

Contents

| | | |
|---|--------------------|----|
| 1 | Basic Test Results | 2 |
| 2 | board.py | 10 |
| 3 | car.py | 13 |
| 4 | game.py | 15 |

1 Basic Test Results

```
1 Sun Dec 27 20:47:46 IST 2020
2 Process Process-561:
3 Traceback (most recent call last):
4   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
5     self.run()
6   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
7     self._target(*self._args, **self._kwargs)
8   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
9     res=target(*args, **kwargs)
10  File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
11    g = Game(b)
12  File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
13    self.__filename = parameters[1] # Uncomment for Command
14  IndexError: list index out of range
15 Process Process-562:
16 Traceback (most recent call last):
17   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
18     self.run()
19   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
20     self._target(*self._args, **self._kwargs)
21   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
22     res=target(*args, **kwargs)
23   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
24     g = Game(b)
25   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
26     self.__filename = parameters[1] # Uncomment for Command
27  IndexError: list index out of range
28 Process Process-563:
29 Traceback (most recent call last):
30   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
31     self.run()
32   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
33     self._target(*self._args, **self._kwargs)
34   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
35     res=target(*args, **kwargs)
36   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
37     g = Game(b)
38   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
39     self.__filename = parameters[1] # Uncomment for Command
40  IndexError: list index out of range
41 Process Process-564:
42 Traceback (most recent call last):
43   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
44     self.run()
45   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
46     self._target(*self._args, **self._kwargs)
47   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
48     res=target(*args, **kwargs)
49   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
50     g = Game(b)
51   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
52     self.__filename = parameters[1] # Uncomment for Command
53  IndexError: list index out of range
54 Process Process-565:
55 Traceback (most recent call last):
56   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
57     self.run()
58   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
59     self._target(*self._args, **self._kwargs)
```

```

60 File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
61     res=target(*args, **kwargs)
62 File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
63     g = Game(b)
64 File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
65     self.__filename = parameters[1] # Uncomment for Command
66 IndexError: list index out of range
67 Process Process-566:
68 Traceback (most recent call last):
69     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
70         self.run()
71     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
72         self._target(*self._args, **self._kwargs)
73     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
74         res=target(*args, **kwargs)
75     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
76         g = Game(b)
77     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
78         self.__filename = parameters[1] # Uncomment for Command
79 IndexError: list index out of range
80 Process Process-567:
81 Traceback (most recent call last):
82     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
83         self.run()
84     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
85         self._target(*self._args, **self._kwargs)
86     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
87         res=target(*args, **kwargs)
88     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
89         g = Game(b)
90     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
91         self.__filename = parameters[1] # Uncomment for Command
92 IndexError: list index out of range
93 Process Process-568:
94 Traceback (most recent call last):
95     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
96         self.run()
97     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
98         self._target(*self._args, **self._kwargs)
99     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
100         res=target(*args, **kwargs)
101     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
102         g = Game(b)
103     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
104         self.__filename = parameters[1] # Uncomment for Command
105 IndexError: list index out of range
106 Process Process-569:
107 Traceback (most recent call last):
108     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
109         self.run()
110     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
111         self._target(*self._args, **self._kwargs)
112     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
113         res=target(*args, **kwargs)
114     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
115         g = Game(b)
116     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
117         self.__filename = parameters[1] # Uncomment for Command
118 IndexError: list index out of range
119 Process Process-570:
120 Traceback (most recent call last):
121     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
122         self.run()
123     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
124         self._target(*self._args, **self._kwargs)
125     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
126         res=target(*args, **kwargs)
127     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner

```

```

128     g = Game(b)
129     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
130         self.__filename = parameters[1] # Uncomment for Command
131 IndexError: list index out of range
132 Process Process-571:
133 Traceback (most recent call last):
134     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
135         self.run()
136     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
137         self._target(*self._args, **self._kwargs)
138     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
139         res=target(*args, **kwargs)
140     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
141         g = Game(b)
142     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
143         self.__filename = parameters[1] # Uncomment for Command
144 IndexError: list index out of range
145 Process Process-572:
146 Traceback (most recent call last):
147     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
148         self.run()
149     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
150         self._target(*self._args, **self._kwargs)
151     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
152         res=target(*args, **kwargs)
153     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
154         g = Game(b)
155     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
156         self.__filename = parameters[1] # Uncomment for Command
157 IndexError: list index out of range
158 Process Process-573:
159 Traceback (most recent call last):
160     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
161         self.run()
162     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
163         self._target(*self._args, **self._kwargs)
164     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
165         res=target(*args, **kwargs)
166     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
167         g = Game(b)
168     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
169         self.__filename = parameters[1] # Uncomment for Command
170 IndexError: list index out of range
171 Process Process-574:
172 Traceback (most recent call last):
173     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
174         self.run()
175     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
176         self._target(*self._args, **self._kwargs)
177     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
178         res=target(*args, **kwargs)
179     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
180         g = Game(b)
181     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
182         self.__filename = parameters[1] # Uncomment for Command
183 IndexError: list index out of range
184 Process Process-575:
185 Traceback (most recent call last):
186     File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
187         self.run()
188     File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
189         self._target(*self._args, **self._kwargs)
190     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
191         res=target(*args, **kwargs)
192     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
193         g = Game(b)
194     File "/tmp/bodek.KRNTQ/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
195         self.__filename = parameters[1] # Uncomment for Command

```

```

196 IndexError: list index out of range
197 Process Process-576:
198 Traceback (most recent call last):
199   File "/usr/lib/python3.7/multiprocessing/process.py", line 297, in _bootstrap
200     self.run()
201   File "/usr/lib/python3.7/multiprocessing/process.py", line 99, in run
202     self._target(*self._args, **self._kwargs)
203   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/autotest.py", line 74, in wrap
204     res=target(*args, **kwargs)
205   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/lib/ex9tests_full.py", line 890, in game_runner
206     g = Game(b)
207   File "/tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/testdir/src/game.py", line 26, in __init__
208     self.__filename = parameters[1] # Uncomment for Command
209 IndexError: list index out of range
210 Sun Dec 27 20:47:46 IST 2020
211 Archive: /tmp/bodek.KRNTQq/intro2cs1/ex9/tsviel/final/submission
212   inflating: src/board.py
213   inflating: src/car.py
214   inflating: src/game.py
215 16 passed tests out of 16 in test set named 'carbase'.
216 result_code   carbase   16   1
217 16 passed tests out of 16 in test set named 'carpossible'.
218 result_code   carpossible 16   1
219 32 passed tests out of 32 in test set named 'carrequire'.
220 result_code   carrequire  32   1
221 18 passed tests out of 18 in test set named 'carprivate'.
222 result_code   carprivate  18   1
223 80 passed tests out of 80 in test set named 'carmove'.
224 result_code   carmove    80   1
225 64 passed tests out of 64 in test set named 'carseq'.
226 result_code   carseq     64   1
227 6 passed tests out of 6 in test set named 'boardbase'.
228 result_code   boardbase   6   1
229 2 passed tests out of 2 in test set named 'boardprivate'.
230 result_code   boardprivate 2   1
231 145 passed tests out of 145 in test set named 'boardonecar'.
232 result_code   boardonecar 145  1
233 32 passed tests out of 32 in test set named 'boardtwocar'.
234 result_code   boardtwocar 32   1
235 138 passed tests out of 138 in test set named 'boardmovecars'.
236 result_code   boardmovecars 138 1
237 7 passed tests out of 7 in test set named 'boardpossible'.
238 result_code   boardpossible 7   1
239 --> BEGIN TEST INFORMATION
240 Test name: boardapi_move1
241 Module tested: board
242 Function call: Board()
243 Expected return value: ([True], [True], [[[4, 2], 'F1']], [['F1', [(4, 2)]]])
244 More test options: {'runseq': (['move_car'], [['F1', '1']], [<function ident at 0x7f5669ff0488>]), 'carlist': [['F1', (3, 3)]]}
245 --> END TEST INFORMATION
246 *****
247 ***** There is a problem:
248 ***** The test named 'boardapi_move1' failed.
249 *****
250 Wrong result, input: []:
251 expected: ([True], [True], [[[4, 2], 'F1']], [['F1', [(4, 2)]]])
252 actual:   ([True], [True], [[[3, 3], 'F1']], [['F1', [(3, 3)]]])
253 result_code   boardapi_move1   wrong   1
254 --> BEGIN TEST INFORMATION
255 Test name: boardapi_move2
256 Module tested: board
257 Function call: Board()
258 Expected return value: ([True, True], [False, True], [[[2, 5], 'F2']], [(3, 3), 'F1']], [['F1', [(3, 3)]], ('F2', [(2, 5)]]])
259 More test options: {'runseq': (['move_car', 'move_car'], [['F1', '3'], ('F2', '9')], [<function ident at 0x7f5669ff0488>]), 'carlist': [['F1', (3, 3)], ('F2', (2, 5))]}
260 --> END TEST INFORMATION
261 *****
262 ***** There is a problem:
263 ***** The test named 'boardapi_move2' failed.

```

```

264 *****
265 Wrong result, input: []:
266 expected: ([True, True], [False, True], [(2, 5), 'F2'], ((3, 3), 'F1')), [('F1', [(3, 3)]), ('F2', [(2, 5)])])
267 actual: ([True, True], [True, True], [(3, 3), 'F1'], ((3, 4), 'F2')), [('F1', [(3, 3)]), ('F2', [(3, 4)])])
268 result_code boardapi_move2 wrong 1
269 2 passed tests out of 4 in test set named 'boardapi'.
270 result_code boardapi 2 1
271 --> BEGIN TEST INFORMATION
272 Test name: game_1
273 Module tested: game
274 Function call: Game.play()
275 Provided input: 'G,r\nB,l\nR,d\nY,u\n!\n'
276 Expected return value: (None, [(1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((2, 5), 'Y'), ((3, 1), 'G'), ((3, 4), 'F1'))
277 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,r\nB,l\nR,d\nY,u\n!\n'}
278 --> END TEST INFORMATION
279 *****
280 ***** There is a problem:
281 ***** The test named 'game_1' failed.
282 *****
283 Test did not complete, exited with exitcode 1.
284 This probably means your code caused an exception to be raised.
285 result_code game_1 exception 1
286 --> BEGIN TEST INFORMATION
287 Test name: game_2
288 Module tested: game
289 Function call: Game.play()
290 Provided input: 'G,l\nG,r\nB,r\nB,r\nB,l\nB,l\nR,u\nR,u\nR,d\nR,d\nR,d\nY,d\nY,d\nY,u\nY,u\nY,u\n!\n'
291 Expected return value: (None, [(1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((2, 5), 'Y'), ((3, 1), 'G'), ((3, 4), 'F1'))
292 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,l\nG,r\nB,r\nB,r\nB,l\nB,l\nR,u\nR,u\nR,d\nR,d\nR,d\nY,d\nY,d\nY,u\nY,u\nY,u\n!\n'}
293 --> END TEST INFORMATION
294 *****
295 ***** There is a problem:
296 ***** The test named 'game_2' failed.
297 *****
298 Test did not complete, exited with exitcode 1.
299 This probably means your code caused an exception to be raised.
300 result_code game_2 exception 1
301 --> BEGIN TEST INFORMATION
302 Test name: game_3
303 Module tested: game
304 Function call: Game.play()
305 Provided input: 'G,r\nG,d\nB,l\nB,u\nR,d\nR,r\nY,u\nY,l\n!\n'
306 Expected return value: (None, [(1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((2, 5), 'Y'), ((3, 1), 'G'), ((3, 4), 'F1'))
307 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,r\nG,d\nB,l\nB,u\nR,d\nR,r\nY,u\nY,l\n!\n'}
308 --> END TEST INFORMATION
309 *****
310 ***** There is a problem:
311 ***** The test named 'game_3' failed.
312 *****
313 Test did not complete, exited with exitcode 1.
314 This probably means your code caused an exception to be raised.
315 result_code game_3 exception 1
316 --> BEGIN TEST INFORMATION
317 Test name: game_4
318 Module tested: game
319 Function call: Game.play()
320 Provided input: 'G,r\nB,l\nR,d\nY,u\nG,r\nG,r\nG,r\nG,r\nG,l\nG,l\n!\n'
321 Expected return value: (None, [(1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((2, 5), 'Y'), ((3, 1), 'G'), ((3, 4), 'F1'))
322 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,r\nB,l\nR,d\nY,u\nG,r\nG,r\nG,r\nG,r\nG,l\nG,l\n!\n'}
323 --> END TEST INFORMATION
324 *****
325 ***** There is a problem:
326 ***** The test named 'game_4' failed.
327 *****
328 Test did not complete, exited with exitcode 1.
329 This probably means your code caused an exception to be raised.
330 result_code game_4 exception 1
331 --> BEGIN TEST INFORMATION

```

```

332 Test name: game_5
333 Module tested: game
334 Function call: Game.play()
335 Provided input: 'G,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n'
336 Expected return value: (None, [((0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
337 More test options: {'carlist': [(['G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n')}
338 --> END TEST INFORMATION
339 *****
340 ***** There is a problem:
341 ***** The test named 'game_5' failed.
342 *****
343 Test did not complete, exited with exitcode 1.
344 This probably means your code caused an exception to be raised.
345 result_code game_5 exception 1
346 --> BEGIN TEST INFORMATION
347 Test name: game_6
348 Module tested: game
349 Function call: Game.play()
350 Provided input: 'G,l\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nR,u\nG,r\nG,r\n'
351 Expected return value: (None, [((0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
352 More test options: {'carlist': [(['G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,l\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nR,u\nG,r\nG,r\n')}
353 --> END TEST INFORMATION
354 *****
355 ***** There is a problem:
356 ***** The test named 'game_6' failed.
357 *****
358 Test did not complete, exited with exitcode 1.
359 This probably means your code caused an exception to be raised.
360 result_code game_6 exception 1
361 --> BEGIN TEST INFORMATION
362 Test name: game_7
363 Module tested: game
364 Function call: Game.play()
365 Provided input: 'W,u\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nC,r\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n'
366 Expected return value: (None, [((0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
367 More test options: {'carlist': [(['G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'W,u\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nC,r\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n')}
368 --> END TEST INFORMATION
369 *****
370 ***** There is a problem:
371 ***** The test named 'game_7' failed.
372 *****
373 Test did not complete, exited with exitcode 1.
374 This probably means your code caused an exception to be raised.
375 result_code game_7 exception 1
376 --> BEGIN TEST INFORMATION
377 Test name: game_8
378 Module tested: game
379 Function call: Game.play()
380 Provided input: 'R,z\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nY,x\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n'
381 Expected return value: (None, [((0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
382 More test options: {'carlist': [(['G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'R,z\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nY,x\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\n')}
383 --> END TEST INFORMATION
384 *****
385 ***** There is a problem:
386 ***** The test named 'game_8' failed.
387 *****
388 Test did not complete, exited with exitcode 1.
389 This probably means your code caused an exception to be raised.
390 result_code game_8 exception 1
391 --> BEGIN TEST INFORMATION
392 Test name: game_1b
393 Module tested: game
394 Function call: Game.play()
395 Provided input: 'G,r\nB,l\nR,d\nY,u\n!\n'
396 Expected return value: (None, [((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((2, 5), 'Y'), ((3, 1), 'G'), ((3, 2), 'G')])
397 More test options: {'carlist': [(['G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,r\nB,l\nR,d\nY,u\n!\n')}
398 --> END TEST INFORMATION
399 *****

```



```

468 Module tested: game
469 Function call: Game.play()
470 Provided input: 'G,l\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nR,u\nG,r\nG,r\n'
471 Expected return value: (None, [(0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
472 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'G,l\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nR,u\nG,r\nG,r\n'}
473 --> END TEST INFORMATION
474 *****
475 ***** There is a problem:
476 ***** The test named 'game_6b' failed.
477 *****
478 Test did not complete, exited with exitcode 1.
479 This probably means your code caused an exception to be raised.
480 result_code game_6b exception 1
481 --> BEGIN TEST INFORMATION
482 Test name: game_7b
483 Module tested: game
484 Function call: Game.play()
485 Provided input: 'W,u\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nC,r\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nG,r\n'
486 Expected return value: (None, [(0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
487 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'W,u\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nC,r\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nG,r\n'}
488 --> END TEST INFORMATION
489 *****
490 ***** There is a problem:
491 ***** The test named 'game_7b' failed.
492 *****
493 Test did not complete, exited with exitcode 1.
494 This probably means your code caused an exception to be raised.
495 result_code game_7b exception 1
496 --> BEGIN TEST INFORMATION
497 Test name: game_8b
498 Module tested: game
499 Function call: Game.play()
500 Provided input: 'R,z\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nY,x\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nG,r\n'
501 Expected return value: (None, [(0, 6), 'R'), ((1, 1), 'B'), ((1, 2), 'B'), ((1, 3), 'B'), ((1, 4), 'B'), ((1, 5), 'Y'), ((1, 6), 'Y'))
502 More test options: {'carlist': [('G', 2, (3, 0), 1), ('B', 4, (1, 2), 1), ('R', 3, (2, 6), 0), ('Y', 2, (3, 5), 0)], 'input': 'R,z\nG,r\nB,l\nR,d\nY,u\nY,u\nR,u\nY,x\nR,u\nR,u\nG,r\nG,r\nG,r\nG,r\nG,r\n'}
503 --> END TEST INFORMATION
504 *****
505 ***** There is a problem:
506 ***** The test named 'game_8b' failed.
507 *****
508 Test did not complete, exited with exitcode 1.
509 This probably means your code caused an exception to be raised.
510 result_code game_8b exception 1
511 0 passed tests out of 16 in test set named 'game'.
512 result_code game 0 1
513 TESTING COMPLETED

```

2 board.py

```
1  BOARD_RANGE = range(7)
2  EMPTY = "_"
3  EXIT = "E"
4  VERTICAL, HORIZONTAL = 0, 1
5  MOVE_UP, MOVE_DOWN, MOVE_LEFT, MOVE_RIGHT = "u", "d", "l", "r"
6  # Message text:
7  INVALID_COORDINATE = "You had entered an invalid Coordinate"
8  CAUSE_THE_CAR = "Causes the car to go"
9
10
11 class Board:
12     """
13     The class creates a board for a new game and updates it accordingly to the
14     users input
15     """
16
17     def __init__(self):
18         initiate_board = []
19         self.__cars_cache = []
20         for j in BOARD_RANGE:
21             initiate_board.append([EMPTY for i in BOARD_RANGE])
22         initiate_board[self.target_location()[0]].insert(
23             self.target_location()[1], EXIT)
24         self.__board = initiate_board
25
26     def __str__(self):
27         """
28         This function is called when a board object is to be printed.
29         :return: A string of the current status of the board
30         """
31         return '\n'.join(['\t'.join([str(cell) for cell in row])
32                             for row in self.__board])
33
34     def cell_list(self):
35         """ This function returns the coordinates of cells in this board
36         :return: list of coordinates
37         """
38         result = []
39         for i in BOARD_RANGE:
40             for j in BOARD_RANGE:
41                 result.append((i, j))
42         result.append(self.target_location())
43         return result
44
45     def possible_moves(self):
46         """ This function returns the legal moves of all cars in this board
47         :return: list of tuples of the form (name,movekey,description)
48                 representing legal moves
49         """
50         # From the provided example car_config.json file,
51         # the return value could be
52         # [('O','d',"some description"),
53         # ('R','r',"some description"),
54         #
55
56         moves_list = []
57         for single_car in self.__cars_cache:
58             for move in single_car.possible_moves().keys():
59                 next_cell = single_car.movement_requirements(move)[0]
```

```

60         if next_cell in self.cell_list() and self.cell_content(
61             next_cell) is None:
62             moves_list.append((single_car.get_name(), move,
63                               (single_car.possible_moves())[move]))
64     return moves_list
65
66 def target_location(self):
67     """
68     This function returns the coordinates of the location which
69     is to be filled for victory.
70     :return: (row,col) of goal location
71     """
72     return 3, 7
73
74 def cell_content(self, coordinate):
75     """
76     Checks if the given coordinates are empty.
77     :param coordinate: tuple of (row,col) of the coordinate to check
78     :return: The name if the car in coordinate, None if empty
79     """
80     if coordinate in self.cell_list():
81         if self.__board[coordinate[0]][coordinate[1]] == EMPTY or \
82            self.__board[coordinate[0]][coordinate[1]] == EXIT:
83             return None
84         else:
85             return self.__board[coordinate[0]][coordinate[1]]
86     else:
87         return INVALID_COORDINATE
88
89 def check_intersections(self, coordinates):
90     """
91     The functions checks if a given car, intersecting another car by
92     checking the intended cells
93     :param coordinates:
94     :return: False for yes, True for no
95     """
96     for coordinate in coordinates:
97         if self.cell_content(coordinate) is not None:
98             return False
99     return True
100
101 def check_car_exist(self, car_name):
102     """
103     The function validate if the car is in the game already
104     :param car_name: The name of the car we want to check
105     :return: False for yes, True for name
106     """
107     for car in self.__cars_cache:
108         if car_name == car.get_name():
109             return True # Car exists
110     return False # Car do not exist
111
112 def check_valid_moves(self, coordinates):
113     """
114     Check if the car is in the range of the board
115     :param coordinates:
116     :return: True if yes, No if it flies to the moon
117     """
118     for coordinate in coordinates: # Test if car in the range of the board
119         if coordinate not in self.cell_list():
120             return False
121     return True
122
123 def add_car(self, car):
124     """
125     Adds a car to the game.
126     :param car: car object of car to add
127     :return: True upon success. False if failed

```

```

128         """
129         car_name = car.get_name()
130         coordinates = car.car_coordinates()
131         if self.check_car_exist(car_name): # if Car exist
132             return False
133         if not self.check_valid_moves(coordinates):
134             return False
135         if not self.check_intersections(coordinates):
136             return False
137         self.__cars_cache.append(car)
138         for cell in car.car_coordinates():
139             self.__board[cell[0]][cell[1]] = car.get_name()
140         return True
141
142     def __update_board(self, car, old_tail):
143         """Update board according to old tail(tuple) of a car"""
144         for cell in car.car_coordinates():
145             self.__board[cell[0]][cell[1]] = car.get_name()
146             self.__board[old_tail[0][0]][old_tail[0][1]] = EMPTY
147         return True
148
149     def __make_car_move(self, movekey, car, old_tail):
150         """Make car object move to movekeys direction using old tail tuple"""
151         if movekey == MOVE_UP: # Moves the car upwards
152             old_tail.append(car.car_coordinates()[-1]) # Old Coords
153             car.move(MOVE_UP) # Make a move
154             self.__update_board(car, old_tail) # Update the board
155         if movekey == MOVE_DOWN: # Moves the car downward
156             old_tail.append(car.car_coordinates()[0])
157             car.move(MOVE_DOWN) # Make a move
158             self.__update_board(car, old_tail) # Update the board
159         if movekey == MOVE_RIGHT: # Moves the car in right direction
160             old_tail.append(car.car_coordinates()[0]) # Old Coords
161             car.move(MOVE_RIGHT) # Make a move
162             self.__update_board(car, old_tail) # Update the board
163         if movekey == MOVE_LEFT: # Moves the car in left direction
164             old_tail.append(car.car_coordinates()[-1]) # Old coords
165             car.move(MOVE_LEFT) # Make a move
166             self.__update_board(car, old_tail) # Update the board
167
168     def move_car(self, name, movekey):
169         """
170         moves car one step in given direction.
171         :param name: name of the car to move
172         :param movekey: Key of move in car to activate
173         :return: True upon success, False otherwise
174         """
175         old_tail = [] # Cache for old coordinates
176         # Check if given car name is in possible moves list:
177         # If all is cool make the appropriate move with for loop
178         if len(self.__cars_cache) == 0:
179             return False
180         possible_move = [item[1] if item[0] == name else None
181                          for item in self.possible_moves()]
182         if movekey not in possible_move:
183             return False
184
185         for car in self.__cars_cache:
186             # Checks for the function:
187             if name == car.get_name(): # Check if we are talking about the
188                 self.__make_car_move(movekey, car, old_tail)
189         return True

```

3 car.py

```
1  VERTICAL, HORIZONTAL = 0, 1
2  MOVE_UP, MOVE_DOWN, MOVE_LEFT, MOVE_RIGHT = "u", "d", "l", "r"
3  VALID_NAMES = {"Y", "B", "O", "W", "G", "R"}
4
5  # Message text:
6  INVALID_ORIENTATION = "You had entered an invalid orientation"
7  CAUSE_THE_CAR = "Causes the car to go"
8  UP = "up"
9  DOWN = "down"
10 LEFT = "left"
11 RIGHT = "right"
12
13
14 class Car:
15     """
16     Add class description here
17     """
18
19     def __init__(self, name, length, location, orientation):
20         """
21         A constructor for a Car object
22         :param name: A string representing the car's name
23         :param length: A positive int representing the car's length.
24         :param location: A tuple representing the car's head (row, col) location
25         :param orientation: One of either 0 (VERTICAL) or 1 (HORIZONTAL)
26         """
27         self.__name = name
28         self.__length = length
29         self.__location = location
30         self.__orientation = orientation
31         self.__location_h = location[1]
32         self.__location_v = location[0]
33
34     def car_coordinates(self):
35         """
36         :return: A list of coordinates(tuples) the car is in
37         """
38         output = []
39         if self.__orientation is VERTICAL: # Case for Vertical Orientation
40             for v in range(self.__length):
41                 output.append((self.__location_v + v, self.__location_h))
42         elif self.__orientation is HORIZONTAL: # Case for Vertical Orientation
43             for h in range(self.__length):
44                 output.append((self.__location_v, self.__location_h + h))
45         else:
46             return INVALID_ORIENTATION
47         return output
48
49     def possible_moves(self):
50         """
51         :return: A dictionary of strings describing possible movements
52         permitted by this car.
53         """
54         result = {}
55         if self.__orientation is VERTICAL: # Case for Vertical Orientation
56             result['u'] = CAUSE_THE_CAR + UP
57             result['d'] = CAUSE_THE_CAR + DOWN
58         if self.__orientation is HORIZONTAL: # Case for Vertical Orientation
59             result['l'] = CAUSE_THE_CAR + LEFT
```

```

60         result['r'] = CAUSE_THE_CAR + RIGHT
61     return result
62
63     def movement_requirements(self, movekey):
64         """
65         :param movekey: A string representing the key of the required move.
66         :return: A list of cell locations which must be empty in order for
67                 this move to be legal.
68         """
69         result = []
70         if movekey == MOVE_UP:
71             result.append((int(self.car_coordinates()[0][0]) - 1,
72                           self.car_coordinates()[0][1]))
73         if movekey == MOVE_DOWN:
74             result.append((int(self.car_coordinates()[-1][0]) + 1,
75                           self.car_coordinates()[-1][1]))
76         if movekey == MOVE_LEFT:
77             result.append((self.car_coordinates()[0][0],
78                           int(self.car_coordinates()[0][1] - 1))
79         if movekey == MOVE_RIGHT:
80             result.append((self.car_coordinates()[-1][0],
81                           int(self.car_coordinates()[-1][1] + 1))
82         return result
83
84     def move(self, movekey):
85         """
86         :param movekey: A string representing the key of the required move.
87         :return: True upon success, False otherwise
88         """
89
90         # Tests for restricted moves
91         if self.__orientation == HORIZONTAL \
92             and movekey not in [MOVE_RIGHT, MOVE_LEFT]:
93             return False
94         if self.__orientation == VERTICAL \
95             and movekey not in [MOVE_UP, MOVE_DOWN]:
96             return False
97
98         # Moves the car
99         if movekey == MOVE_RIGHT: # Move Right
100             self.__location = (self.__location_v, self.__location_h + 1)
101         if movekey == MOVE_LEFT: # Move Left
102             self.__location = (self.__location_v, self.__location_h - 1)
103         if movekey == MOVE_UP: # Move up
104             self.__location = (self.__location_v - 1, self.__location_h)
105         if movekey == MOVE_DOWN: # Move Down
106             self.__location = (self.__location_v + 1, self.__location_h)
107         self.__location_v, self.__location_h = self.__location
108         return True # If everything is cool, return
109
110     def get_name(self):
111         """
112         :return: The name of this car.
113         """
114         return self.__name

```

4 game.py

```
1  from board import *
2  from car import *
3  from sys import argv as parameters
4  from sys import exit as exit_game
5  from helper import *
6  # Constants
7  VERTICAL, HORIZONTAL = 0, 1
8  MOVE_UP, MOVE_DOWN, MOVE_LEFT, MOVE_RIGHT = "u", "d", "l", "r"
9  VALID_NAMES = {"Y", "B", "O", "W", "G", "R"}
10 GAME_OVER = "!"
11 VALID_DIRECTIONS = {MOVE_UP, MOVE_DOWN, MOVE_LEFT, MOVE_RIGHT}
12 MAKE_A_MOVE = "Make your move (ex: 0,u):\n"
13
14
15 class Game:
16     """
17     Add class description here
18     """
19     def __init__(self, board):
20         """
21         Initialize a new Game object.
22         :param board: An object of type board
23         """
24         # You may assume board follows the API
25         self.__board = board
26         self.__filename = parameters[1] # Uncomment for Command
27         # self.__filename = DEFAULT_JSON_PATH_NAME # Uncomment for debugging
28
29         file = load_json(self.__filename)
30         keys = list(file.keys())
31         values = list(file.values())
32         self.__car_list = []
33         for i in range(len(keys)): # Create Cars
34             car = Car(keys[i], values[i][0], values[i][1], values[i][2])
35             if car.get_name() not in VALID_NAMES:
36                 continue
37             self.__car_list.append(car)
38             i += 1
39         for car in self.__car_list: # add all cars to board
40             self.__board.add_car(car)
41         print(self.__board)
42         self.play()
43
44     def __check_move(self, car_name, movekey):
45         """Check possible moves for car"""
46         possible_move = [item[1] if item[0] == car_name else None
47                          for item in self.__board.possible_moves()]
48         if movekey not in possible_move:
49             return False
50         return True
51
52     def __single_turn(self, move):
53         """
54         A single turn recursive function
55         :return True if win, False if not
56         """
57         name, movekey = move[0], move[1]
58         if movekey not in VALID_DIRECTIONS:
59             print("Invalid move key, valid choices are ",
```

```

60         "," .join(VALID_DIRECTIONS))
61     return
62 if name not in VALID_NAMES:
63     print("You are trying to move an invalid car")
64     return
65 if not self.__check_move(name, movekey):
66     print("Can't move car in this direction")
67     return
68 self.__board.move_car(str(move[0]).upper(), str(move[1]).lower())
69
70 def __is_win(self):
71     """
72     :param target: tuple - winning coordinate
73     :return: true if win, false if not
74     """
75     row, col = self.__board.target_location()
76     target = self.__board.cell_content((row,col))
77     prev = self.__board.cell_content((row, col-1))
78     if target and prev:
79         return True
80     return False
81
82 def play(self):
83     """The main driver of the Game. Manages the game until completion"""
84     print("Welcome to Rush-hour")
85     user_input = input(MAKE_A_MOVE)
86     print(self.__board)
87     print(user_input)
88     while user_input != GAME_OVER:
89         if user_input == "" or not(len(user_input) in range(2, 4)):
90             print("Please provide a move and car to move")
91             print(self.__board)
92             user_input = input(MAKE_A_MOVE)
93             continue
94         move = tuple(user_input.split(","))
95         self.__single_turn(move)
96         print(self.__board)
97         if self.__is_win():
98             # if __win returns print win message and break the game loop
99             print("You win")
100             break
101         user_input = input(MAKE_A_MOVE) # Wait for next users input
102         # If nothing breaks the turn, iterate to next one
103
104     # If game loop was terminated
105     print("Thank you and Goodbye")
106     return
107
108
109 def main():
110     """
111     This is the main function of the program which calls and combines all the
112     functions of the program.
113     This function of the program handles its integrity.
114     :return: None if fail to run game
115     """
116     try:
117         if len(parameters) <= 1 or len(parameters) > 2:
118             # if not given exact amount of parameters
119             raise ValueError
120         new_board = Board()
121         Game(new_board)
122     except ValueError: # Should be Manually Raised
123         print("Not Enough Arguments was entered. make sure to enter correct",
124             "filename and path")
125         return
126     except FileNotFoundError: # If config file not found except FileNotFoundError
127         print("Python FileNotFoundError")

```



```
128         print("Can't load game because car config File was not found")
129         return
130     exit_game() # sys.exit
131
132
133 if __name__ == "__main__":
134     main()
```