# Iteration

# Announcements

Return

# Return Statements

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

   f(x) for user-defined function f: switch to a new environment; execute f's body

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

   f(x) for user-defined function f: switch to a new environment; execute f's body

   **return** statement within f: switch back to the previous environment; f(x) now has a value

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

  f(x) for user-defined function f: switch to a new environment; execute f's body

  **return** statement within f: switch back to the previous environment; f(x) now has a value

Only one return statement is ever executed while executing the body of a function

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

   f(x) for user-defined function f: switch to a new environment; execute f's body

   **return** statement within f: switch back to the previous environment; f(x) now has a value

Only one return statement is ever executed while executing the body of a function

```python
def end(n, d):
    """Print the final digits of N in reverse order until D is found.

    >>> end(34567, 5)
    7
    6
    5
    """
```

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

   f(x) for user-defined function f: switch to a new environment; execute f's body

  **return** statement within f: switch back to the previous environment; f(x) now has a value

Only one return statement is ever executed while executing the body of a function

```python
def end(n, d):
    """Print the final digits of N in reverse order until D is found.

    >>> end(34567, 5)
    7
    6
    5
    """
    while n > 0:
        last, n = n % 10, n // 10
        print(last)
```

# Return Statements

A return statement completes the evaluation of a call expression and provides its value

  f(x) for user-defined function f: switch to a new environment; execute f's body

  **return** statement within f: switch back to the previous environment; f(x) now has a value

Only one return statement is ever executed while executing the body of a function

```python
def end(n, d):
    """Print the final digits of N in reverse order until D is found.

    >>> end(34567, 5)
    7
    6
    5
    """
    while n > 0:
        last, n = n % 10, n // 10
        print(last)
        if d == last:
            return None
```

## Return Statements

A return statement completes the evaluation of a call expression and provides its value

  f(x) for user-defined function f: switch to a new environment; execute f's body

  return statement within f: switch back to the previous environment; f(x) now has a value

Only one return statement is ever executed while executing the body of a function

```python
def end(n, d):
    """Print the final digits of N in reverse order until D is found.

    >>> end(34567, 5)
    7
    6
    5
    """
    while n > 0:
        last, n = n % 10, n // 10
        print(last)
        if d == last:
            return None
```
(Demo)

# Self-Reference

(Demo)

# Returning a Function Using Its Own Name

```
1   def print_sums(n):
2       print(n)
3       def next_sum(k):
→ 4           return print_sums(n+k)
5       return next_sum
6
➡ 7   print_sums(1)(3)(5)
```

Frames

Global frame

print_sums

f1: print_sums [parent=Global]
n    1
next_sum
Return
value

f2: next_sum [parent=f1]
k    3
Return
value

f3: print_sums [parent=Global]
n    4
next_sum
Return
value

f4: next_sum [parent=f3]
k    5

f5: print_sums [parent=Global]
n    9
next_sum
Return
value

Objects

func print_sums(n) [parent=Global]

func next_sum(k) [parent=f1]

func next_sum(k) [parent=f3]

func next_sum(k) [parent=f5]

# Function Example: Sounds

# WAV Files

https://en.wikipedia.org/wiki/Triangle_wave
https://en.wikipedia.org/wiki/Sampling_(signal_processing)

# WAV Files

The Waveform Audio File Format
encodes a sampled sound wave

https://en.wikipedia.org/wiki/Triangle_wave
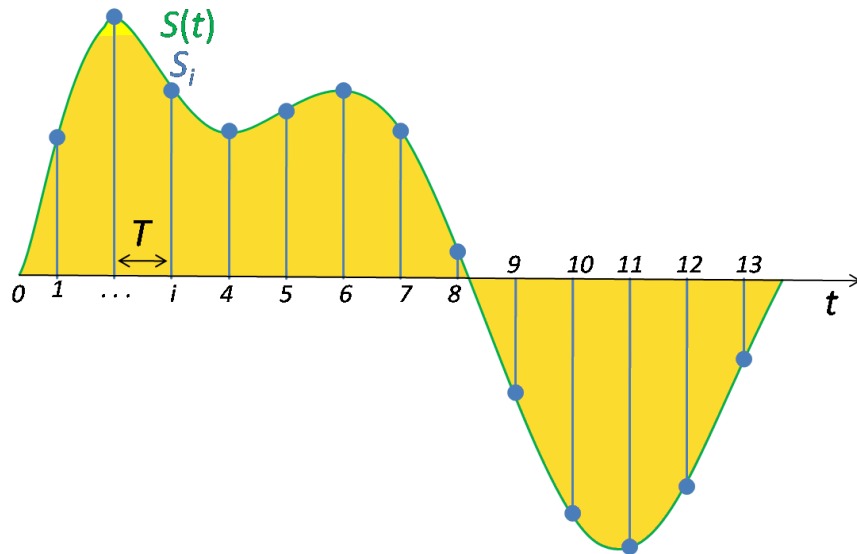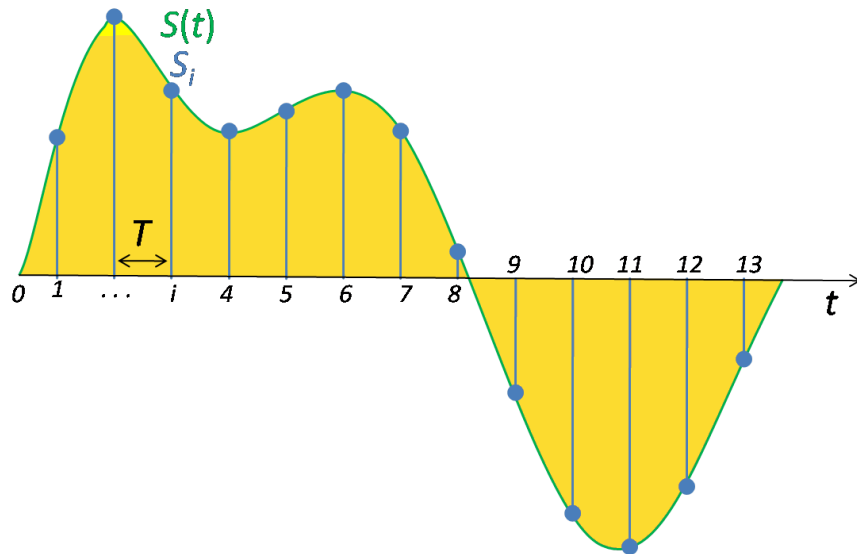https://en.wikipedia.org/wiki/Sampling_(signal_processing)

# WAV Files

The Waveform Audio File Format
encodes a sampled sound wave

# WAV Files
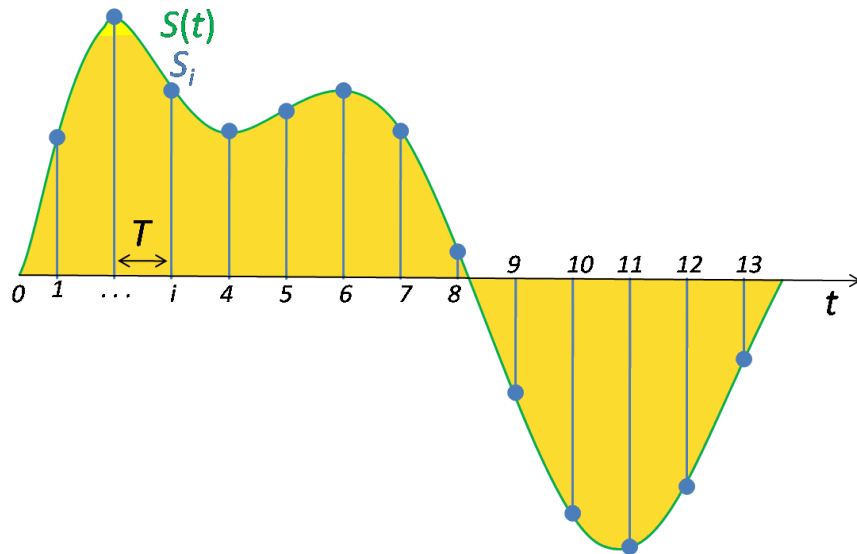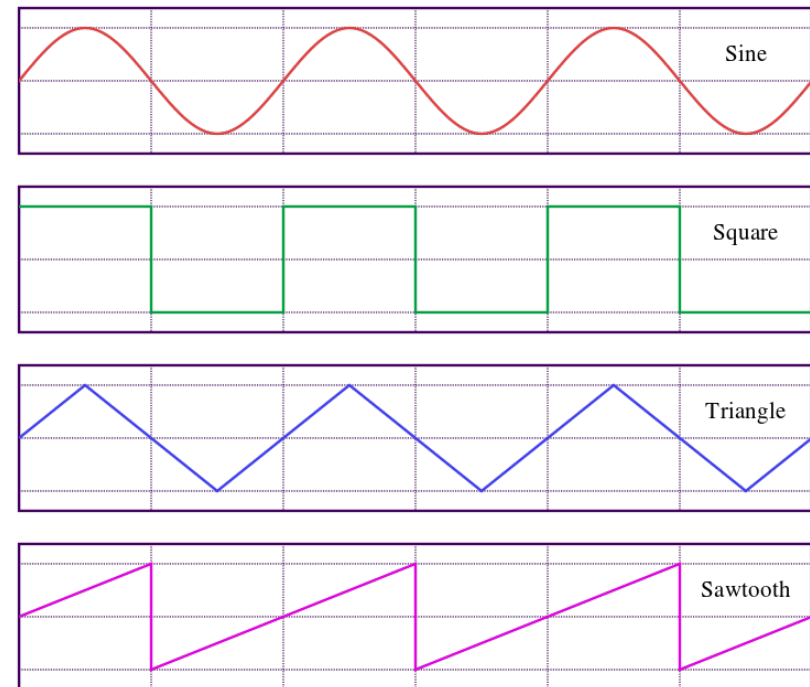
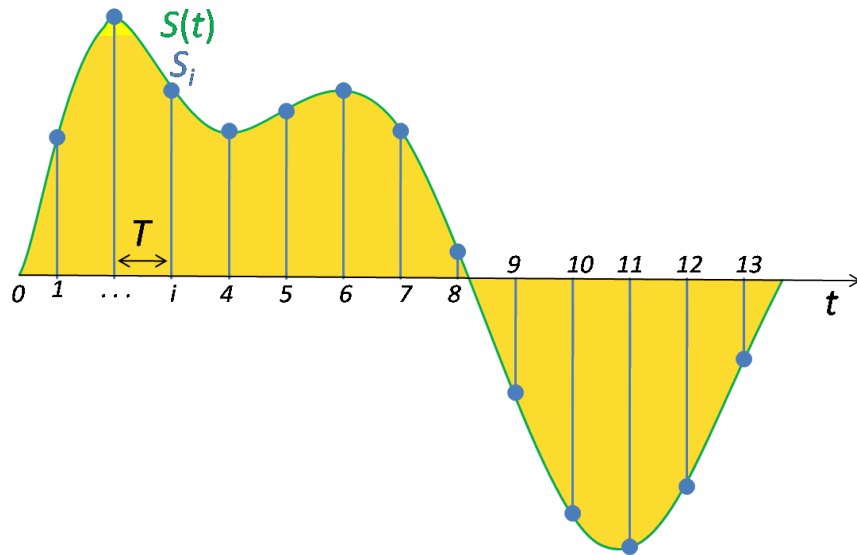The Waveform Audio File Format
encodes a sampled sound wave

A triangle wave is the simple wave
form with the most pleasing sound

# WAV Files

The Waveform Audio File Format
encodes a sampled sound wave



A triangle wave is the simple wave
form with the most pleasing sound

https://en.wikipedia.org/wiki/Triangle_wave
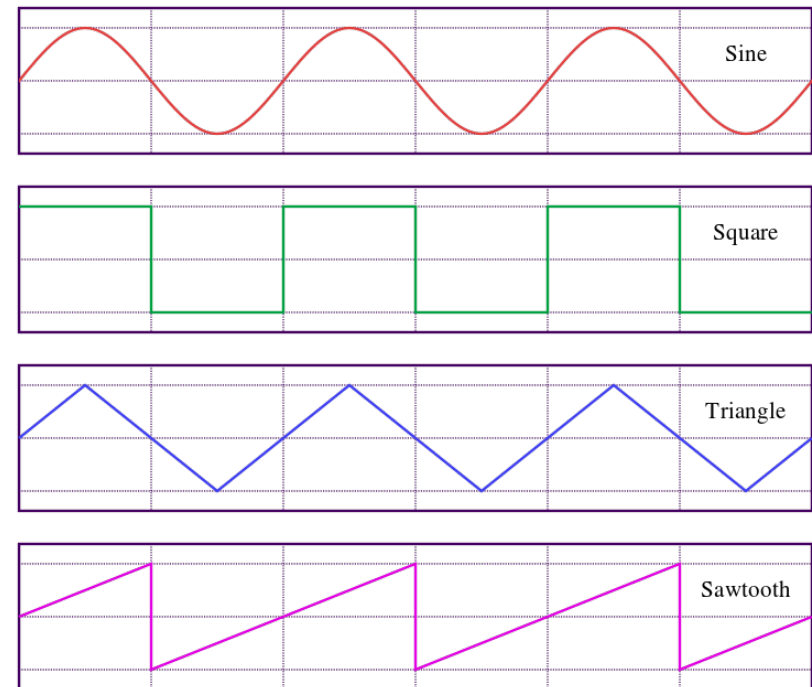https://en.wikipedia.org/wiki/Sampling_(signal_processing)

# WAV Files

The Waveform Audio File Format
encodes a sampled sound wave

A triangle wave is the simple wave
form with the most pleasing sound



(Demo)

https://en.wikipedia.org/wiki/Triangle_wave
https://en.wikipedia.org/wiki/Sampling_(signal_processing)