

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

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

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

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

```
int main ( int argc, char * * argv ) {
    int n = atoi(argv[1]);
    k = 1; do {
        b = 1; do {
            if ( simpleCompare ) {
                g = 1; do {
                    i = n; do {
                        if ( simpleCompare ) {
                            if ( simpleCompare ) {
                                e = 1; do {
                                    c = 1; while ( c < n ) {
                                        simpleStatement;
                                        c *= 5; }
                                    e += 3; } while ( e < n );
                                } else {
                                    simpleStatement;
                                }
                            }
                        i -= 1; } while ( i > 1 );
                    g += 10; } while ( g * g < n );
                } else {
                    if ( simpleCompare ) {
                        if ( simpleCompare ) {
                            if ( simpleCompare ) {
                                if ( simpleCompare ) {
                                    simpleStatement;
                                }
                            }
                        } else {
                            simpleStatement;
                        }
                    } else {
                        f = n; do {
                            simpleStatement;
                            f /= 5; } while ( f > 1 );
                    }
                }
            b += 2; } while ( b * b < n );
        k *= 2; } while ( k < 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^4 (C) $n^2 \lg^2 n$ (E) $n\sqrt{n} \lg^2 n$ (G) $n \lg^3 n$ (I) $\sqrt{n} \lg n$
(B) n^3 (D) $n^2 \lg n$ (F) $n\sqrt{n}$ (H) $n \lg^2 n$ (J) $\lg n$

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

QB2

Bro Colton

CS 201 Big Oh with logs and roots

5

Total points 10.

q10 v1115959560

QB2

Answer Key (points per line)

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

Total points 10.