

Problem Set 7, Problems 0 and 1

Problem 0: Reading and response

Put your response to the reading below.

IMPORTANT: Your entire response should fit on this page.

The most obvious advantage to evolving programs to solve programs instead of writing it to perform specific tasks is that it is able to be adapted in many different situations. When evolved, the program is able to handle unforeseen tasks to a certain extent through its adaptations built from evolution. However, there are problems with using evolution to create programs. One thing is that it isn't as efficient as code created for a specific tasks. This is because the evolved code has to be able to handle a wide range of problems at the cost of being efficient at specific tasks. This is common within nature as that evolution usually comes at a compromise. On top of that, if a pathway to an efficient solution isn't present in the beginning, then the program might develop into a program that eventually doesn't work. This is because evolution and natural selection can only work on existing traits. If this is the case and the programs fails at some point, then it would be very difficult to fix it as well. This is because individual pieces of code may not function within a vacuum and needs the whole in order to work properly. Comparatively, code created for specific tasks can be adapted fairly easily as edits to it might not destroy everything else completely (for the most part).

Problem 1: Working with nested loops and 2-D lists

IMPORTANT: This heading should appear at the very top of the second page.

1-1

| x | range(1, x) | y | value printed |
|---|-------------|---|---------------|
| 2 | [1] | 1 | 3 |
| 4 | [1,2,3] | 1 | 5 |
| | | 2 | 6 |
| | | 3 | 7 |
| 6 | [1,2,3,4,5] | 1 | 7 |
| | | 2 | 8 |
| | | 3 | 9 |
| | | 4 | 10 |
| | | 5 | 11 |
| | | | |
| | | | |
| | | | |
| | | | |
| | | | |

1-2

a) `twoD[2][1] = 16`

b)

```
for r in range(len(twoD)):
    print(twoD[r][-1])
```

c)

```
for x in range(len(twoD)):
    print(twoD[r][r])
```