HW1

a) while loop runs depending on dummy value i which doubles every iteration

let us say that last value of i is the input. (n)

k = number of times the while loop runs for input of n.

$n = 2^{2^k}$ | $\log n = 2^k \log(2)$ ←

$\frac{\log n}{\log 2} = 2^k$ | $\log\left(\frac{\log n}{\log 2}\right) = k\log(2)$

$\frac{\log\left(\frac{\log n}{\log 2}\right)}{\log(2)} = k$ | $\Theta \log(\log(n))$

iteration while loop	i
1	2
2	4
3	16
k	$2^{2^k} = n$

b) for loop runs n times outside, for loop runs i^3 times inside

worst case = if statement is true.

runtime $= \Theta\left(\sum_{i=1}^{n} \Theta(1) + O\left(\sum_{i=0}^{n^3} \Theta(1)\right)\right) =$

The if will run $\Theta(\sqrt{n})$ of the cases of n

n input	#s if statement is true
1	1 (i=1)
4	2 (i=2, 4)
9	3 (i=3, i=6, i=9)

runtime $= \Theta(n) + \sum_{i=0}^{\sqrt{n}} \left(\sum_{i=0}^{n^3} \Theta(1)\right) = \Theta(n) + \Theta\, \sqrt{n} \cdot n^3 = \Theta n^{3.5} = \Theta(n^{3.5})$

c) two nested for loops both running n times.

one for loop that doubles m. | the if statement could be always true since we dont know contents of array

runtime $= \Theta\left(\sum_{i=1}^{n} \left(\Theta \sum_{i=1}^{n} \left(\Theta \sum_{i=1}^{\frac{\log m}{\log 2}} \Theta(1)\right)\right)\right) = \Theta(n^2 * \log(n))$

Iteration of loop	m value
1	2
2	4
3	8
n	$2^n = m$

→ $n = \frac{\log(m)}{\log(2)}$ → inner for loop runs $\Theta \log(n)$

d) for loop that runs n times / if statement true worst case.
if true, run size times,

runtime $= \Theta(1) + \Theta\left(\sum_{n=1}^{n}\left(\Theta\left(\frac{3}{2} \cdot size\right)\right) + \Theta(1)\right)$

$= \Theta(1) + \Theta\left(\sum_{n=1}^{n}(\Theta(1)) + \Theta(\log(j))\right)$

$= \Theta(n)$

Times J loop runs	j value
1	10
2	15
j	$10\left(\frac{3}{2}\right)^n$

$\frac{j}{10} = \frac{3}{2}^n$

$\log \frac{j}{10} = n \log \frac{3}{2}$

$n = \log \frac{j}{10} / \log \frac{3}{2} = \Theta(\log(j))$