

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^3 (C) $n\sqrt{n}$ (E) $n \lg n$ (G) $\sqrt{n} \lg n$ (I) $\lg^2 n$
 (B) $n^2 \lg n$ (D) $n \lg^2 n$ (F) $\sqrt{n} \lg^2 n$ (H) \sqrt{n} (J) 1

```
int main ( int argc, char * * argv ) {
  int n = atoi(argv[1]);
  for ( k = 1 ; k < n ; k *= 2 ) {
    b = 1; while ( b < n ) {
      c = n; do {
        if ( simpleCompare ) {
          if ( simpleCompare ) {
            if ( simpleCompare ) {
              simpleStatement;
            }
          } else {
            simpleStatement;
          }
        }
        c /= 2; } while ( c > 1 );
      b += 5; }
    }
  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^3 (C) n^2 (E) $n\sqrt{n}$ (G) $n \lg n$ (I) $\lg^2 n$
 (B) $n^2 \lg n$ (D) $n\sqrt{n} \lg n$ (F) $n \lg^2 n$ (H) n (J) 1

```
int main ( int argc, char * * argv ) {
  int n = atoi(argv[1]);
  h = n; while ( h > 1 ) {
    if ( simpleCompare ) {
      f = 1; do {
        for ( b = 1 ; b * b < n ; b += 1 ) {
          simpleStatement;
        }
        f += 5; } while ( f < n );
    } else {
      if ( simpleCompare ) {
        simpleStatement;
      }
    }
    h /= 2; }
  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) $n^2\sqrt{n}$ (C) $n\sqrt{n}\lg n$ (E) $n\lg n$ (G) \sqrt{n} (I) $\lg n$
(B) n^2 (D) $n\sqrt{n}$ (F) n (H) $\lg^2 n$ (J) 1

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    if ( simpleCompare ) {
        i = 1; while ( i * i < n ) {
            if ( simpleCompare ) {
                if ( simpleCompare ) {
                    if ( simpleCompare ) {
                        if ( simpleCompare ) {
                            simpleStatement;
                        } else {
                            simpleStatement;
                        }
                    } else {
                        simpleStatement;
                    }
                } else {
                    simpleStatement;
                }
            } else {
                if ( simpleCompare ) {
                    simpleStatement;
                } else {
                    simpleStatement;
                }
            }
        }
        i++; }
    } else {
        if ( simpleCompare ) {
            if ( simpleCompare ) {
                d = n; do {
                    if ( simpleCompare ) {
                        if ( simpleCompare ) {
                            simpleStatement;
                        }
                    } else {
                        simpleStatement;
                    }
                }
                d /= 3; } while ( d > 1 );
            }
        }
    }
    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) $n^2 \lg^2 n$ (C) n^2 (E) $n\sqrt{n} \lg n$ (G) n (I) $\sqrt{n} \lg n$
(B) $n^2 \lg n$ (D) $n\sqrt{n} \lg^2 n$ (F) $n \lg^3 n$ (H) $\sqrt{n} \lg^3 n$ (J) \sqrt{n}

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    f = 1; do {
        a = 1; while ( a < n ) {
            g = 1; do {
                if ( simpleCompare ) {
                    if ( simpleCompare ) {
                        if ( simpleCompare ) {
                            if ( simpleCompare ) {
                                b = 1; do {
                                    simpleStatement;
                                    b++; } while ( b * b < n );
                                } else {
                                    simpleStatement;
                                }
                            } else {
                                if ( simpleCompare ) {
                                    simpleStatement;
                                } else {
                                    simpleStatement;
                                }
                            }
                        }
                    }
                }
            } else {
                i = 1; do {
                    if ( simpleCompare ) {
                        if ( simpleCompare ) {
                            simpleStatement;
                        }
                    } else {
                        simpleStatement;
                    }
                    i++; } while ( i * i < n );
                }
                g *= 3; } while ( g < n );
            a++; }
        f *= 2; } while ( f < n );
    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) $n^2\sqrt{n}$ (C) n^2 (E) $n \lg n$ (G) \sqrt{n} (I) $\lg n$
(B) $n^2 \lg n$ (D) $n\sqrt{n} \lg n$ (F) n (H) $\lg^2 n$ (J) 1

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

QB2

Bro Colton

CS 201 Big Oh with logs and roots

5

Total points 10.

QB2

Answer Key (points per line)

- | | |
|--------|---------------------------|
| 1 (2). | D ($n \lg^2 n$) |
| 2 (2). | D ($n\sqrt{n} \lg n$) |
| 3 (2). | G (\sqrt{n}) |
| 4 (2). | D ($n\sqrt{n} \lg^2 n$) |
| 5 (2). | F (n) |

Total points 10.