QB2

Big Oh with logs and roots

Do NOT write on this test. Record all answers on the bubble sheet. **Closed book. No notes.** Work strictly from memory. No time limit. **No calculators. Scratch paper okay.**

1/2p. Give a tight big-oh $\Theta()$ bound on the running time T(n) of this program.
Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.
(A) $n^2 \lg n$ (C) $n\sqrt{n} \lg n$ (E) $n \lg n$ (G) $\sqrt{n}$ (I) $\lg n$
(B) n^2 (D) $n\sqrt{n}$ (F) $\sqrt{n} \lg n$ (H) $\lg^2 n$ (J) 1

```
int main ( int argc, char * * argv ) {
   int n = atoi(argv[1]);
   c = 1; do {
      j = 1; do {
         if ( simpleCompare ) {
            simpleStatement;
         } else {
            simpleStatement;
         }
      j += 5; } while ( j < n );
   c += 3; } while ( c * c < n );
   return 0; }
```

2/2p. Give a tight big-oh $\Theta()$ bound on the running time T(n) of this program.
Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.
(A) $n^2\sqrt{n}$ (C) $n\sqrt{n}$ (E) $n \lg n$ (G) $\sqrt{n}$ (I) $\lg^2 n$
(B) n^2 (D) $n \lg^2 n$ (F) n (H) $\lg^3 n$ (J) $\lg n$

```
int main ( int argc, char * * argv ) {
   int n = atoi(argv[1]);
   e = n; do {
      if ( simpleCompare ) {
         k = 1; do {
            d = 1; while ( d * d < n ) {
               if ( simpleCompare ) {
                  if ( simpleCompare ) {
                     simpleStatement;
                  }
               }
            d++; }
         k += 1; } while ( k < n );
      } else {
         b = 1; while ( b < n ) {
            for ( f = 1 ; f < n ; f *= 3 ) {
               simpleStatement;
            }
         b += 3; }
      }
   e -= 3; } while ( e > 1 );
   return 0; }
```

QB2

3/2p. Give a tight big-oh $\Theta()$ bound on the running time T(n) of this program.
Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

(A) n^6 (C) $n^3\sqrt{n}$ (E) $n^3 \lg^2 n$ (G) $n^2\sqrt{n}\lg^3 n$ (I) $n \lg^6 n$
(B) $n^3\sqrt{n}\lg^2 n$ (D) $n^3 \lg^4 n$ (F) $n^3 \lg n$ (H) $n^2 \lg n$ (J) $n \lg n$

```
int main ( int argc, char * * argv ) {
   int n = atoi(argv[1]);
   j = 1; do {
      f = 1; do {
         if ( simpleCompare ) {
            if ( simpleCompare ) {
               d = 1; do {
                  h = 1; while ( h < n ) {
                     if ( simpleCompare ) {
                        if ( simpleCompare ) {
                           a = n; while ( a > 1 ) {
                              if ( simpleCompare ) {
                                 simpleStatement;
                              } else {
                                 simpleStatement;
                              }
                           a /= 3; }
                        } else {
                           if ( simpleCompare ) {
                              simpleStatement;
                           } else {
                              simpleStatement;
                           }
                        }
                     } else {
                        c = 1; do {
                           for ( i = 1 ; i * i < n ; i += 3 ) {
                              b = n; while ( b > 1 ) {
                                 simpleStatement;
                              b /= 2; }
                           }
                        c *= 3; } while ( c < n );
                     }
                  h++; }
               d++; } while ( d * d < n );
            }
         }
      f *= 5; } while ( f < n );
   j++; } while ( j * j < n );
   return 0; }
```

QB2

4/2p. Give a tight big-oh $\Theta()$ bound on the running time T(n) of this program.
Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.
(A) $n^2\sqrt{n}$ (C) n^2 (E) $n\sqrt{n}$ (G) $\sqrt{n}\lg n$ (I) $\lg^2 n$
(B) $n^2 \lg n$ (D) $n\sqrt{n}\lg n$ (F) $n \lg n$ (H) $\sqrt{n}$ (J) $\lg n$

```
int main ( int argc, char * * argv ) {
   int n = atoi(argv[1]);
   if ( simpleCompare ) {
      j = n; while ( j > 1 ) {
         if ( simpleCompare ) {
            if ( simpleCompare ) {
               for ( c = n ; c > 1 ; c /= 2 ) {
                  h = n; do {
                     simpleStatement;
                  h /= 2; } while ( h > 1 );
               }
            }
         } else {
            g = 1; while ( g < n ) {
               simpleStatement;
            g += 10; }
         }
      j -= 5; }
   } else {
      if ( simpleCompare ) {
         if ( simpleCompare ) {
            if ( simpleCompare ) {
               simpleStatement;
            }
         } else {
            simpleStatement;
         }
      } else {
         if ( simpleCompare ) {
            simpleStatement;
         } else {
            simpleStatement;
         }
      }
   }
   return 0; }
```

QB2

5/2p. Give a tight big-oh $\Theta()$ bound on the running time T(n) of this program.
Assume `atoi`, `simpleStatement`, and `simpleCompare` each run in $\Theta(1)$ time.

(A) n^4 (C) $n^2\sqrt{n}\lg n$ (E) $n^2\lg n$ (G) $n\sqrt{n}\lg n$ (I) $\sqrt{n}\lg^3 n$
(B) n^3 (D) $n^2\lg^2 n$ (F) $n\sqrt{n}\lg^2 n$ (H) n (J) $\lg^4 n$

```
int main ( int argc, char * * argv ) {
   int n = atoi(argv[1]);
   for ( d = n ; d > 1 ; d /= 2 ) {
      if ( simpleCompare ) {
         if ( simpleCompare ) {
            if ( simpleCompare ) {
               a = n; while ( a > 1 ) {
                  if ( simpleCompare ) {
                     simpleStatement;
                  } else {
                     simpleStatement;
                  }
               a /= 5; }
            } else {
               simpleStatement;
            }
         } else {
            if ( simpleCompare ) {
               if ( simpleCompare ) {
                  simpleStatement;
               } else {
                  simpleStatement;
               }
            } else {
               simpleStatement;
            }
         }
      } else {
         g = n; while ( g > 1 ) {
            if ( simpleCompare ) {
               for ( k = 1 ; k * k < n ; k++ ) {
                  e = 1; while ( e < n ) {
                     simpleStatement;
                  e++; }
               }
            } else {
               simpleStatement;
            }
         g /= 5; }
      }
   }
   return 0; }
```

QB2

Total points 10.

QB2

Answer Key (points per line)

1 (2).	`D` $(n\sqrt{n})$
2 (2).	`A` $(n^2\sqrt{n})$
3 (2).	`G` $(n^2\sqrt{n}\lg^3 n)$
4 (2).	`C` (n^2)
5 (2).	`F` $(n\sqrt{n}\lg^2 n)$

Total points 10.