

NPM: _____
Nama: _____

Kode Soal: WWW
Kelas: _____

Kuis 1
Struktur Data dan Algoritma
Fakultas Ilmu Komputer Universitas Indonesia
Durasi: 30 menit

Petunjuk: Tuliskan jawaban Anda sesingkat-singkatnya (**tanpa penjelasan apapun!**) di dalam kotak jawaban yang disediakan pada kolom paling kanan.

No.	Soal	Jawaban
1.	Urutkan kompleksitas algoritma berikut ini, mulai dari yang paling rendah (paling kecil <i>growth rate</i> -nya) sampai yang paling tinggi (paling besar <i>growth rate</i> -nya). a. $O(N^2 \cdot \log \sqrt{N})$ b. $O(\log(2^N) \cdot \log(10^N))$ c. $O(\log N^3 \cdot \log N^2)$ d. $O(\sqrt{N} \log N^2)$ e. $O(N!)$ f. $O(2^N (N/2)!)$	Tuliskan huruf-hurufnya saja:
2.	Algoritma A dan B kompleksitasnya sama yaitu $O(N^2)$, tentukan apakah pernyataan berikut ini BENAR atau SALAH:	BENAR/SALAH:
	a. Untuk data yang kecil waktu eksekusi A pasti sama dengan B.	
	b. Untuk data yang besar waktu eksekusi A pasti sama dengan B.	
	c. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan waktu eksekusi B.	
	d. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan 4 kali waktu eksekusi B.	
3.	Apakah kompleksitas waktu potongan kode di bawah ini dalam notasi Big-Oh? <pre>for (int i = 1; i <= N; i *= 3) { for (int j = 1; j <= i*N; j += i) { System.out.println(i+" "+j); } }</pre>	
4.	Diberikan fungsi rekursif berikut: <pre>int f(int n) { if (n <= 1) return n; else if (n <= 5) return f(n-2)+n; else return f(n/2)+n; }</pre> Berapakah nilai $f(25)$?	

5.	<p>Diberikan fungsi rekursif algoritma binary search sebagai berikut:</p> <pre> int binsearch(data[], x, low, high){ mid = (low+high) / 2; if(data[mid] == x) return mid; else if (x < data[mid]) return binsearch(data[], x, low, mid); else return binsearch(data[], x, mid+1, high); } </pre> <p>Jika <i>a</i> adalah sebuah <i>sorted array</i> berisi bilangan 1 sampai dengan 100 (berturut-turut), berapa kali <i>recursive call</i> yang terjadi pada pemanggilan <code>binsearch(a, 80, 0, a.length-1)</code> termasuk pemanggilan pertama?</p>	
6.	<p>Apa output dari potongan program di bawah ini?</p> <pre> String[] values = {"Sumatera", "Kalimantan", "Jawa", "Sulawesi", "Bali", "Papua"}; Stack<String> stackSatu = new Stack<String> (); Deque<String> queueDua = new ArrayDeque<String> (); for(int i = 0; i < values.length; i++) { if(i % 2 == 0) { stackSatu.push(values[i]); } else { queueDua.add(values[i]); } } while(!stackSatu.empty()) { String x = stackSatu.pop(); queueDua.add(x); } while(!queueDua.isEmpty()) { System.out.println(queueDua.poll()); } </pre>	
7.	<p>Jika <i>swapping</i> adalah proses menukarkan isi dua elemen array, berapa kali dilakukan proses <i>swapping</i> oleh algoritma Bubble-sort jika dijalankan untuk mengurutkan (dari kecil ke besar) array berikut ini: {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}?</p>	

NPM: _____
Nama: _____

Kode Soal: XXX
Kelas: _____

Kuis 1
Struktur Data dan Algoritma
Fakultas Ilmu Komputer Universitas Indonesia
Durasi: 30 menit

Petunjuk: Tuliskan jawaban Anda sesingkat-singkatnya (**tanpa penjelasan apapun!**) di dalam kotak jawaban yang disediakan pada kolom paling kanan.

No.	Soal	Jawaban
1.	Urutkan kompleksitas algoritma berikut ini, mulai dari yang paling rendah (paling kecil <i>growth rate</i> -nya) sampai yang paling tinggi (paling besar <i>growth rate</i> -nya). a. $O(\sqrt{N} \log N^2)$ b. $O(N!)$ c. $O(2^N (N/2)!)$ d. $O(N^2 \cdot \log \sqrt{N})$ e. $O(\log(2^N) \cdot \log(10^N))$ f. $O(\log N^3 \cdot \log N^2)$	Tuliskan huruf-hurufnya saja:
2.	Algoritma A dan B kompleksitasnya sama yaitu $O(N^2)$, tentukan apakah pernyataan berikut ini BENAR atau SALAH:	BENAR/SALAH:
	a. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan waktu eksekusi B.	
	b. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan 4 kali waktu eksekusi B.	
	c. Untuk data yang kecil waktu eksekusi A pasti sama dengan B.	
	d. Untuk data yang besar waktu eksekusi A pasti sama dengan B.	
3.	Apakah kompleksitas waktu potongan kode di bawah ini dalam notasi Big-Oh? <pre>for (int i = 1; i <= N; i *= 4) { for (int j = 1; j <= i*N; j += i) { System.out.println(i+" "+j); } }</pre>	
4.	Diberikan fungsi rekursif berikut: <pre>int f(int n) { if (n <= 1) return n; else if (n <= 5) return f(n-2)+n; else return f(n/2)+n; }</pre> Berapakah nilai $f(28)$?	

5.	<p>Diberikan fungsi rekursif algoritma binary search sebagai berikut:</p> <pre> int binsearch(data[], x, low, high){ mid = (low+high) / 2; if(data[mid] == x) return mid; else if (x < data[mid]) return binsearch(data[], x, low, mid); else return binsearch(data[], x, mid+1, high); } </pre> <p>Jika <i>a</i> adalah sebuah <i>sorted array</i> berisi bilangan 1 sampai dengan 100 (berturut-turut), berapa kali <i>recursive call</i> yang terjadi pada pemanggilan <code>binsearch(a, 84, 0, a.length-1)</code> termasuk pemanggilan pertama?</p>	
6.	<p>Apa output dari potongan program di bawah ini?</p> <pre> String[] values = {"Sumatera", "Kalimantan", "Jawa", "Sulawesi", "Bali", "Papua"}; Stack<String> stackSatu = new Stack<String> (); Deque<String> queueDua = new ArrayDeque<String> (); for(int i = 0; i < values.length; i++) { if(i % 2 == 0) { stackSatu.push(values[i]); } else { queueDua.add(values[i]); } } while(!queueDua.isEmpty()) { String x = queueDua.poll(); stackSatu.push(x); } while(!stackSatu.empty()) { System.out.println(stackSatu.pop()); } </pre>	
7.	<p>Jika <i>swapping</i> adalah proses menukarkan isi dua elemen array, berapa kali dilakukan proses <i>swapping</i> oleh algoritma Selection-sort jika dijalankan untuk mengurutkan (dari kecil ke besar) array berikut ini: {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}?</p>	

NPM: _____
Nama: _____

Kode Soal: YYY
Kelas: _____

Kuis 1
Struktur Data dan Algoritma
Fakultas Ilmu Komputer Universitas Indonesia
Durasi: 30 menit

Petunjuk: Tuliskan jawaban Anda sesingkat-singkatnya (**tanpa penjelasan apapun!**) di dalam kotak jawaban yang disediakan pada kolom paling kanan.

No.	Soal	Jawaban
1.	Urutkan kompleksitas algoritma berikut ini, mulai dari yang paling rendah (paling kecil <i>growth rate</i> -nya) sampai yang paling tinggi (paling besar <i>growth rate</i> -nya). a. $O(\log N^3 \cdot \log N^2)$ b. $O(\sqrt{N} \log N^2)$ c. $O(N!)$ d. $O(2^N (N/2)!)$ e. $O(N^2 \cdot \log \sqrt{N})$ f. $O(\log(2^N) \cdot \log(10^N))$	Tuliskan huruf-hurufnya saja:
2.	Algoritma A dan B kompleksitasnya sama yaitu $O(N^2)$, tentukan apakah pernyataan berikut ini BENAR atau SALAH:	BENAR/SALAH:
	a. Untuk data yang besar waktu eksekusi A pasti sama dengan B.	
	b. Untuk data yang kecil waktu eksekusi A pasti sama dengan B.	
	c. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan 4 kali waktu eksekusi B.	
	d. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 2N, waktu eksekusi A akan sama dengan waktu eksekusi B.	
3.	Apakah kompleksitas waktu potongan kode di bawah ini dalam notasi Big-Oh? <pre>for (int i = 1; i <= N; i *= 5) { for (int j = 1; j <= i*N; j += i) { System.out.println(i+" "+j); } }</pre>	
4.	Diberikan fungsi rekursif berikut: <pre>int f(int n) { if (n <= 1) return n; else if (n <= 5) return f(n-2)+n; else return f(n/2)+n; }</pre> Berapakah nilai $f(31)$?	

5.	<p>Diberikan fungsi rekursif algoritma binary search sebagai berikut:</p> <pre>int binsearch(data[], x, low, high){ mid = (low+high) / 2; if(data[mid] == x) return mid; else if (x < data[mid]) return binsearch(data[], x, low, mid); else return binsearch(data[], x, mid+1, high); }</pre> <p>Jika <i>a</i> adalah sebuah <i>sorted array</i> berisi bilangan 1 sampai dengan 100 (berturut-turut), berapa kali <i>recursive call</i> yang terjadi pada pemanggilan <code>binsearch(a, 91, 0, a.length-1)</code> termasuk pemanggilan pertama?</p>	
6.	<p>Apa output dari potongan program di bawah ini?</p> <pre>String[] values = {"Sumatera", "Kalimantan", "Jawa", "Sulawesi", "Bali", "Papua"}; Deque<String> queueSatu = new ArrayDeque<String> (); Stack<String> stackDua = new Stack<String> (); for(int i = 0; i < values.length; i++) { if(i % 2 == 0) { queueSatu.add(values[i]); } else { stackDua.push(values[i]); } } while(!stackDua.empty()) { String x = stackDua.pop(); queueSatu.add(x); } while(!queueSatu.isEmpty()) { System.out.println(queueSatu.poll()); }</pre>	
7.	<p>Jika <i>swapping</i> adalah proses menukarkan isi dua elemen array, berapa kali dilakukan proses <i>swapping</i> oleh algoritma Bubble-sort jika dijalankan untuk mengurutkan (dari kecil ke besar) array berikut ini: {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}?</p>	

NPM: _____
Nama: _____

Kode Soal: ZZZ
Kelas: _____

Kuis 1
Struktur Data dan Algoritma
Fakultas Ilmu Komputer Universitas Indonesia
Durasi: 30 menit

Petunjuk: Tuliskan jawaban Anda sesingkat-singkatnya (**tanpa penjelasan apapun!**) di dalam kotak jawaban yang disediakan pada kolom paling kanan.

No.	Soal	Jawaban
1.	Urutkan kompleksitas algoritma berikut ini, mulai dari yang paling rendah (paling kecil <i>growth rate</i> -nya) sampai yang paling tinggi (paling besar <i>growth rate</i> -nya). a. $O(N!)$ b. $O(2^N (N/2)!)$ c. $O(N^2 \cdot \log \sqrt{N})$ d. $O(\log(2^N) \cdot \log(10^N))$ e. $O(\log N^3 \cdot \log N^2)$ f. $O(\sqrt{N} \log N^2)$	Tuliskan huruf-hurufnya saja:
2.	Algoritma A dan B kompleksitasnya sama yaitu $O(N^2)$, tentukan apakah pernyataan berikut ini BENAR atau SALAH:	BENAR/SALAH:
	a. Untuk data yang besar waktu eksekusi A pasti sama dengan B.	
	b. Untuk data yang kecil waktu eksekusi A pasti sama dengan B.	
	c. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 3N, waktu eksekusi A akan sama dengan waktu eksekusi B.	
	d. Jika untuk data berukuran N yang besar waktu eksekusi A dua kali waktu eksekusi B, maka untuk ukuran data 3N, waktu eksekusi A akan sama dengan 6 kali waktu eksekusi B.	
3.	Apakah kompleksitas waktu potongan kode di bawah ini dalam notasi Big-Oh? <pre>for (int i = 1; i <= N; i *= 6) { for (int j = 1; j <= i*N; j += i) { System.out.println(i+" "+j); } }</pre>	
4.	Diberikan fungsi rekursif berikut: <pre>int f(int n) { if (n <= 1) return n; else if (n <= 5) return f(n-2)+n; else return f(n/2)+n; }</pre> Berapakah nilai $f(35)$?	

5.	<p>Diberikan fungsi rekursif algoritma binary search sebagai berikut:</p> <pre> int binsearch(data[], x, low, high){ mid = (low+high) / 2; if(data[mid] == x) return mid; else if (x < data[mid]) return binsearch(data[], x, low, mid); else return binsearch(data[], x, mid+1, high); } </pre> <p>Jika <i>a</i> adalah sebuah <i>sorted array</i> berisi bilangan 1 sampai dengan 100 (berturut-turut), berapa kali <i>recursive call</i> yang terjadi pada pemanggilan <code>binsearch(a, 97, 0, a.length-1)</code> termasuk pemanggilan pertama?</p>	
6.	<p>Apa output dari potongan program di bawah ini?</p> <pre> String[] values = {"Sumatera", "Kalimantan", "Jawa", "Sulawesi", "Bali", "Papua"}; Deque<String> queueSatu = new ArrayDeque<String> (); Stack<String> stackDua = new Stack<String> (); for(int i = 0; i < values.length; i++) { if(i % 2 == 0) { queueSatu.add(values[i]); } else { stackDua.push(values[i]); } } while(!queueSatu.isEmpty()) { String x = queueSatu.poll(); stackDua.push(x); } while(!stackDua.empty()) { System.out.println(stackDua.pop()); } </pre>	
7.	<p>Jika <i>swapping</i> adalah proses menukarkan isi dua elemen array, berapa kali dilakukan proses <i>swapping</i> oleh algoritma Selection-sort jika dijalankan untuk mengurutkan (dari kecil ke besar) array berikut ini: {10, 9, 8, 7, 6, 5, 4, 3, 2, 1}?</p>	