**Loose ends**

Plan for today:
1. Lists in algorithms
2. Diagonalization and halting
3. Paradox and provability
4. Homework review

**We’ve seen lots of lists**

In finding prime numbers
- Worked through a list of numbers

In thinking about programs
- Considered lists of functions, tuples

And we’ll have more examples
- Sorting, searching, …
Concrete lists

You can make lists in Scratch

• (Demo)

Key operations:

• Get length of a list (number of elements)
• Get specific element by number
• Update value of a specific element
• Add element to the end of the list

Lists, loops and variables

A very very common pattern:

• set index = 1
• repeat until index > length of wholelist
  • value = item index of wholelist
  • compute with value
  • change index by 1

Called: “operating over a list”
Check your understanding

Given list Ws =
(the, quick, brown, fox, jumps, over, the, lazy, dog)

What is

set index = 1
repeat until index > length of wholelist
• value = item index of wholelist
• say value
• change index by 2

A. the brown jumps the dog  C. brown jumps the dog
B. quick fox over lazy      D. the quick brown fox jumps
What is

set index = 2
repeat until index > length of wholelist
  • value = item index of wholelist
  • say value
  • change index by 2

A. the brown jumps the dog  C. brown jumps the dog
B. quick fox over lazy       D. the quick brown fox jumps

What is

set index = length of wholelist
repeat until index < 1
  • value = item index of wholelist
  • say value
  • change index by -1

A. lazy lazy lazy lazy       B. dog lazy the over jumps fox brown quick the
C. lazy the over jumps fox brown quick