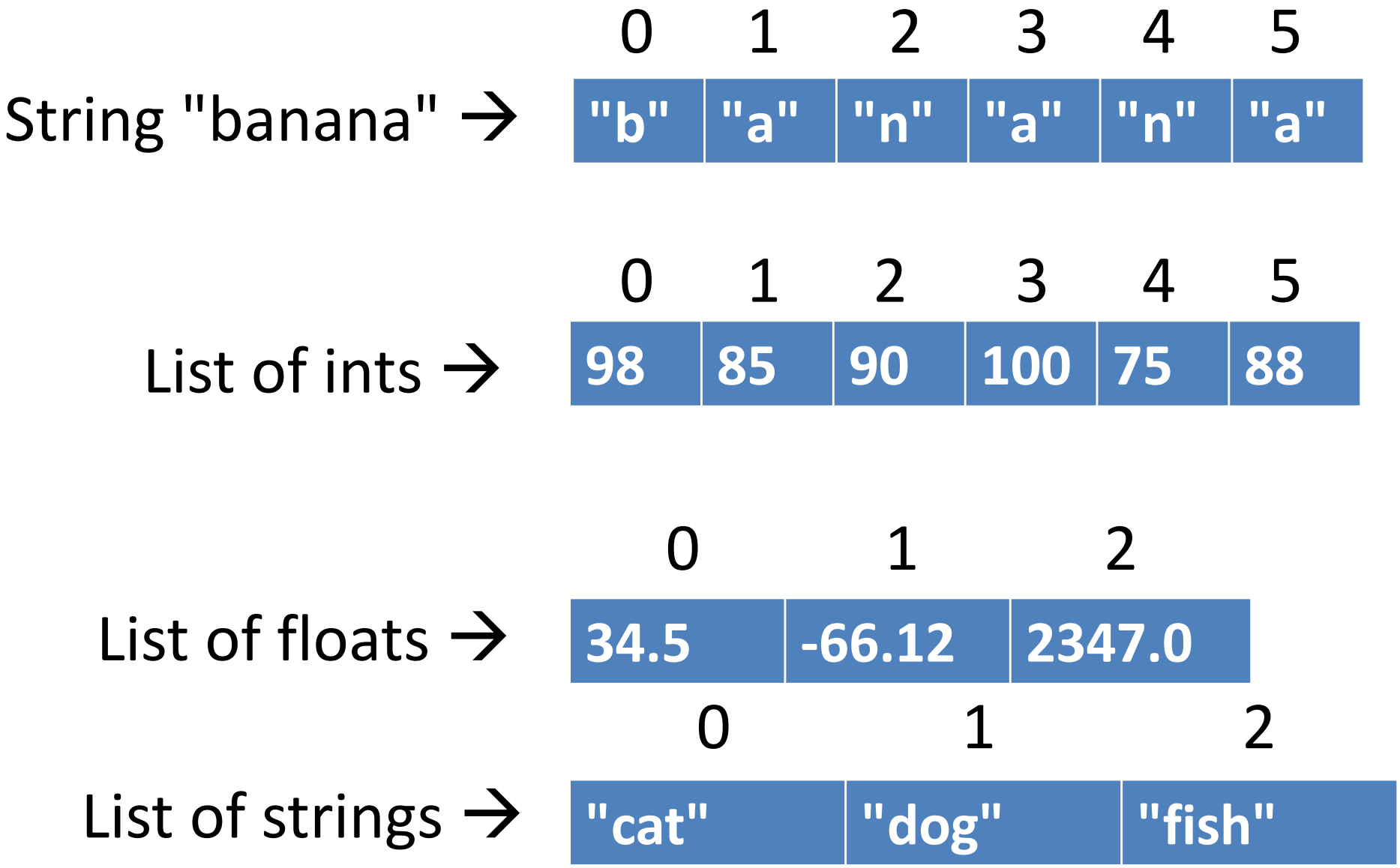
Lists II

Why use lists?

- Lists exist so programmers can store multiple related variables together, using a single name.
- Instead of having ten variables named num1, num2, num3...etc, have a list variable indexed by numbers:
  - numbers[0], numbers[1], numbers[2], ...etc


0 1 2 3 4 5
String "banana" →
"b" "a" "n" "a" "n" "a"
0 1 2 3 4 5
List of ints →
98 85 90 100 75 88
0 1 2
List of floats →
34.5 -66.12 2347.0
0 1 2
List of strings →
"cat" "dog" "fish"

```
lstA = [5, 10, 15, 20]   # make a list
print(lstA[0])           # prints 5
print(lstA[-1])          # prints 20
print(lstA[1:3])      # prints [10,15]

lstB = []              # make empty list
lstB.append(40)        # lstB is now [40]
lstB.append(60)        # lstB is [40, 60]

lstB[0] = 50           # lstB is [50, 60]
lstB[1] += 5           # lstB is [50, 65]
```

Loops over lists

- Just like loops over strings!

```
for pos in range(0, len(str_var)):
   do something

for pos in range(0, len(list_var)):
   do something
```

The most basic list loop

```
lst = ...a list of strings...

for pos in range(0, len(lst)):
  print(lst[pos])
```

What does this do?

```
lst = ...a list of ints...

for pos in range(0, len(lst)):
  if lst[pos] > 10:
    print(lst[pos])
```

Print all ints in the list that are greater than 10.

What does this do?

```
lst = ...a list of strings...

for pos in range(0, len(lst)):
  if lst[pos].startswith("a"):
    print(lst[pos])
```

Print all strings in the list that start with a lowercase "a".

What does this do?

```
lst = ...a list of strings...

for pos in range(0, len(lst)):
  if lst[pos][0].islower():
    print(lst[pos])
```

Print all strings in the list that start with a lowercase letter.

What does this do?

```
lst = ...a list of ints...

for pos in range(0, len(lst)):
  if lst[pos] == pos:
    print(lst[pos])
```

Print all ints in the list whose value is the same as their index.

What does this do?

```
lst = ...a list of ints...

for pos in range(0, len(lst)):
  lst[pos] *= 2
```

Doubles all items in the list.

`find()` doesn't exist for lists

- **`list_var.index(item)`**
- Searches left to right, returns position where found, but crashes if not found.
- Let's build an algorithm that replicates **`find()`**, but works for lists (returns -1 if not found).
- Assume we don't have access to the **`index`** function.

Class work

- Write a function to find the largest integer in a list and print it out.
- Write a function that takes a list of numbers and prints out sums of adjacent pairs of numbers in the list (don't use the sliding window; use indices).
- Write a function that takes a list of strings and prints out all the strings that start and end with the same letter.
- Write a function that takes a list of strings and ***RETURNS*** a list of all the strings that have more a's than b's.
- Write a function that takes a list and shifts all the elements in the list one spot to the left, without using slices! (the left-most element disappears)
  - So [1, 2, 3, 4, 5] turns into [2, 3, 4, 5, 5]

– Can you make a function that shifts to the right?