

HW #1

$$i = 2^k$$

$$k = \log_2(i)$$

$$i = 2^{2^k}$$

(a) There's C^k	0	1	2	3	...	$k \rightarrow \log(\log(n))$	$\log_2(i) = 2^k$
i after Her	2	4	16	256	...	$2^{2^k} \Rightarrow n$	$\log_2(\log_2(i)) = k$

$$\sum_{k=2}^{\log(\log n)} i \left(\theta(1) \right) = \sum_{k=2}^{\log(\log n)} \theta(1) = \boxed{\theta(\log_2(\log_2 n))}$$

$i \% (\text{int}) \sqrt{n} == 0$
 i must be mult. of $\sqrt{2}$..
 so runs \sqrt{n} times...

$$(b) \sum_{i=1}^n \left(\theta(1) + o\left(\sum_{k=0}^{i-1} \theta(1)\right) \right) = \sum_{i=1}^n \theta(1) + \sum_{i=1}^n \sum_{k=0}^{i-1} \theta(1)$$

$$= \theta(n) + \sum_{k=0}^{\sqrt{n}} \theta(i^3) \quad \sum_{i=0}^{\sqrt{n}} \theta(i^3) = o\left(\sum_{i=0}^{\sqrt{n}} i^3\right) = o(n^2)$$

calculator, sorry!

$$\dots = \theta(n) + \theta(n^2) = \boxed{\theta(n^2)}$$

~~scribbled out text~~

$$(c) \sum_{i=1}^n \left(\theta(1) + \sum_{k=1}^n \left(\theta(1) + o\left(\sum_{m=1}^{\log n} \theta(1)\right) \right) \right) = \sum_{i=1}^n \left(\theta(1) + \sum_{k=1}^n \theta(1) + \sum_{k=1}^n \sum_{m=1}^{\log n} \theta(1) \right)$$

$$= \sum_{i=1}^n \left(\theta(1) + \theta(n) + \sum_{k=0}^n \theta(\log n) \right) = \sum_{i=1}^n \theta(1) + \sum_{i=1}^n \theta(n) + \sum_{i=1}^n \theta(n \log n)$$

$$= \theta(n) + \theta(n^2) + \theta(n^2 \log_2 n) = \boxed{\theta(n^2 \cdot \log_2 n)}$$

$$(d) \sum_{i=0}^{n-1} \left(\theta(1) + \right) = \boxed{\theta(n)}$$

NO proper for loop inside
 the outer loop. Syntax
 is wrong PLUS no
 indent...

HW #1 : Q2

2)

notes:

- `llvec()` takes in 2 LL and merges in alternating recursive order:

Base Cases: 1) `in1 == nullptr`: return `in2`

2) `in2 == nullptr`: return `in1`

Rec. Step: \Rightarrow sets `in1 \rightarrow next` to `llvec(in2, in1 \rightarrow next)`

\hookrightarrow thus, alternates nodes between `in1` and `in2`!!

a) `in1 = 1, 2, 3, 4`

`in2 = 5, 6`

\downarrow

`llvec(1, 5)`

• `in1 = 1, in2 = 5`

• `1 \rightarrow next = llvec(5, 2)`
`(5 \rightarrow next = 6)`

`llvec(5, 2)`

• `in1 = 5, in2 = 2`

• `5 \rightarrow next = llvec(2, 6)`
`(6 \rightarrow next = 4)`

`llvec(2, 6)`

• `in1 = 2, in2 = 6`

• `2 \rightarrow next = llvec(6, 3)`
`(6 \rightarrow next = 4)`

`llvec(6, 3)`

• `in1 = 6, in2 = 3`

• `6 \rightarrow next = llvec(3, 4)`

`(6 \rightarrow next = 3)`

`llvec(3, 4)`

• `in1 = 3, in2 = 4`

• `3 \rightarrow next = llvec(4, nullptr)`

`(3 \rightarrow next = 4)`

`llvec(4, nullptr)`

• ~~`in1 = 4, in2 = nullptr`~~ 4

• returns `in1 (4)`

\rightarrow now, you can go backward since we got return value.

4

`3 \rightarrow next = 4`

`6 \rightarrow next = 3`

`2 \rightarrow next = 6`

`5 \rightarrow next = 2`

`1 \rightarrow next = 5`

`llvec(nullptr, 2)`

• `in1 = nullptr, in2 = 2`
 returns 2

2

b)

`in1 = nullptr`
`in2 = 2`