Reagan Landis
CS 104
HW 1
Problem 3 Answers

a)
```
void f1(int n){
    int i=2;
    while (i<n){
        /* do something O(1) time*/
        i=i*i;
    }
}
```

k	i
1	2
2	4
3	16
4	256

$i = 2^{2^k}$

$\log i = \log(2^{2^k})$

$\log i = 2^k$

$\log(\log i) = \log(2^k)$

$\log(\log i) = k$

$O(\log(\log n))$ runtime

b)
```
void f2(int n){
    for (int i=1; i<=n; i++){
        if ( (i%(int)sqrt(n))==0){
            for(int k=0; k<pow(i,3); k++){
                /* do something that takes O(1) time*/
            }
        }
    }
}
```

$O(n)$

can execute at most $\sqrt{n}$ times true when $i = c\sqrt{n}$

n^3 times = worst case

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

$$O(n) + \sum_i O(i^3) =$$

$$O(n) + \sum_{c=1}^{\sqrt{n}} O(c\sqrt{n})^3 = O\left(n^{3/2}\sum_{c=1}^{\sqrt{n}} c^3\right) =$$

$$O(n) + O(n^{3/2} n^{4/2}) = O(n) + O(n^{7/2})$$

$O(n^{7/2})$ runtime

$\sqrt{16} = 4$

4, 8, 12, 16 ← inc. by 4, 4 times

$\sqrt{36} = 6$

6, 12, 18, 24, 30, 36 ← inc. by 6, 6 times

c)
```
for (int i=1; i<=n; i++){        // O(n)
  for (int k=1; k<=n; k++){      // O(n)   } O(n^2)
    if (A[k]==i){                // true MAX n times
      for (int m=1; m<=n; m=m+m){   } O(logn)
        // O(1) function
      }                          } O(nlogn)
    }
  }
}
```

$O(n^2) + O(n \log n)$

$O(n^2)$ runtime

k	m
1	1
2	2
3	4
4	8
5	16
	n

$2^{k-1} = n$

$\log 2^{k-1} = n$

$k-1 = n$

$O(\log n)$

2·2·2·2·2
4·4·2
16·2

d)
```
int f(int n){
  int *a = new int[10];
  int size=10;
  for (int i=0; i<n; i++){
    if (i==size){
      int newsize= 3*size/2;
      int*b= new int[newsize];
      for (int j=0; j<size; j++)  b[j]=a[j];
      delete[] a;
      a=b;
      size= new size;
    }
    a[i] = i*i;
  }
}
```

i	size
0	10
1	15
2	22
3	33
4	50

$n = 10\left(\frac{3}{2}\right)^j$

$\log_{3/2}\left(\frac{n}{10}\right) = \log_{3/2}\left(\frac{3}{2}\right)^j$

$\log_{3/2}\left(\frac{n}{10}\right) = j$

$$O\left(\sum_{i=0}^{\log_{3/2}\left(\frac{n}{10}\right)} 10 \cdot \frac{3}{2}^j\right) + O(n)$$

↑ runtime when if statement executes

↑ runtime when if statement doesn't execute

$$O\left(10 \cdot \left(\frac{3}{2}\right)^{\log_{3/2}\left(\frac{n}{10}\right)}\right)$$

$$O\left(10 \cdot \frac{n}{10}\right)$$

$O(n) + O(n) = O(n)$ runtime