a.

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

① ② ③ ④ ⑤ ⑥
2　4　16　⚡6　⚡6²

2^1　2^2　2^4　2^8　2^{16}　$2^{2^{x-1}}$

$$2^{2^{x-1}} = n$$

$$2^{x-1} = \log_2 n$$

$$x = \log_2 (\log_2 n) + 1$$

since　　$i < n$　　not　$i \geq n$

$$x = \log_2 (\log_2 n)$$

The runtime is $\log_2 (\log_2 n)$

(b)

```
void f2(int n)
{
  n for(int i=1; i <= n; i++){
     |if( (i % (int)sqrt(n)) == 0){  } n/√n
           for(int k=0; k < pow(i,3); k++) {
     n³    /* do something that takes O(1) time */
           }
         }
      }
}
```

$$1 \% (int) sqrt(n) == 0 \implies n/\sqrt{n} \text{ True(s)}$$

$$\text{Runtime is } \theta\left(\sqrt{n} \cdot n^3\right) = \theta\left(n^{\frac{7}{2}}\right)$$

(c)

```
for(int i=1; i <= n; i++){
    for(int k=1; k <= n; k++){
        if( A[k] == i){
            for(int m=1; m <= n; m=m+m){
                // do something that takes O(1) time
                // Assume the contents of the A[] array are
                not changed
            }
        }
    }
}
```

n (for i loop)
n (for k loop)
n (brace)
$1 + \log_2 n$ (for m loop)

$(m=1; m \le n; m=m+m)$

$m = 1 \quad 2 \quad 4 \quad 8 \quad 16 \quad 2^x \quad n$

$2^{x-1} = n$

$x - 1 = \log_2 n$

$x = \log_2 n + 1$

inner loop runtime is $O(\log_2 n)$

if $(A[k] == i)$ can be true for

at most n times

Runtime is: $n^2 + n \log_2 n$

if statement $\qquad$ entered inner loop

$\Theta(n^2) + \Theta(n \log_2 n) = \Theta(n^2)$

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 = newsize;
                }
            a[i] = i*i;
        }
}
```

for the if $(i == size)$, assume: x times this will be strue

$$
\begin{array}{cccccc}
1 & 2 & 3 & 4 & & x \\
10 & \frac{3}{2} \times 10 & (\frac{3}{2})^2 \times 10 & (\frac{3}{2})^3 \times 10 & & (\frac{3}{2})^{x-1} \times 10 = n
\end{array}
$$

$$x = \log_{(3/2)} \left(\frac{n}{10}\right) + 1$$

inner for loop:

Runtime $u_b =$ $\quad 10 \quad \frac{3}{2} \times 10 \quad (\frac{3}{2})^2 \times 10 \quad (\frac{3}{2})^{x-1} \times 10$

Total Runtime : if statement + entered if statements

$$\Theta(n) \quad + \Theta\left(\sum_{i=1}^{x} 10 \left(\frac{3}{2}\right)^{x-1}\right)$$

$$= \Theta(n) + \Theta\left(\sum_{i=0}^{x-1} 10 \cdot \left(\frac{3}{2}\right)^x\right)$$

$$= \Theta(n) + \Theta\left(\frac{10 \left(\frac{3}{2}\right)^x - 10}{\frac{3}{2} - 1}\right)$$

$$= \Theta(n) + \Theta\left(20 \times \left(\frac{3}{2}\right)^{\log_{3/2}\left(\frac{n}{10}\right)+1} - 20\right)$$

$$= \Theta(n) + \Theta\left(20 \times \left(\frac{n}{10} + 1\right) - 20\right) \quad = \Theta(n)$$