**Analisis Algoritma**

Tugas 4



­­

**Dibuat oleh:**

Muhammad Iqbal Alif Fadilla

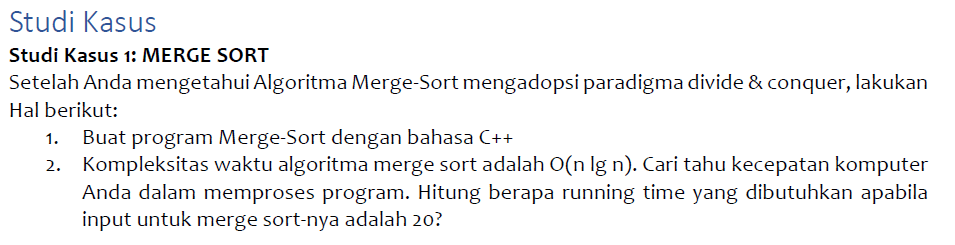
140810180020

**Universitas Padjadjaran**

**Fakultas Matematika dan Ilmu Pengetahuan**

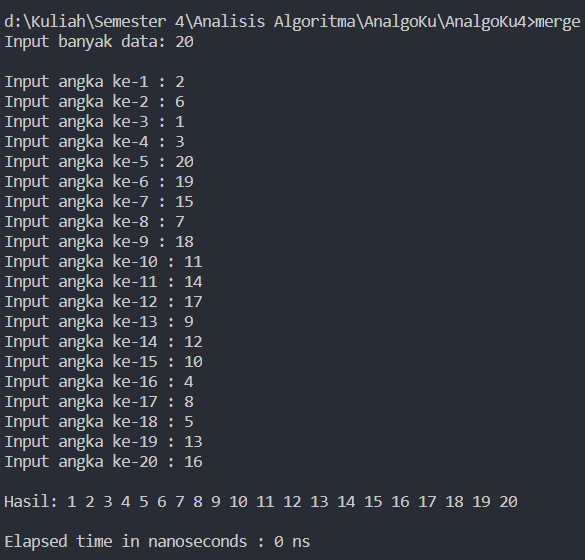
**2020**

**Worksheet 4**



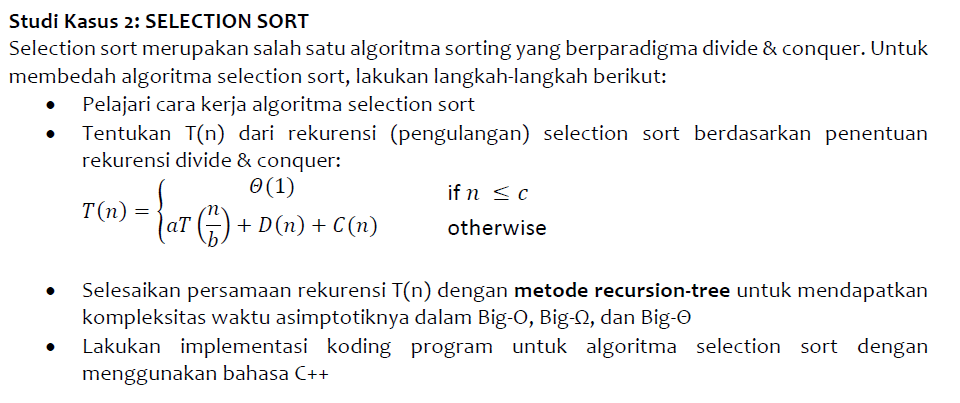
* Program C++

1. /\*
2. Nama        : Muhammad Iqbal Alif Fadilla
3. Kelas       : B
4. NPM         : 140810180020
5. Deskripsi   : Merge Sort
6. \*/
8. #include <iostream>
9. #include <chrono>
10. **using** **namespace** std;
12. **void** satu(**int** \*in, **int** p, **int** q, **int** r)
13. {
14. **int** n1 = q - p + 1;
15. **int** n2 = r - q;
16. **int** L[n1 + 1];
17. **int** R[n2 + 1];
18. **for** (**int** i = 1; i <= n1; i++)
19. {
20. L[i - 1] = in[(p - 1) + i - 1];
21. }
23. **for** (**int** j = 1; j <= n2; j++)
24. {
25. R[j - 1] = in[(q - 1) + j];
26. }
28. **int** i = 0;
29. **int** j = 0;
30. L[n1] = 2147483647;
31. R[n2] = 2147483647;
33. **for** (**int** k = (p - 1); k < r; k++)
34. {
35. **if** (L[i] <= R[j])
36. {
37. in[k] = L[i];
38. i = i + 1;
39. }
40. **else**
41. {
42. in[k] = R[j];
43. j = j + 1;
44. }
45. }
46. }
48. **void** msort(**int** \*in, **int** p, **int** r)
49. {
50. **int** q;
51. **if** (p < r)
52. {
53. q = (p + r) / 2;
54. msort(in, p, q);
55. msort(in, q + 1, r);
57. satu(in, p, q, r);
58. }
59. }
61. **void** input(**int** \*a, **int** &n)
62. {
63. cout << "Input banyak data: ";
64. cin >> n;
65. cout << "\n";
66. **for** (**int** i = 0; i < n; i++)
67. {
68. cout << "Input angka ke-" << i + 1 << " : ";
69. cin >> a[i];
70. }
71. }
73. **int** main()
74. {
75. **int** in[100];
76. **int** n;
77. input(in, n);
78. auto start = chrono::steady\_clock::now();
79. msort(in, 1, n);
80. auto end = chrono::steady\_clock::now();
81. cout << "\nHasil: ";
82. **for** (**int** i = 0; i < n; i++)
83. {
84. cout << in[i] << " ";
85. }
87. auto elapsed = chrono::duration\_cast<chrono::nanoseconds>(end - start);
89. cout << endl;
90. cout << "\nElapsed time in nanoseconds : " << elapsed.count() << " ns" << endl;
92. **return** 0;
93. }



* Running Time

Sudah dicoba menggunakan steady clock, high resolution, system clock. Tetap 0 ns. Mungkin error atau memang terlalu cepat





T(n) = cn + cn-c +cn-2c + ..... + 2c +cn

= c((n-1)(n-2)/2) + cn

= c((n2-3n+2)/2) + cn

= c(n2/2)-(3n/2)+1 + cn

= O(n2)

T(n) = cn + cn-c +cn-2c + ..... + 2c +cn

= c((n-1)(n-2)/2) + cn

= c((n2-3n+2)/2) + cn

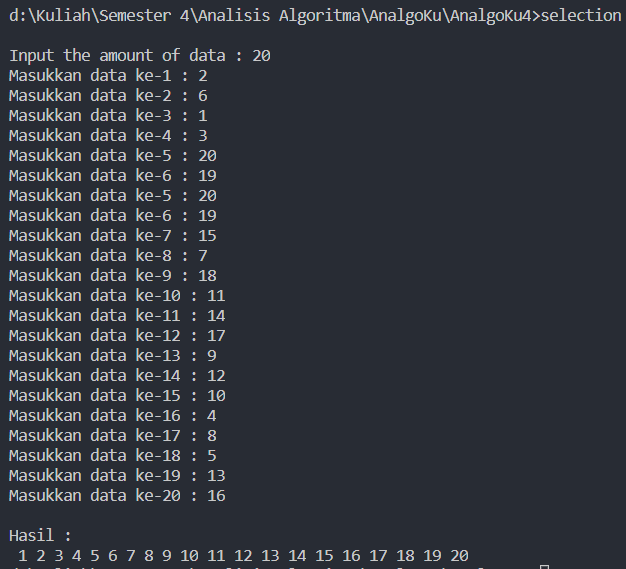
= c(n2/2)-(3n/2)+1 + cn

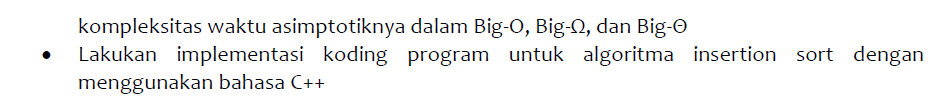
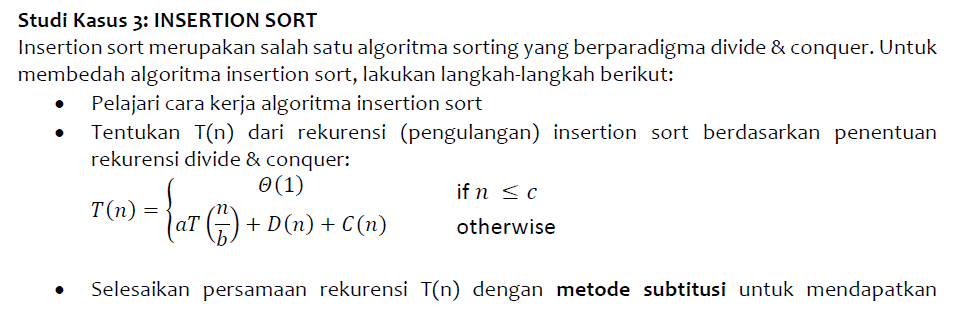
= Ω (n2)

T(n) = cn2

= Θ(n2)

* Program C++
* /\*
* Nama        : Muhammad Iqbal Alif Fadilla
* Kelas       : B
* NPM         : 140810180020
* Deskripsi   : Selection Sort
* \*/
* #include <iostream>
* **using** **namespace** std;
* **int** arr[100], arr2[100];
* **int** n;
* **void** swap(**int** a, **int** b)
* {
* **int** t;
* t = arr[b];
* arr[b] = arr[a];
* arr[a] = t;
* }
* **void** SelectionSort()
* {
* **int** pos, i, j;
* **for** (i = 1; i <= n - 1; i++)
* {
* pos = i;
* **for** (j = i + 1; j <= n; j++)
* {
* **if** (arr[j] < arr[pos])
* pos = j;
* }
* **if** (pos != i)
* swap(pos, i);
* }
* }
* **int** main()
* {
* cout << "Masukkan banyak data : ";
* cin >> n;
* cout << "\n";
* **for** (**int** i = 1; i <= n; i++)
* {
* cout << "Masukkan data ke-" << i << " : ";
* cin >> arr[i];
* arr2[i] = arr[i];
* }
* SelectionSort();
* cout << "\nHasil : " << endl;
* **for** (**int** i = 1; i <= n; i++)
* {
* cout << " " << arr[i];
* }
* **return** 0;
* }





T(n) = cn + cn-c +cn-2c + ..... + 2c +cn <= 2cn2 + cn2

= c((n-1)(n-2)/2) + cn<= 2cn2 + cn2

= c((n2-3n+2)/2) + cn<= 2cn2 + cn2

= c(n2/2)-c(3n/2)+c+cn <= 2cn2 + cn2

= O(n2)

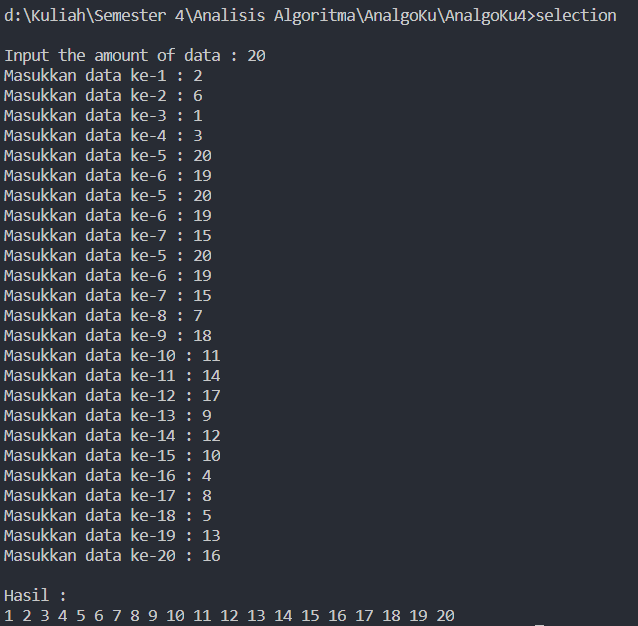
T(n) = cn <= cn

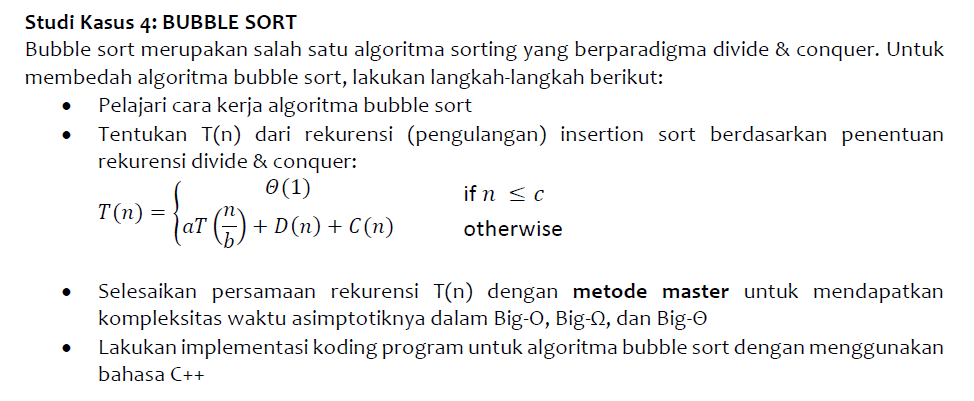
= Ω (n)

T(n) = (cn + cn2)/n

= Θ(n)

* Program C++
* /\*
* Nama        : Muhammad Iqbal Alif Fadilla
* Kelas       : B
* NPM         : 140810180020
* Deskripsi   : Insertion search
* \*/
* #include <iostream>
* **using** **namespace** std;
* **int** arr[100], arr2[100], n;
* **void** InsertionSort()
* {
* **int** temp, i, j;
* **for** (i = 1; i <= n; i++)
* {
* temp = arr[i];
* j = i - 1;
* **while** (arr[j] > temp && j >= 0)
* {
* arr[j + 1] = arr[j];
* j--;
* }
* arr[j + 1] = temp;
* }
* }
* **int** main()
* {
* cout << "Masukkan banyak data : ";
* cin >> n;
* cout << endl;
* **for** (**int** i = 1; i <= n; i++)
* {
* cout << "Masukkan data ke-" << i << " : ";
* cin >> arr[i];
* arr2[i] = arr[i];
* }
* InsertionSort();
* cout << "\nHasil : " << endl;
* **for** (**int** i = 1; i <= n; i++)
* {
* cout << arr[i] << " ";
* }
* **return** 0;
* }





T(n) = cn + cn-c +cn-2c + ..... + 2c +c <= 2cn2 + cn2

= c((n-1)(n-2)/2) + c<= 2cn2 + cn2

= c((n2-3n+2)/2) + c<= 2cn2 + cn2

= c(n2/2)-c(3n/2)+2c <= 2cn2 + cn2

= O(n2)

T(n) = cn + cn-c +cn-2c + ..... + 2c +c <= 2cn2 + cn2

= c((n-1)(n-2)/2) + c<= 2cn2 + cn2

= c((n2-3n+2)/2) + c<= 2cn2 + cn2

= c(n2/2)-c(3n/2)+2c <= 2cn2 + cn2

= Ω (n2)

T(n) = cn2 + cn2

= Θ(n2)

* Program C++
* /\*
* Nama        : Muhammad Iqbal Alif Fadilla
* Kelas       : B
* NPM         : 140810180020
* Deskripsi   : Bubble Sort
* \*/
* #include <iostream>
* #include <conio.h>
* **using** **namespace** std;
* **int** main()
* {
* **int** arr[100], n, temp;
* cout << "Masukkan banyak data : ";
* cin >> n;
* cout << "\n";
* **for** (**int** i = 0; i < n; ++i)
* {
* cout << "Masukkan data ke-" << i + 1 << " : ";
* cin >> arr[i];
* }
* **for** (**int** i = 1; i < n; i++)
* {
* **for** (**int** j = 0; j < (n - 1); j++)
* {
* **if** (arr[j] > arr[j + 1])
* {
* temp = arr[j];
* arr[j] = arr[j + 1];
* arr[j + 1] = temp;
* }
* }
* }
* cout << "\nHasil : ";
* **for** (**int** i = 0; i < n; i++)
* {
* cout << " " << arr[i];
* }
* }

