION
156 Test name: compare_t8
157 Module tested: ex7
158 Function call: compare_2d_lists([], [1], [2, 3], [4, 5, 6]), [], [1], [2, 3], [4, 5, 6])
159 Expected return value: True
160 More test options: {}
161 --> END TEST INFORMATION
162 *****
163 ***** There is a problem:
164 ***** The test named 'compare_t8' failed.
165 *****
166 result_code    compare_t8    modified    1
167 --> BEGIN TEST INFORMATION
168 Test name: compare_t10
169 Module tested: ex7
170 Function call: compare_2d_lists([], [1], [2, 3], [4, 5, 6], [7, 8], [9], [], [], [1], [2, 3], [4, 5, 6], [7, 8], [9], [])
171 Expected return value: True
172 More test options: {}
173 --> END TEST INFORMATION
174 *****
175 ***** There is a problem:
176 ***** The test named 'compare_t10' failed.
177 *****
178 result_code    compare_t10    modified    1
179 --> BEGIN TEST INFORMATION
180 Test name: compare_t11
181 Module tested: ex7
182 Function call: compare_2d_lists([], [1], [2, 3], [4, 5, 6], [7, 8], [9, 10], [], [], [1], [2, 3], [4, 5, 6], [7, 8], [9])
183 Expected return value: False
184 More test options: {}
185 --> END TEST INFORMATION
186 *****
187 ***** There is a problem:
188 ***** The test named 'compare_t11' failed.
189 *****
190 result_code    compare_t11    modified    1
191 --> BEGIN TEST INFORMATION
192 Test name: compare_t14
193 Module tested: ex7
194 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,
195 Expected return value: True

```

```

196 More test options: {}
197 --> END TEST INFORMATION
198 *****
199 ***** There is a problem:
200 ***** The test named 'compare_t14' failed.
201 *****
202 result_code    compare_t14    modified    1
203 --> BEGIN TEST INFORMATION
204 Test name: compare_t15
205 Module tested: ex7
206 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,
207 Expected return value: False
208 More test options: {}
209 --> END TEST INFORMATION
210 *****
211 ***** There is a problem:
212 ***** The test named 'compare_t15' failed.
213 *****
214 result_code    compare_t15    modified    1
215 --> BEGIN TEST INFORMATION
216 Test name: compare_t16
217 Module tested: ex7
218 Function call: compare_2d_lists([[4, 5, 8], [1, 2]], [[4, 5, 8], [1, 2]])
219 Expected return value: True
220 More test options: {}
221 --> END TEST INFORMATION
222 *****
223 ***** There is a problem:
224 ***** The test named 'compare_t16' failed.
225 *****
226 result_code    compare_t16    modified    1
227 --> BEGIN TEST INFORMATION
228 Test name: compare_t17
229 Module tested: ex7
230 Function call: compare_2d_lists([[4, 5, 6], [1, 2]], [[4, 5, 8], [1, 2]])
231 Expected return value: False
232 More test options: {}
233 --> END TEST INFORMATION
234 *****
235 ***** There is a problem:
236 ***** The test named 'compare_t17' failed.
237 *****
238 result_code    compare_t17    modified    1
239 6 passed tests out of 18 in test set named 'compare'.
240 result_code    compare        6        1
241 20 passed tests out of 20 in test set named 'magic'.
242 result_code    magic          20        1
243 Running mypy
244 Success: no issues found in 2 source files
245 Finished running mypy
246
247 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 recurse
6 # STUDENTS I DISCUSSED THE EXERCISE WITH:
7 # WEB PAGES I USED:
8 # NOTES: Recursion is a must, see exercise 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 efficiently -  $O(\log(y))$ """
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 appears 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

```