Takeaway from class-

Increment => $O(n)\ (i = i + c)$

Doubling => $O(\log n)\ (i = 2 * i)$

Exponentiation => $O(\log \log n)\ (i = i * i)$

Consider the following program fragment:

```
for (int i=0; i * i < N; i++)
    for (int j=0; j * j < N; j++)
        printf("Hello!");
```

Which one of the following statements about the runtime $R(N)$ is true?

A $R(N) = \Theta(\log N)$

B $R(N) = \Theta(\sqrt{N})$

```
for( int i = n; i > 0; i /= 2 ) {
    for( int j = 1; j < n; j *= 2 ) {
        for( int k = 0; k < n; k += 2 ) {
            ... // constant number of operations
        }
    }
}
```

```
for ( int k = n; k >0; k /= 3 ) {
  for ( int i = 0; i < n; i += 2 ) {
     // constant number C of elementary operations
  }
  for ( int j = 2; j < n; j = (j*j)) {
     // constant number C of elementary operations
  }
}
```

```
int i=1;
while (i<= n) {
    int j = i;
    while (j > 0) {
        j = j/2;
    }
    i++;
}
```

```
for (k = 1; k <= n; k += 1)
{
    for (i = 1; i <= n; i *= 3)
    {
        j = i;
        while (j > 1)
        {
            sum += 1;
            j /= 3;
        }
    }
```

```
for(int i = 0; i < N*N; i++) {
    for(int j = 0; j < i; j++) {
        //something O(1)
    }
}
```

```
for (int i = 0; i < n; i++) {
    for (int j = 0; j < n * n; j++) {
        for (int k = 0; k < j; k++) {
            sum++;
        }
    }
}
```

```
i = 1;
k = 1;

while(k<n){
      k = k+ i;
      i = i + 1;
}
```

```
for(int i =1; i<=n;i++)
  {
   for(int j=i ; j<=n; j+=i*2);
  }
```

```
p = 0
for( i=1; i<n; i=i*2 ) {
    p++
}

for( j=1; j<p; j=j*2 ) {
    some_statement
}
```

```
for (int j = 2; j < N; j++) {
    for (int k = 2*j; k <= N; k += j) {
        some_statement
    }
}
```

```
int sum = 0;
for (int i = 1; i < n; i++) {
    for (int j = 0; j < n/i; j++) {
        sum++;
    }
}
```

```
for (i=1;i<=n;i*=2){
   for (j=1;j<=i;j++) {
      // some O(1) operation
   }
}
```

```
for (int i = 1; i < n; i*=2)
  for (int j = 0; j < i; j +=2)
  {
    // some contstant time operations
  }
```

```
for ( int i = 1; i < n*n*n; i *= n ) {
    for ( int j = 0; j < n; j += 2 ) {
        for ( int k = 1; k < n; k *= 3 ) {

            // some contstant time operations
        }
    }
}
```

```
for(int i = 0; i < n^3; i++){
    if(i % 3 == 0){
        break;
    }
    else{
        print ":D"
    }
}
```

```
for(i = 1; i < n; i = i * 2) {
      for(j = 1; j < i; j++) {
            sum++;
      }
}
```

```
int n;
int sum;
for (int i = 1; i < n; i++)
{
  for (int j = 0; j < i*i; j++)
    {
      if (j % i == 0)
        {
          for (int k = 0; k < j; k++)
```

```
for (i = 1; i <= N; i = i*2)
    for (j = 1; j <= i^2; j=j*2)
        sum++;
```

```
for ( i=1; i < n; i *= 2 )
for ( j = n; j > 0; j /= 2 )
for ( k = j; k < n; k += 2 ) {
sum += (i + j * k );
}
```