DFT: $X_k = \sum_{n=0}^{N-1} x(n)e^{-j2\pi nk/N}; x(n) = \frac{1}{N}\sum_{k=0}^{N-1} X_k e^{j2\pi nk/N}$; length$\leq N$.
DTFT: $X_k = X(z)|_{z=e^{j2\pi k/N}} = X(e^{j\omega})|_{\omega=2\pi k/N}$ sampled on unit circle.
DTFS: $\mathcal{F}\{$periodic extension of $\{x(n)\}\}$ =periodic extension of $\{X_k\}$.
except: factor of $1/N$ moved. **Note:** $k = 0, 1 \ldots N-1$ and $n = 0, 1 \ldots N-1$.
Text: Uses $X(k)$ not X_k. I *hate* that–too easy to confuse with $X(z)$!

What: Use DFT to compute DTFT at $\omega = \frac{2\pi k}{N}$: equispaced samples on u.c.
Why: Use for spectral analysis, and to recover $x(n)$ from DTFT. BUT:
Finite length: $x(n)$ has finite length $L \rightarrow$using $N \geq L \rightarrow$recover $x(n)$ from X_k.
But $N < L \rightarrow$can only recover $\sum_k x(n+kN) \rightarrow$ *aliased* $x(n)$.

EX: $x(n) = \{1,2,3,4,5\} \rightarrow X(e^{j\omega}) = 1 + 2e^{-j\omega} + 3e^{-2j\omega} + 4e^{-j3\omega} + 5e^{-j4\omega}$
Sample DTFT on unit circle at $\omega = 0, \frac{\pi}{2}, \pi, \frac{3\pi}{2}$ to get DFT:
DFT: $X_0 = 1+2+3+4+5 = 15; X_1 = 1-2j-3+4j+5 = 3+2j;$
N=4: $X_2 = 1-2+3-4+5 = 03; X_3 = 1+2j-3-4j+5 = 3-2j = X_1^*$.
IDFT: $x(0) = \frac{1}{4}[15+(3+2j)+3+(3-2j)] = 6$ incorrect (6=1+5 aliased).
$x(1) = \frac{1}{4}[15(1)+(3+2j)(j)+3(-1)+(3-2j)(-j)] = 2$ correct.
$x(2) = \frac{1}{4}[15(1)+(3+2j)(-1)+3(1)+(3-2j)(-1)] = 3$ correct.
$x(3) = \frac{1}{4}[15(1)+(3+2j)(-j)+3(-1)+(3-2j)(j)] = 4$ correct.
Why? *Undersampled* $X(e^{j\omega})$ on unit circle$\rightarrow$aliasing (just like before).

Inter-polate DTFT: Let $x(n)$ have length $L \leq N$. We're given $X(e^{j2\pi k/N}), k = 0, 1 \ldots N-1$.
Then interpolate $X(e^{j\omega}) = \sum_{k=0}^{N-1} X(e^{j2\pi k/N}) S(\omega - \frac{2\pi k}{N})$
$S(\omega) = \frac{\sin(\omega N/2)}{N\sin(\omega/2)} e^{-j\omega(N-1)/2} = DTFT\{\frac{1}{N}\sum_{i=0}^{N-1}\delta(n-i)\}$.

Zero padding: $N > L \rightarrow DFT\{x(0)\ldots x(L-1), 0\ldots 0(zero-pad)\} \rightarrow$above.
This smoothes DTFT (finer sampling in ω), BUT: no additional info.
EX: $DFT_N\{\sum_{i=0}^{L-1}\delta(n-i)\} = \frac{\sin(\pi kL/N)}{\sin(\pi k/N)} e^{-j\pi k(L-1)/N}$. Window blurs.

Cyclic convol: $y(n) = h(n)Ⓒx(n) = \sum_{i=0}^{N-1} h(i)x((n-i))_N \Leftrightarrow Y_k = H_k X_k$.
$(x(n))_N \Leftrightarrow$N-point periodic extension of $x(n)$. "Cyclic"="circular."
Ex: $\{1,2,3\}Ⓒ\{4,5,6\}$: Take one complete cycle of each:
$y(0) = \begin{smallmatrix}1,2,3,1,2,3\\4,6,5,4,6,5\end{smallmatrix} = 1\cdot 4 + 2\cdot 6 + 3\cdot 5 = 31$.
$y(1) = \begin{smallmatrix}1,2,3,1,2,3\\5,4,6,5,4,6\end{smallmatrix} = 1\cdot 5 + 2\cdot 4 + 3\cdot 6 = 31$.
$y(2) = \begin{smallmatrix}1,2,3,1,2,3\\6,5,4,6,5,4\end{smallmatrix} = 1\cdot 6 + 2\cdot 5 + 3\cdot 4 = 28$.
Check: $H_0 X_0 = (1+2+3)(4+5+6) = 90 = (31+31+28) = Y_0$ checks.

Goal: Compute DFT $X_k = \sum_{n=0}^{N-1} x(n) W_N^{nk}$ where $W_N = e^{-j2\pi/N}$.
Names: **FFT** is an *algorithm* for computing the **DFT**, which is a *transform*.
Why? Direct computation requires N^2 mults and $N(N-1)$ adds: too many!

Cooley-Tukey: 1965 at IBM. Serious DSP dates from this algorithm.
Divide up large DFT into smaller DFTs: $N = N_1 N_2$.

coarse: $n = n_1 + N_1 n_2$ where $n_1 = 0 \ldots N_1 - 1$ and $n_2 = 0 \ldots N_2 - 1$.
vernier: $k = k_2 + N_2 k_1$ where $k_1 = 0 \ldots N_1 - 1$ and $k_2 = 0 \ldots N_2 - 1$.
indices $nk = (n_1 + N_1 n_2)(k_2 + N_2 k_1) = n_1 k_2 + N_1 n_2 k_2 + N_2 n_1 k_1 + N_1 N_2 n_2 k_1$.
exponent: $W_N^{nk} = W_N^{n_1 k_2} W_{N_2}^{n_2 k_2} W_{N_1}^{n_1 k_1}$ using $W_{N_1 N_2}^{N_1} = W_{N_2}$ and $W_N^{N_1 N_2} = 1$.

DFT: $X_k = X_{k_2 + N_2 k_1} = \sum_{n_1=0}^{N_1-1} \sum_{n_2=0}^{N_2-1} x(n_1 + N_1 n_2) W_N^{(n_1 + N_1 n_2)(k_2 + N_2 k_1)}$.
rewrite: $X_{k_2 + N_2 k_1} = \sum_{n_1=0}^{N_1-1} W_{N_1}^{n_1 k_1} \left[W_N^{n_1 k_2} \sum_{n_2=0}^{N_2-1} W_{N_2}^{n_2 k_2} x(n_1 + N_1 n_2) \right]$.

1. Compute N_1 (for each n_1) N_2-point DFTs of $x(n_1 + N_1 n_2)$ (in n_2).
2. Multiply result by *twiddle factors* $W_N^{n_1 k_2}$. These are *twiddle mults.*
3. Compute N_2 (for each k_2) N_1-point DFTs of the result. Now done!
4. $(N_1 N_2)$-point$\rightarrow$ $N_1(N_2$-point$)+N_2(N_1$-point$)+(N_1-1)(N_2-1)$ twiddle since $n_1 = 0$ or $k_2 = 0 \rightarrow W_N^{n_1 k_2} = 1$. This is important below.

Visual: $(N_1 \times N_2)$ arrays: $x_{n_1,n_2} = x(n_1 + N_1 n_2); X_{k_1,k_2} = X(k_2 + N_2 k_1)$.

1. Take N_2-point DFT of each row (fixed n_1). Yields $\hat{x}_{n_1,k_2}$.
2. Multiply $\hat{x}_{n_1,k_2}$ point-by-point by twiddle factor $W_N^{n_1 k_2}$.
3. Take N_1-point DFT of each column (fixed k_2). Yields X_{k_1,k_2}.

Radix-2 Cooley-Tukey Fast Fourier Transforms

$N = 2\frac{N}{2} \rightarrow$ *Decimation-in-time* FFT: $N_1 = 2; N_2 = N/2; \quad n_1 = 0,1; k_1 = 0,1$.

$$X_{k_2} = 1 \sum_{n_2=0}^{N/2-1} W_{N/2}^{n_2 k_2} x(2n_2) + W_N^{1k_2} \sum_{n_2=0}^{N/2-1} W_{N/2}^{n_2 k_2} x(2n_2 + 1).$$

$$X_{k_2+N/2} = 1 \sum_{n_2=0}^{N/2-1} W_{N/2}^{n_2 k_2} x(2n_2) - W_N^{1k_2} \sum_{n_2=0}^{N/2-1} W_{N/2}^{n_2 k_2} x(2n_2 + 1).$$

N-point$\rightarrow$ $2(\frac{N}{2}$-point$)+\frac{N}{2}(2$-point$)+(\frac{N}{2}-1)$twiddle mults. Total: $\frac{N}{2}\log_2 N$ mults.
Note: $(N_1 - 1)(N_2 - 1) = (2-1)(\frac{N}{2} - 1) = \frac{N}{2} - 1$: $\frac{1}{2}$twiddle mults trivial!

$N = \frac{N}{2}2 \rightarrow$ *Decimation-in-freq.* FFT: $N_2 = 2; N_1 = N/2; \quad n_2 = 0,1; k_2 = 0,1$.

$$X_{2k_1} = \sum_{n_1=0}^{N/2-1} W_{N/2}^{n_1 k_1} [x(n_1) + x(n_1 + N/2)]1.$$

$$X_{2k_1+1} = \sum_{n_1=0}^{N/2-1} W_{N/2}^{n_1 k_1} [x(n_1) - x(n_1 + N/2)] W_N^{1n_1}.$$

N-point$\rightarrow$ $\frac{N}{2}(2$-point$)+2(\frac{N}{2}$-point$)+(\frac{N}{2}-1)$twiddle mults. Total: $\frac{N}{2}\log_2 N$ mults.