Integer multiplication and the Fast Fourier Transform

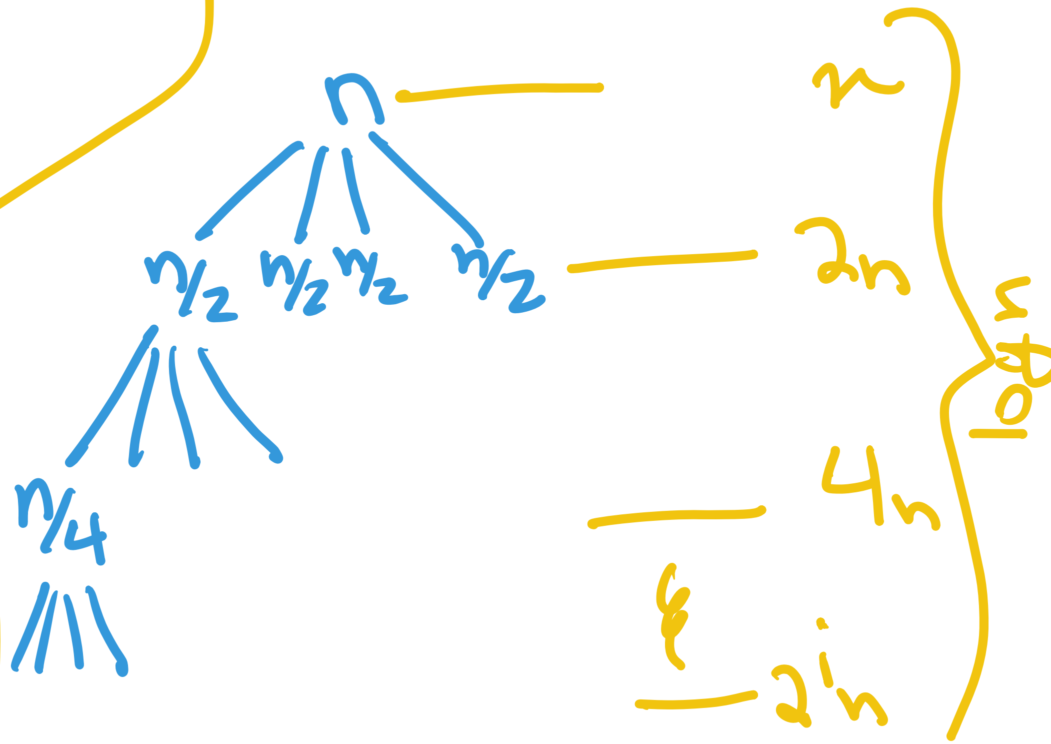
Multiplying n-bit integers

Karatsuba

Input: $x, y \in \{0,1\}^n$

Output: product $(xy) \in \{0,1\}^{2n}$

$$
\begin{array}{r}
1234 \\
\times\ 5678 \\
\hline
9672 \\
8638 \\
\vdots \\
+ \\
\hline
\end{array}
$$



$$\sum_{i=1}^{\log n} 2^i n = O(2^{\log_2 n} n) = O(n^2)$$

Better?

Divide bits in half

$$x \quad \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{x_1 \in \{0,1\}^{n/2}}\;\underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{x_2 \in \{0,1\}^{n/2}}$$

$$y \quad \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{y_1 \in \{0,1\}^{n/2}}\;\underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{y_2 \in \{0,1\}^{n/2}}$$

$$x = x_1 2^{n/2} + x_2, \quad y = y_1 2^{n/2} + y_2$$

$$(x_1 2^{n/2} + x_2)(y_1 2^{n/2} + y_2)$$

$$x \cdot y = \boxed{x_1 y_1}\, 2^n + \underbrace{(x_1 y_2 + y_1 x_2)}\, 2^{n/2} + \boxed{x_2 y_2}$$

which multiplies 4 pairs of $(n/2)$-bit numbers

$$\underbrace{\boxed{(x_1+x_2)(y_1+y_2)}}_{(n/2+1)\text{ bits}} = x_1y_1 + x_1y_2 + x_2y_1 + x_2y_2$$

$$\boxed{x \cdot y = x_1 y_1 2^n + (x_1 y_2 + y_1 x_2) 2^{n/2} + x_2 y_2}$$

Karatsuba's Trick

$$x \cdot y = x_1y_1 2^n + (x_1y_2 + y_1x_2)2^{n/2} + x_2y_2$$

Karatsuba's Trick

already computed

☆ $$\underbrace{(x_1+x_2)(y_1+y_2)}_{\text{one multiply}} = \underbrace{x_1y_1}_{+} + \underbrace{\underbrace{x_1y_2 + x_2y_1}_{\text{middle term}}}_{-} + \underbrace{x_2y_2}_{+}$$

$$x_1y_2 + x_2y_1 = x_1y_1 + x_2y_2 - (x_1+x_2)(y_1+y_2)$$

2 multiplies for the price of 1 (more)!

☆ $$(x_1-x_2)(y_1-y_2) = x_1y_1 + x_2y_2 - x_1y_2 - x_2y_1$$

① Recursive specification

multiply($x[1..,n]$, $y[1...n]$):

given two integers x, y (as bit strings), return their product (as bit strings)

Designing a recursive algorithm

① **Recursive specification**
- declares input/output of algo
- "contract" algo must fulfill
- induction hypothesis for correctness

② **Implement spec**

③ **Prove correctness**

argue algo satisfies recursive spec for all n by induction on n

---

Divide bits in half

$$x = \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{x_1 \in \{0,1\}^{n/2}}\ \underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{x_2 \in \{0,1\}^{n/2}}$$

$$y = \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{y_1 \in \{0,1\}^{n/2}}\ \underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{y_2 \in \{0,1\}^{n/2}}$$

$$x = x_1 2^{n/2} + x_2, \quad y = y_1 2^{n/2} + y_2$$

$$x \cdot y = x_1 y_1 2^n + (x_1 y_2 + y_1 x_2) 2^{n/2} + x_2 y_2$$

which multiplies 4 pairs of $(n/2)$-bit numbers

Karatsuba's Trick

already computed

$$\underbrace{(x_1+x_2)(y_1+y_2)}_{\text{one multiply}} = x_1y_1 + \underbrace{x_1y_2 + x_2y_1}_{\text{middle term}} + x_2y_2$$

$$x_1y_2 + x_2y_1 = x_1y_1 + x_2y_2 - (x_1+x_2)(y_1+y_2)$$

2 multiplies for the price of 1 (more)!

① Recursive specification

Designing a recursive algorithm

① Recursive specification
- declares input/output of algo
- "contract" algo must fulfill
- induction hypothesis for correctness

② Implement spec

③ Prove correctness
argue algo satisfies recursive spec
for all n by induction on n

multiply$(x[1..n], y[1..n])$ = the ($2n$-bit integer) product of x and y (as n-bit integers).

Divide bits in half

$$x \quad \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{x_1 \in \{0,1\}^{n/2}}\;\underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{x_2 \in \{0,1\}^{n/2}}$$

$$y \quad \underbrace{1\,0\,1\,0\,1\,1\,0\,0}_{y_1 \in \{0,1\}^{n/2}}\;\underbrace{1\,0\,1\,0\,1\,1\,0\,1}_{y_2 \in \{0,1\}^{n/2}}$$

$$x = x_1 2^{n/2} + x_2, \quad y = y_1 2^{n/2} + y_2$$

$$x \cdot y = x_1 y_1 2^n + (x_1 y_2 + y_1 x_2) 2^{n/2} + x_2 y_2$$

which multiplies 4 pairs of $(n/2)$-bit numbers

Karatsuba's Trick

already computed

$$\underbrace{(x_1+x_2)(y_1+y_2)}_{\text{one multiply}} = x_1 y_1 + \underbrace{x_1 y_2 + x_2 y_1}_{\text{middle term}} + x_2 y_2$$

$$x_1 y_2 + x_2 y_1 = x_1 y_1 + x_2 y_2 - (x_1+x_2)(y_1+y_2)$$

2 multiplies for the price of 1 (more) !

① Recursive specification

`select`$(k, A[1..n])$: Given an array $A[1..n]$ of comparable elements and an integer $k \in \{1, \ldots, n\}$, returns the kth smallest element out of $A[1..n]$.

② Implement spec

`multiply`$(x[1..n], y[1..n])$

1. If $n = 0$ then return 0.

2. If $n = 1$ then return $x[1] * y[1]$.

3. If n is not even then pad x and y with a leading 0, making n even. // $O(n)$.

4. Let $x_1 = x[1..n/2]$, $x_2 = [n/2+1..n]$, $y_1 = y[1..n/2]$, and $y_2 = y[n/2+1..n]$ (as $n/2$-bit integers). // $O(n)$.

5. Let $x_3[1..n/2+1] = x_1 + x_2$ and $y_3[1..n/2+1] = (y_1 + y_2)$ (as $(n/2+1)$-bit integers). // $O(n)$.

6. Let $Z_1 =$ `multiply`(x_1, y_1), $Z_2 =$ `multiply`(x_2, y_2), and $Z_3 =$ `multiply`(x_3, y_3). // $3T((n+2)/2)$.

/* *Note that multiplying by 2^n is just a bit-shift and takes $O(n)$ time.* */

7. Return $2^n * Z_1 + 2^{n/2}(Z_3 - Z_1 - Z_2) + Z_2$

Designing a recursive algorithm

① Recursive specification
- declares input/output of algo
- "contract" algo must fulfill
- induction hypothesis for correctness

② Implement spec

③ Prove correctness

argue algo satisfies recursive spec for all n by induction on n

4. Running time

$$T(n) = 3T(n/2) + O(1)$$

n

$\frac{3}{2} n$

$\left(\frac{3}{2}\right)^2 n$

$\vdots$

$\left(\frac{3}{2}\right)^i n$

$$\log_2(3/2) = \log_2(3) - 1$$

$$\sum_{i=1}^{\log_2 n} \left(\frac{3}{2}\right)^i n = O\left(\left(\frac{3}{2}\right)^{\log_2 n} n\right)$$

$$\left(\frac{3}{2}\right)^{\log_2 n} = \left(2^{\log_2(3/2)}\right)^{\log_2 n} = \left(2^{\log_2 n}\right)^{\log_2(3/2)} = n^{\log_2(3/2)}$$

multiply($x[1..n], y[1..n]$)

1. If $n = 0$ then return 0.
2. If $n = 1$ then return $x[1] * y[1]$.
3. If n is not even then pad x and y with a leading 0, making n even. // $O(n)$.
4. Let $x_1 = x[1..n/2]$, $x_2 = [n/2+1..n]$, $y_1 = y[1..n/2]$, and $y_2 = y[n/2+1..n]$ (as $n/2$-bit integers). // $O(n)$.
5. Let $x_3[1..n/2+1] = x_1 + x_2$ and $y_3[1..n/2+1] = (y_1 + y_2)$ (as $(n/2+1)$-bit integers). // $O(n)$.
6. Let $Z_1 =$ multiply(x_1, y_1), $Z_2 =$ multiply(x_2, y_2), and $Z_3 =$ multiply(x_3, y_3). // $3T((n+2)/2)$.

/* *Note that multiplying by 2^n is just a bit-shift and takes $O(n)$ time.* */

7. Return $2^n * Z_1 + 2^{n/2}(Z_3 - Z_1 - Z_2) + Z_2$

4. Running time

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n)$$

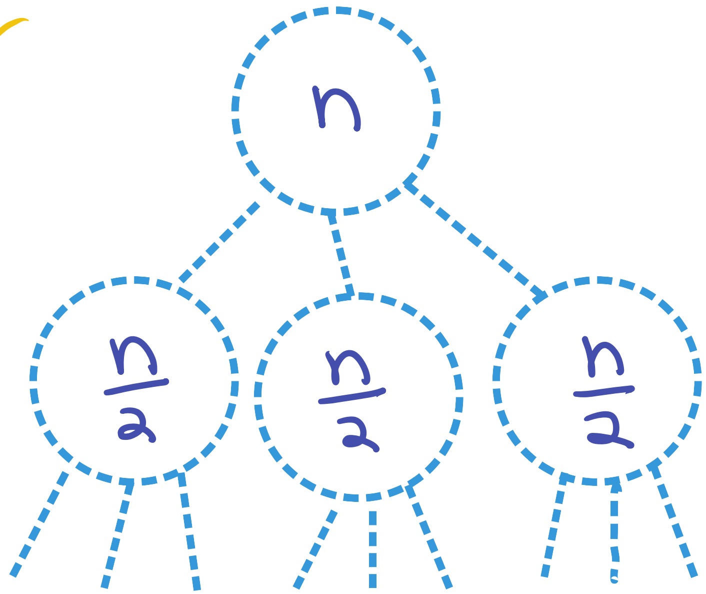
$n^{\log_2 3}$

<u>multiply($x[1..n]$, $y[1..n]$)</u>

1. If $n = 0$ then return 0.
2. If $n = 1$ then return $x[1] * y[1]$.
3. If n is not even then pad x and y with a leading 0, making n even. *// $O(n)$.*
4. Let $x_1 = x[1..n/2]$, $x_2 = [n/2+1..n]$, $y_1 = y[1..n/2]$, and $y_2 = y[n/2+1..n]$ (as $n/2$-bit integers). *// $O(n)$.*
5. Let $x_3[1..n/2+1] = x_1 + x_2$ and $y_3[1..n/2+1] = (y_1 + y_2)$ (as $(n/2+1)$-bit integers). *// $O(n)$.*
6. Let $Z_1 =$ multiply(x_1, y_1), $Z_2 =$ multiply(x_2, y_2), and $Z_3 =$ multiply(x_3, y_3). *// $3T((n+2)/2)$.*

/ Note that multiplying by 2^n is just a bit-shift and takes $O(n)$ time. */*

7. Return $2^n * Z_1 + 2^{n/2}(Z_3 - Z_1 - Z_2) + Z_2$



$h = \log_2 n$

n

$3 \times \frac{n}{2} = \frac{3}{2} n$

$\left(\frac{3}{2}\right)^2 n$

$\vdots$

$\left(\frac{3}{2}\right)^h n$

Running time

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n)$$

$$\sum_{i=0}^{\log_2 n} \left(\frac{3}{2}\right)^i n =$$

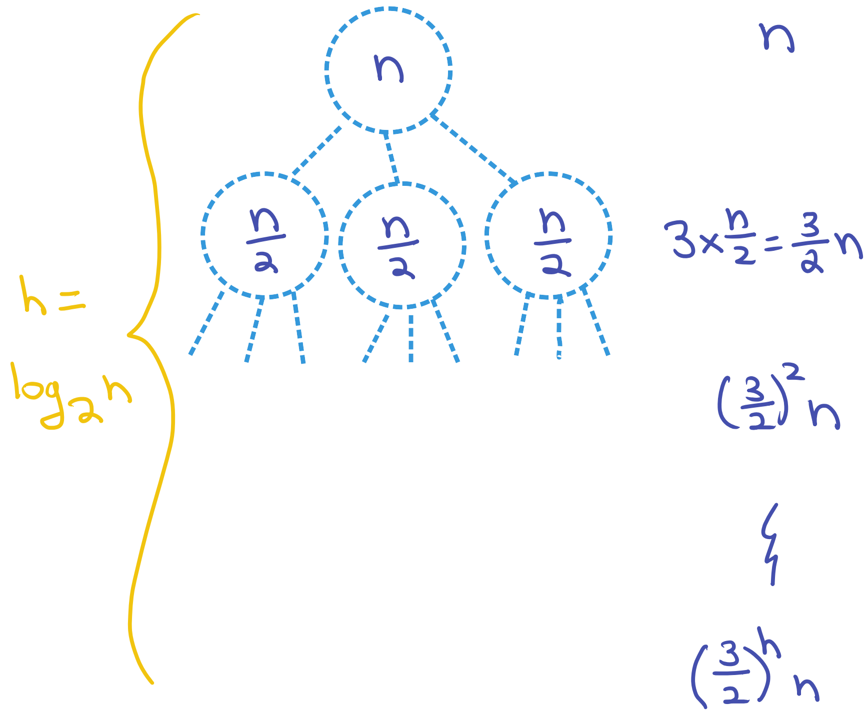
4. Running time

multiply($x[1..n]$, $y[1..n]$)

1. If $n = 0$ then return 0.
2. If $n = 1$ then return $x[1] * y[1]$.
3. If n is not even then pad x and y with a leading 0, making n even. // $O(n)$.
4. Let $x_1 = x[1..n/2]$, $x_2 = [n/2+1..n]$, $y_1 = y[1..n/2]$, and $y_2 = y[n/2+1..n]$ (as $n/2$-bit integers). // $O(n)$.
5. Let $x_3[1..n/2+1] = x_1 + x_2$ and $y_3[1..n/2+1] = (y_1 + y_2)$ (as $(n/2+1)$-bit integers). // $O(n)$.
6. Let Z_1 = multiply(x_1, y_1), Z_2 = multiply(x_2, y_2), and Z_3 = multiply(x_3, y_3). // $3T((n+2)/2)$.

/* *Note that multiplying by 2^n is just a bit-shift and takes $O(n)$ time.* */

7. Return $2^n * Z_1 + 2^{n/2}(Z_3 - Z_1 - Z_2) + Z_2$



Running time

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n)$$

$$\sum_{i=0}^{\log_2 n} \left(\frac{3}{2}\right)^i n = O\left(\left(\frac{3}{2}\right)^{\log_2 n} n\right)$$

$$= O\left(n^{\underline{\log_2(3/2)} + 1}\right)$$

$$= O\left(n^{\log_2(3)}\right)$$

$$T(n) = 3T\left(\frac{n}{2}\right) + O(n) = O\left(n^{\log_2 3}\right)$$

$$\approx n^{1.585}$$

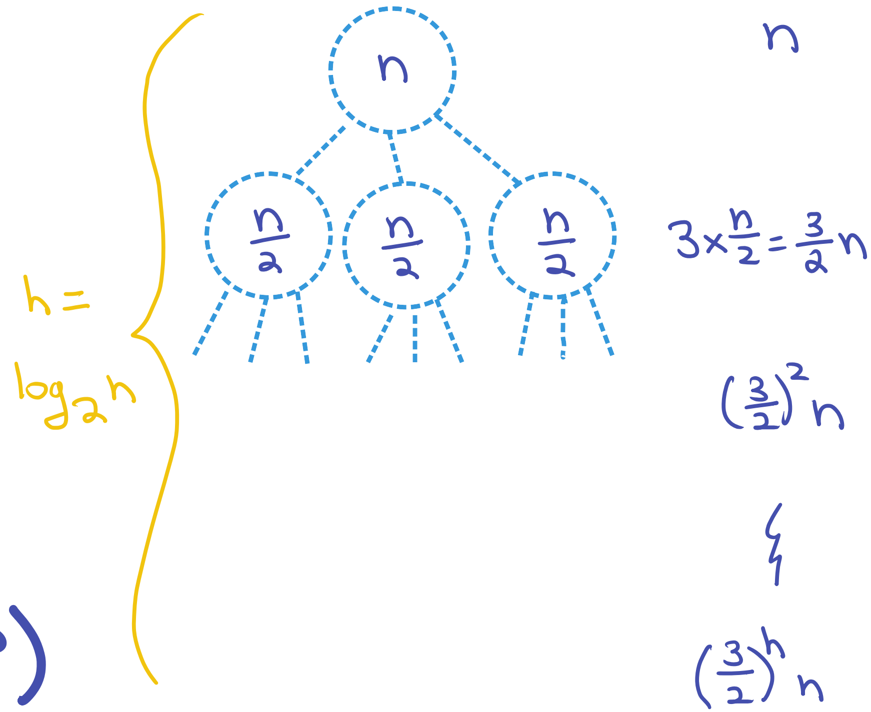
4. Running time

multiply($x[1..n]$, $y[1..n]$)

1. If $n = 0$ then return 0.
2. If $n = 1$ then return $x[1] * y[1]$.
3. If n is not even then pad x and y with a leading 0, making n even. // $O(n)$.
4. Let $x_1 = x[1..n/2]$, $x_2 = [n/2+1..n]$, $y_1 = y[1..n/2]$, and $y_2 = y[n/2+1..n]$ (as $n/2$-bit integers). // $O(n)$.
5. Let $x_3[1..n/2+1] = x_1 + x_2$ and $y_3[1..n/2+1] = (y_1 + y_2)$ (as $(n/2+1)$-bit integers). // $O(n)$.
6. Let $Z_1 =$ multiply(x_1, y_1), $Z_2 =$ multiply(x_2, y_2), and $Z_3 =$ multiply(x_3, y_3). // $3T((n+2)/2)$.

/* *Note that multiplying by 2^n is just a bit-shift and takes $O(n)$ time.* */

7. Return $2^n * Z_1 + 2^{n/2}(Z_3 - Z_1 - Z_2) + Z_2$



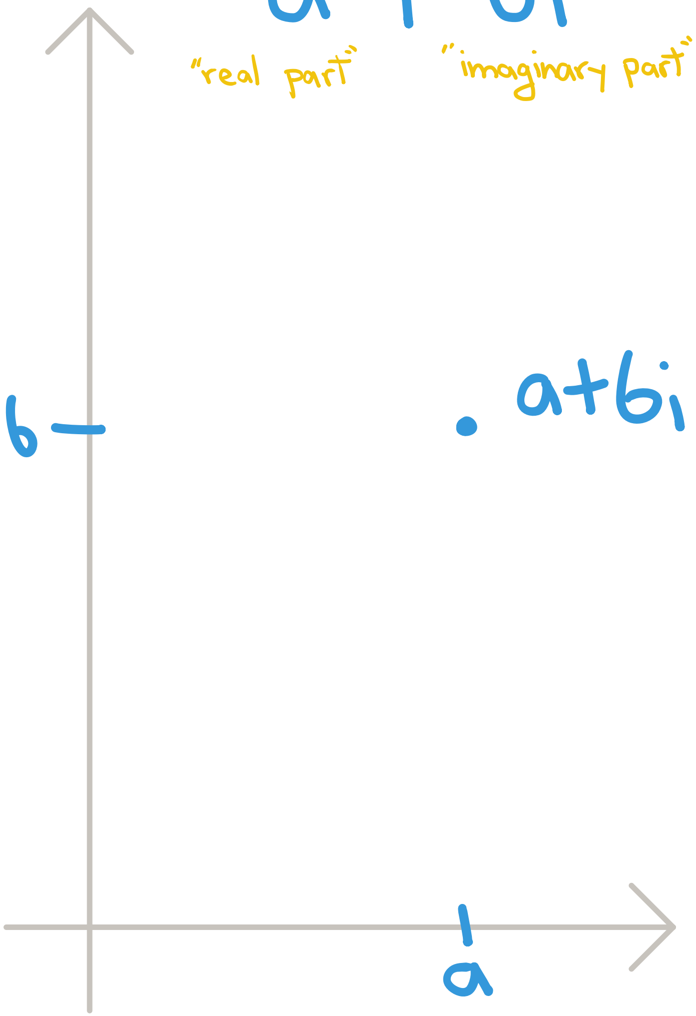
Integer multiplication and the Fast Fourier Transform

Complex numbers $\mathbb{C}$

"imaginary i" satisfying $i^2 = -1$

$a + bi$

"real part" "imaginary part"



multiplication:

$$(a+bi)(c+di) = ac + (bc+ad)i - bd$$

$\Rightarrow$ polynomials over $\mathbb{C}$

$$p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_{n-1}x^{n-1}$$

$\Rightarrow$ power series over $\mathbb{C}$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

Euler's Formula

$$e^{\theta i} = \cos\theta + i\sin\theta$$

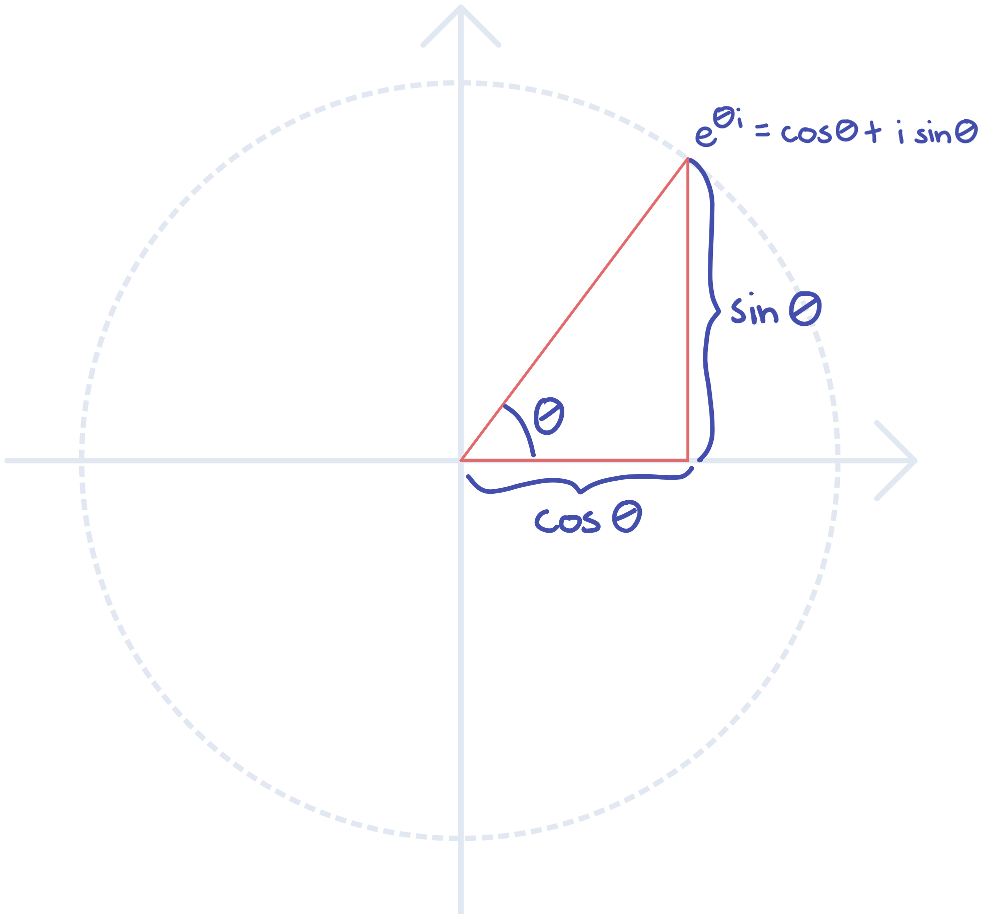
$a + bi$

"real part" "imaginary part"

"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$



$\Rightarrow$ Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

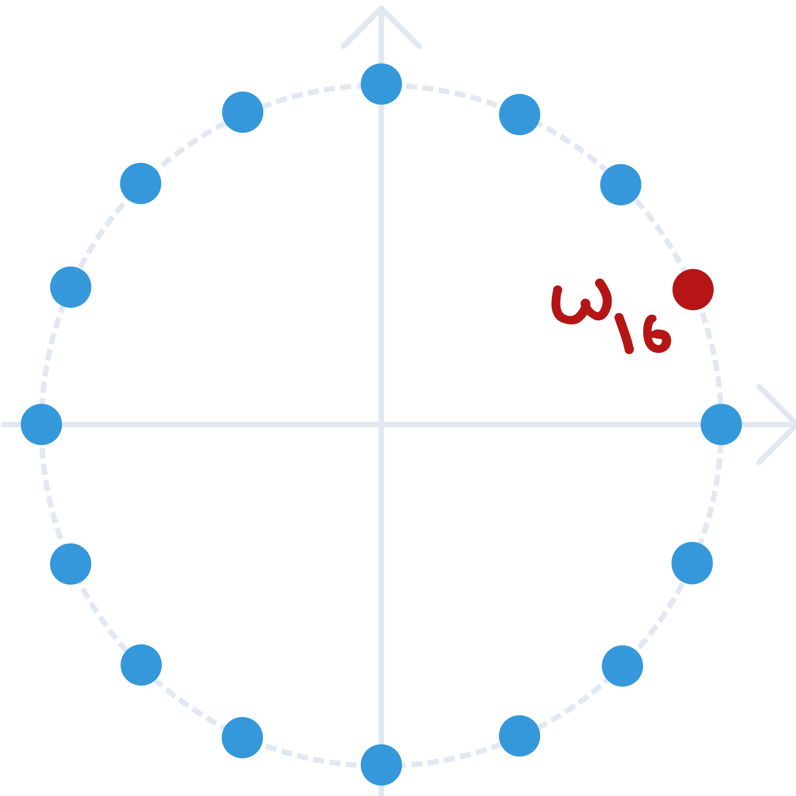
nth roots of unity:

any $\omega \in \mathbb{C}$ s.t. $\omega^n = 1$

e.g. $\omega = 1$ for all n

$\omega = -1$ for n even

$\omega = i$ for $n = 4$



let

$$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$

are all nth roots of unity

$a + bi$

"real part" "imaginary part"

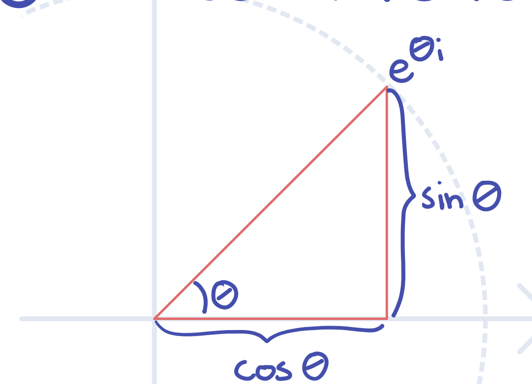
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

Euler's formula:

$$e^{\theta i} = \cos\theta + i\sin\theta$$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

(nth) discrete Fourier transform F_n

Input: $a_0, a_1, a_2, \ldots, a_{n-1} \in \mathbb{C}$

Output: n values

$$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \ldots, p(\omega_n^{n-1}))$$

where $p(x) = a_0 + a_1 x + a_1 x^2 + \cdots + a_{n-1} x^{n-1}$

conjugate Fourier transform $F_n^{\star}$

Input: $a_0, a_1, a_2, \ldots, a_{n-1} \in \mathbb{C}$

Output: n values

$$F_n^{\star} a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \ldots, p(\omega_n))$$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

$a + bi$

"real part" "imaginary part"

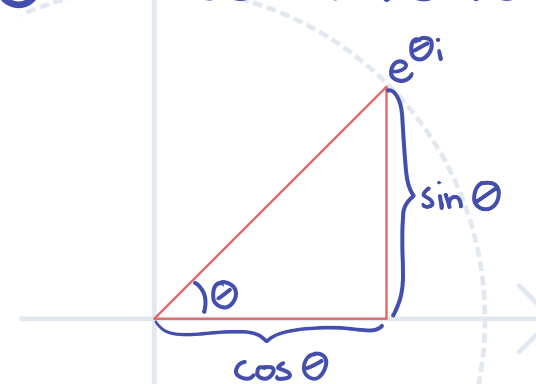
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$$e^{\theta i} = \cos\theta + i \sin\theta$$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \stackrel{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \ldots, \omega_n^{n-1}$ are all nth roots of unity

Input: $a_0, a_1, a_2, \cdots, a_{n-1} \in \mathbb{C}$

Output: n values

$$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$$

$$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

(i.e., F_n and F_n^* are essentially inverse)

$$a + bi$$

"real part" "imaginary part"

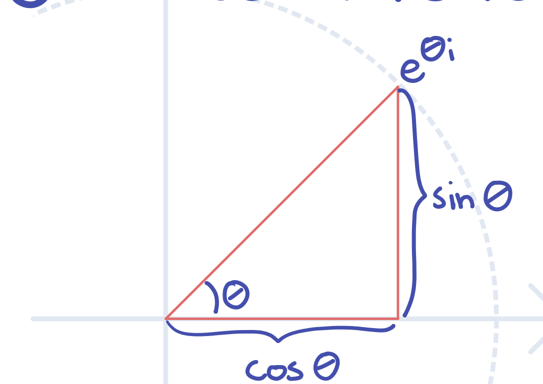
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$$

Euler's formula:

$$e^{\theta i} = \cos\theta + i \sin\theta$$



Euler's identity:

- $e^{\pi i} = -1$ • $e^{2\pi i} = 1$

$$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

Input: $a_0, a_1, a_2, \cdots, a_{n-1} \in \mathbb{C}$

Output: n values

$$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$$

$$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a$ and $F_n^* a$ can be computed in $O(n \log n)$ time.

$a + bi$

"real part" "imaginary part"

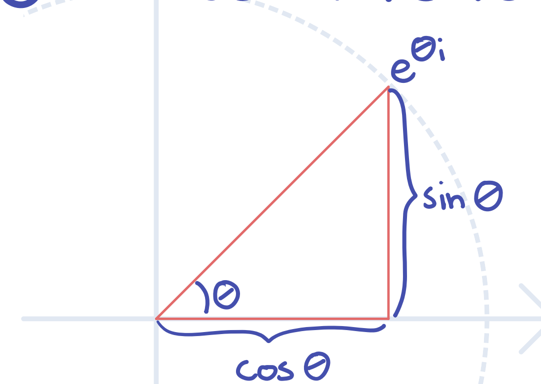
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

Application: multiplying polynomials

$$p(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1}$$

$$q(x) = b_0 + b_1 x + \cdots + b_{n-1} x^{n-1}$$

goal: compute

$$r(x) = p(x)\,q(x)$$

$$= c_0 + c_1 x + \cdots + c_{2n-2} x^{n-2}$$

where $c_k = \sum_{i=0}^{k} a_i b_{k-i}$

$a + bi$

"real part" "imaginary part"

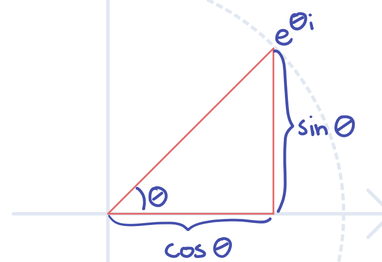
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$ • $e^{2\pi i} = 1$

$$\omega_n \stackrel{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^\star a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^\star F_n a = F_n F_n^\star a = na$

Theorem: $F_n a$, $F_n^\star a$ in $O(n \log n)$

Input: $p(x) = a_0 + a_1 x + \cdots + a_{n-1} x^{n-1}$

$q(x) = b_0 + b_1 x + \cdots + b_{n-1} x^{n-1}$

goal: $r(x) = p(x)\,q(x) = c_0 + c_1 x + \cdots + c_{2n-2} x^{n-2}$

Idea: $r(\omega_{2n}^i) = p(\omega_{2n}^i)\,q(\omega_{2n}^i)$

$$(p, q)$$

$\downarrow$ $O(n \log n)$

$$(F_{2n} a, F_{2n} b)$$

$\downarrow$ $O(n)$ by mult. pairwise

$$F_{2n} c$$

$\downarrow$ $O(n \log n)$

$$q = \frac{1}{2n} F_{2n}^{\star} F_{2n} c$$

$$a + bi$$

"real part" "imaginary part"

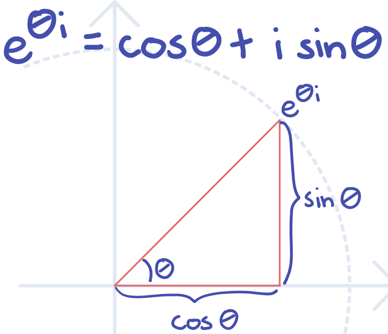
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^{\star} a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^{\star} F_n a = F_n F_n^{\star} a = na$

Theorem: $F_n a$, $F_n^{\star} a$ in $O(n \log n)$

Input: $a_0, a_1, a_2, \cdots, a_{n-1} \in \mathbb{C}$

Output: n values

$$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$$

$$F_n^{*} a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^{*} F_n a = F_n F_n^{*} a = na$

Theorem: $F_n a$ and $F_n^{*} a$ can be computed in $O(n \log n)$ time.

$a + bi$

"real part" "imaginary part"

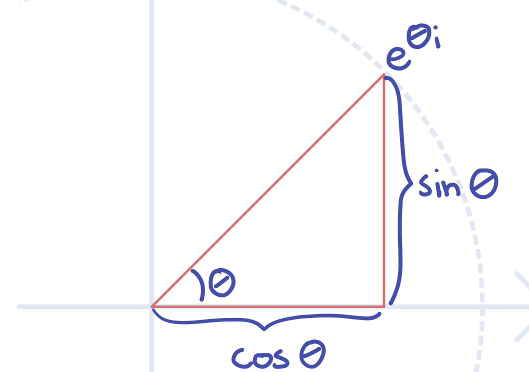
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \stackrel{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \cdots, a_{2n-1})$.

$$(F_{2n}a)_k = a_0 + a_1\omega_{2n}^k + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}$$

$$= \left(a_0 + a_2(\omega_{2n}^k)^2 + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right) \quad \text{(even)}$$

$$+ \left(a_1\omega_{2n}^k + a_3(\omega_{2n}^k)^2 + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}\right) \quad \text{(odd)}$$

$a + bi$

"real part" "imaginary part"

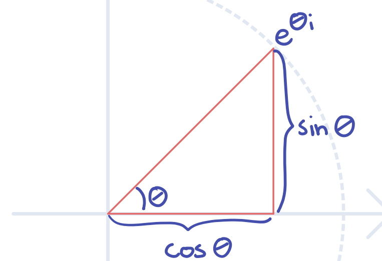
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i\sin\theta$



Euler's identity:

- $e^{\pi i} = -1$ • $e^{2\pi i} = 1$

$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a$, $F_n^* a$ in $O(n \log n)$

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \cdots, a_{2n-1})$.

$$(F_{2n}a)_k = a_0 + a_1\omega_{2n}^k + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}$$

$a + bi$

"real part" "imaginary part"

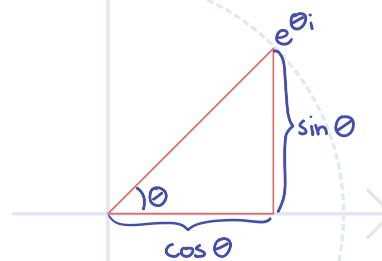
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i\sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^\star a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^\star F_n a = F_n F_n^\star a = na$

Theorem: $F_n a$, $F_n^\star a$ in $O(n\log n)$

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \cdots, a_{2n-1})$.

$$(F_{2n}a)_k = a_0 + a_1\omega_{2n}^k + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}$$

$$= \left(a_0 + a_2(\omega_{2n}^k)^2 + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right) \quad \text{(even)}$$

$$+ \left(a_1\omega_{2n}^k + a_3(\omega_{2n}^k)^2 + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}\right) \quad \text{(odd)}$$

$$\left(a_0 + a_2(\omega_{2n}^k)^2 + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right)$$

$=$

$a + bi$

"real part" "imaginary part"

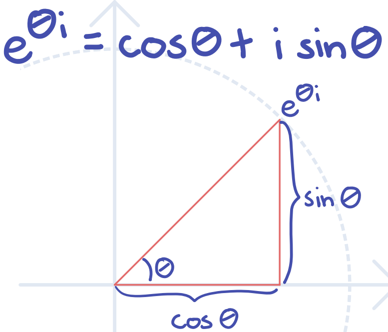
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i\sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a, F_n^* a$ in $O(n\log n)$

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \cdots, a_{2n-1})$.

$$(F_{2n}a)_k = a_0 + a_1\omega_{2n}^k + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}$$

$$= \left(a_0 + a_2(\omega_{2n}^k)^2 + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right) \quad \text{(even)}$$

$$+ \left(a_1\omega_{2n}^k + a_3(\omega_{2n}^k)^2 + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}\right) \quad \text{(odd)}$$

$$\left(a_0 + a_2\underline{(\omega_{2n}^k)^2} + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right)$$

$$= a_0 + a_2\underline{\omega_n^k} + \cdots + a_{2n-2}(\omega_n^k)^{n-1}$$

$$= (F_n a_{even})_k$$

w/ $a_{even} = (a_0, a_2, \cdots, a_{2n-2})$

$a + bi$

"real part" "imaginary part"

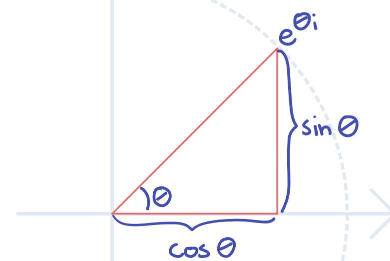
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i\sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \overset{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_nx^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a$, $F_n^* a$ in $O(n \log n)$

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \ldots, a_{2n-1})$.

$$(F_{2n}a)_k = a_0 + a_1\omega_{2n}^k + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}$$

$$= \left(a_0 + a_2(\omega_{2n}^k)^2 + \cdots + a_{2n-2}(\omega_{2n}^k)^{2n-2}\right) \quad \text{(even)}$$

$$+ \left(a_1\omega_{2n}^k + a_3(\omega_{2n}^k)^2 + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}\right) \quad \text{(odd)}$$

$$\left(a_1\omega_{2n}^k + a_3(\omega_{2n}^k)^2 + \cdots + a_{2n-1}(\omega_{2n}^k)^{2n-1}\right)$$

$$= \omega_{2n}^k\left(a_1 + a_3\omega_n^k + \cdots + a_{2n-1}(\omega_n^k)^{n-1}\right)$$

$$= \omega_{2n}^k (F_n a_{odd})_k$$

w/ $a_{odd} = (a_1, a_3, \ldots, a_{2n-1})$

$a + bi$ — "real part" (a), "imaginary part" (b)

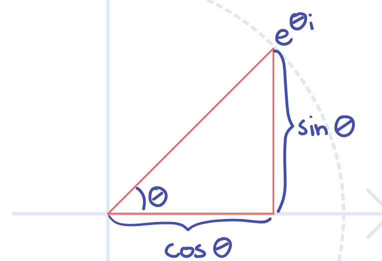
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i\sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$\omega_n \stackrel{\text{def}}{=} e^{2\pi i/n}$

$1, \omega_n, \omega_n^2, \ldots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \ldots, p(\omega_n^{n-1}))$

$F_n^{*} a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \ldots, p(\omega_n))$

where $p(x) = a_0 + a_1x + a_2x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^{*} F_n a = F_n F_n^{*} a = na$

Theorem: $F_n a$, $F_n^{*} a$ in $O(n \log n)$

Idea: $\omega_{2n}^2 = \omega_n$

let $a = (a_0, a_1, \cdots, a_{2n-1})$.

$$(F_{2n} a)_k = a_0 + a_1 \omega_{2n}^k + \cdots + a_{2n-1} (\omega_{2n}^k)^{2n-1}$$

$$= \left(a_0 + a_2 (\omega_{2n}^k)^2 + \cdots + a_{2n-2} (\omega_{2n}^k)^{2n-2}\right) \quad \text{(even)}$$

$$+ \left(a_1 \omega_{2n}^k + a_3 (\omega_{2n}^k)^2 + \cdots + a_{2n-1} (\omega_{2n}^k)^{2n-1}\right) \quad \text{(odd)}$$

$$= (F_n a_{even})_k + \omega_{2n}^k (F_n a_{odd})_k$$

recurse!

$a + bi$

"real part" "imaginary part"

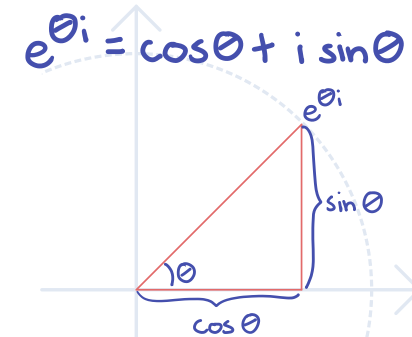
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \stackrel{\text{def}}{=} e^{2\pi i / n}$$

$1, \omega_n, \omega_n^2, \cdots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \cdots, p(\omega_n^{n-1}))$

$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \cdots, p(\omega_n))$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a$, $F_n^* a$ in $O(n \log n)$

Fast Fourier Transform$(a_0, \ldots, a_n)$:

1. if $n=0$ return $\emptyset$
2. if $n=1$ return (a_0)
3. let $(y_0, \ldots, y_{\lceil n/2 \rceil - 1}) = \mathrm{FFT}(a_{even})$
4. let $(z_0, \ldots, z_{\lfloor n/2 \rfloor - 1}) = \mathrm{FFT}(a_{odd})$
5. return $(x_0, \ldots, x_{n-1})$

where $x_k = y_k + \omega_n^k z_k \quad k=0,\ldots,n-1$

$$T(n) = 2T(n/2) + O(n)$$
$$= \Theta(n \log n)$$

$a + bi$

"real part" "imaginary part"

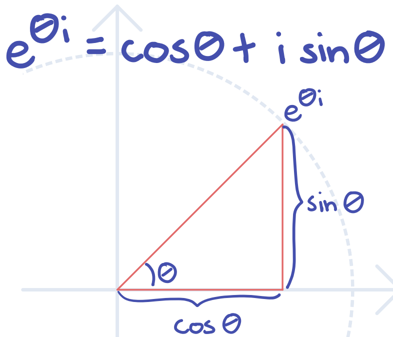
"imaginary i" s.t. $i^2 = -1$

$(a+bi)(c+di) = ac + (bc+ad)i - bd$

$e^x = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \cdots$

Euler's formula:

$e^{\theta i} = \cos\theta + i \sin\theta$



Euler's identity:

- $e^{\pi i} = -1$
- $e^{2\pi i} = 1$

$$\omega_n \stackrel{\text{def}}{=} e^{2\pi i/n}$$

$1, \omega_n, \omega_n^2, \ldots, \omega_n^{n-1}$ are all nth roots of unity

$F_n a = (p(1), p(\omega_n), p(\omega_n^2), \ldots, p(\omega_n^{n-1}))$

$F_n^* a = (p(1), p(\omega_n^{n-1}), p(\omega_n^{n-2}), \ldots, p(\omega_n))$

where $p(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_n x^{n-1}$

Fact: $F_n^* F_n a = F_n F_n^* a = na$

Theorem: $F_n a$, $F_n^* a$ in $O(n \log n)$