

# 07 Checkpoint: Lists

## Purpose

---

Reinforce your understanding that numbers are passed to a function by value and lists are passed to a function by reference.

## Helpful Documentation

---

Within a Python program, when a number is passed as an argument into a function, the computer copies the number in the argument into the parameter. In other words, the parameter gets a copy of the value that is in the argument. However, when a list is passed as an argument into a function, the computer does not copy the list. Instead, the computer passes a reference to the list into the function. This means the argument and parameter refer to the same list. If a called function changes a list that was passed into the function, this will change the list in the calling function because both the argument and parameter refer to the same list.

## Assignment

---

Do the following:

1. Download and save the [pass\\_args.py](#) Python file and then open it in VS Code.
2. Run the `pass_args.py` program on your computer. Examine the code and the output and answer the following three questions:
  - a. In `main`, there are two local variables named `x` and `lx`. At line 14, both of these local variables are passed to the `modify_args` function. After the call to `modify_args` finished, which of `main`'s two local variables did the `modify_args` function change?
    - i. `x`
    - ii. `lx`
    - iii. both of them
    - iv. neither of them
  - b. Why is the `modify_args` function NOT able to change `main`'s local variable `x`?
    - i. Because `x` is a number and therefore passed to a function by value
    - ii. Because `x` is a list and therefore passed to a function by reference
  - c. Why is the `modify_args` function able to change `main`'s local variable `lx`?
    - i. Because `lx` is a number and therefore passed to a function by value
    - ii. Because `lx` is a list and therefore passed to a function by reference

## Sample Run

---

```
> python pass_args.py
main()
  Before calling modify_args(): x 5  lx [7, -2]
  modify_args(n, alist)
    Before changing n and alist: n 5  alist [7, -2]
    After changing n and alist:  n 6  alist [7, -2, 4]
  After calling modify_args():  x 5  lx [7, -2, 4]
```

## Submission

When complete, report your progress in the associated I-Learn quiz.