

## Discussion 9B: Midterm Review

### Recursion

1. In the space below, write a block that recursively converts binary to decimal. You may find the following blocks helpful:

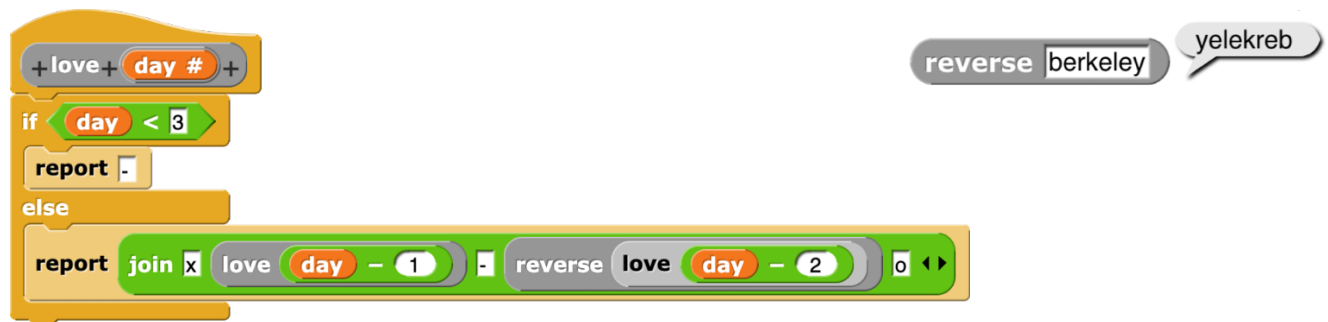


```
convert (bitstring) to decimal
```

What is the runtime of your block?

- Constant    
  Logarithmic    
  Linear    
  Quadratic    
  Exponential

2. You have just gotten into a relationship and have written the block `love` to track your affection through your relationship (x = kisses, o = hugs, - = hanging out). The reverse block, which reverses its input text, has been provided to you.



- a. What is `love 3` ? \_\_\_\_\_
- b. What is `love 4` ? \_\_\_\_\_
- c. What are the first three and last three letters of `love 9999` ? \_\_\_\_\_, \_\_\_\_\_
- d. Which of the following are possible?
  - Hanging out four times in a row (i.e. “----”)
  - Hug immediately followed by a kiss (i.e. “ox”)
  - Hug twice in a row (i.e. “oo”)

### 3. Short Changed

When you get change back, you don't care about how many possibilities there are (silly **Count Change**). What you do want to know is what are the fewest number of coins you need to carry in your pocket. Write the **Short Changed** block that will return the minimum number of coins you can make change with.



Hint 1: We don't need to use a minimum block.

Hint 2: This block has a few similarities with **Count Change**, but has its differences as well.

Hint 3: This block can be solved in  $101_2 + (0x7B - 0b11) \div 100_2 \div F_{16}$  lines of code.

## Mutability

1. We've created the copy of block to make a copy of a list, but it doesn't work as we expected. Write out what the sprite would say after the following blocks of code have been run.

a.

```

script variables x y
set x to list 1 2
set y to copy of x
add 3 to y
say x for 2 secs
  
```

```

+copy+of+ list : +
script variables new list
set new list to list
for each item in list
  add item to new list
report new list
  
```

b.

```

script variables x y
set x to list list 1 2
set y to copy of x
add 3 to item 1 of y
say x for 2 secs
  
```

2. We've created a block to square all the numbers in a list, as shown to the right. However, our tests are giving us puzzling results. Answer the questions below to help us understand the behavior of our block.

```

+square+numbers+in+ list : +
script variables new list
set new list to list
for i = 1 to length of new list
  replace item i of new list with
    item i of list × item i of list
report new list
  
```

a. When we run the script below, the sprite first says "true", but then says "false." Why does this happen, and how can we fix it?

```
script variables test list ▶
set test list to list 1 2 3 ◀▶
say
test square numbers in [ ] ▶ w/ input test list ◀▶ expecting output
list 1 4 9 ◀▶
for 2 secs
say
test square numbers in [ ] ▶ w/ input test list ◀▶ expecting output
list 1 4 9 ◀▶
for 2 secs
```

---

---

b. Assuming we haven't changed the square numbers block, what do we expect the sprite to say when we run the code below? Why is this different from part a?

```
say
test square numbers in [ ] ▶ w/ input list 1 2 3 ◀▶ ◀▶ expecting output
list 1 4 9 ◀▶
for 2 secs
say
test square numbers in [ ] ▶ w/ input list 1 2 3 ◀▶ ◀▶ expecting output
list 1 4 9 ◀▶
for 2 secs
```

---

---