**LAPORAN PRAKTIKUM 2**

**ANALISIS ALGORITMA**



**DISUSUN OLEH**

AHMAD IRFAN FADHOLI

140810180034

**PROGRAM STUDI TEKNIK INFORMATIKA**

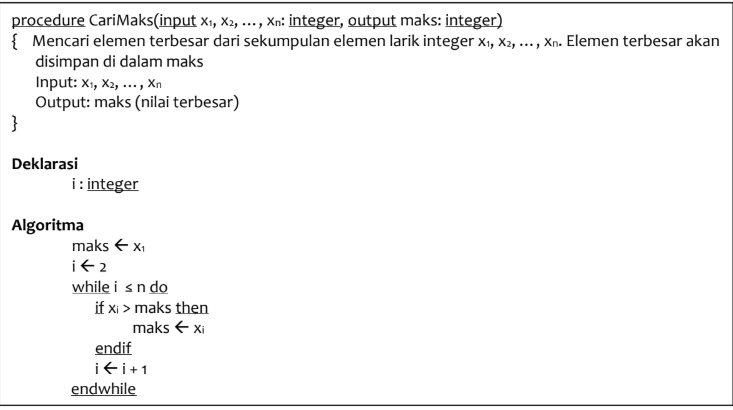
**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

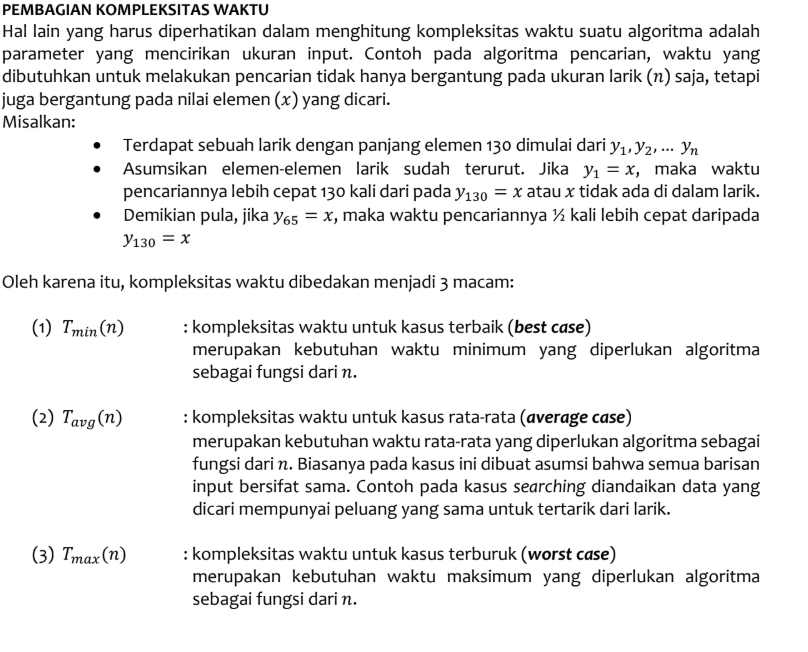
**UNIVERSITAS PADJADJARAN**

**2020**

**Studi Kasus 1: Pencarian Nilai Maksimal**

Buatlah programnya dan hitunglah kompleksitas waktu dari algoritma berikut:

Algoritma Pencarian Nilai Maksimal



1. /\*
2. Nama : Ahmad Irfan Fadholi
3. NPM : 14081080034
4. Tgl : 10 Maret 2020
5. Deskripsi : Studi Kasus No 1\*/
7. #include <iostream>
8. **using** **namespace** std;
10. **int** main(){
11. **int** x[5] = {500, 333, 231, 5, 1};
12. **int** n = **sizeof**(x)/**sizeof**(x[0]);
13. **int** maks = x[0];
14. **int** i = 2;
16. **while** (i <= n){
17. **if** (x[i] > maks)
18. maks = x[i];
19. i = i + 1;
20. }
21. cout << "Maximum = " << maks;
23. **return** 0;
24. }

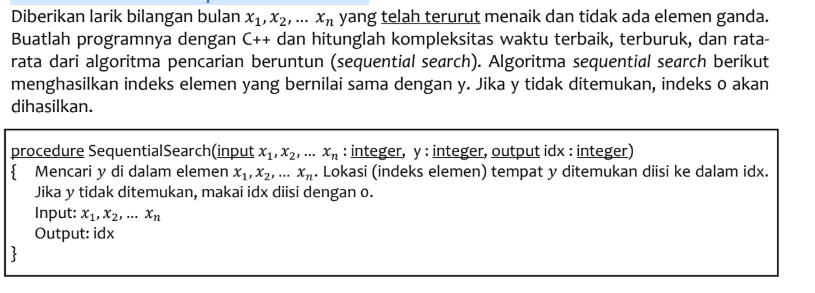
Operasi Assignment = 1 + 1 + (n-1) + (n-1) = 2n

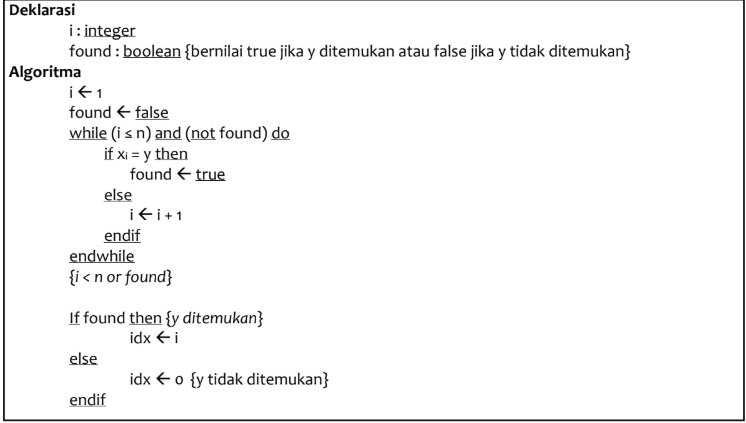
Operasi Perbandingan = n-1

Operasi Penjumlahan = n-1

Maka *Tmax(n)* = 4n-2

Studi Kasus 2: Sequential Search





1. /\*
2. Nama : Ahmad Irfan Fadholi
3. NPM : 14081080034
4. Tgl : 10 Maret 2020
5. Deskripsi : Studi Kasus No 2\*/
7. #include <iostream>
8. **using** **namespace** std;
10. **int** main()
11. {
12. **int** x[5] = {739, 323, 69, 1, 33};
13. **int** find = 739;
14. **int** n = **sizeof**(x) / **sizeof**(x[0]);
16. **int** index;
17. **int** i = 1;
18. **bool** found = **false**;
20. **while** (i <= n && not found){
21. **if** (x[i] == find)
22. found = **true**;
23. **else**
24. i = i + 1;
25. }
27. **if** (found == **true**){
28. index = i;
29. cout << "Ditemukan pada Index " << index;
30. }
32. **else**{
33. index = 0;
34. cout << "Not Found";
35. }
37. **return** 0;
38. }

Jawab :

1.Operasi Assignment = 4

2.Operasi Perbandingan = 2

*Tmin(n)* = 4 + 2 = 6

1.Operasi Assignment = 1 + 1 + n + 1 = 3 + n

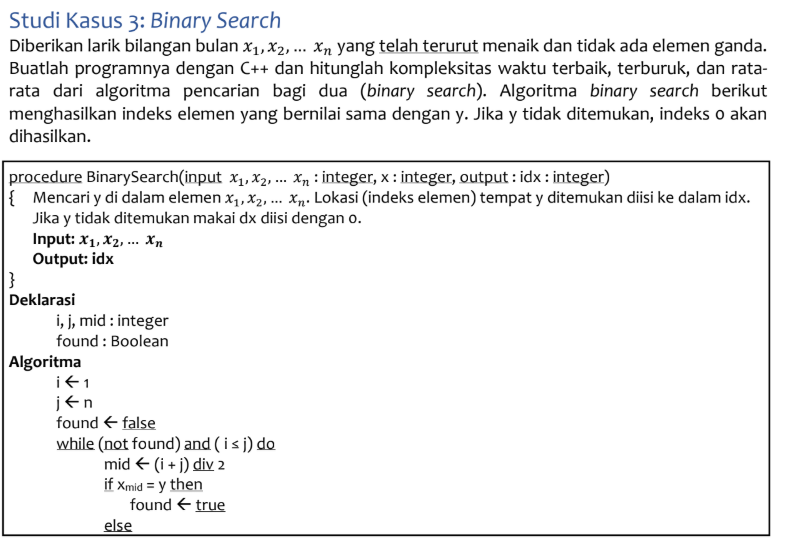
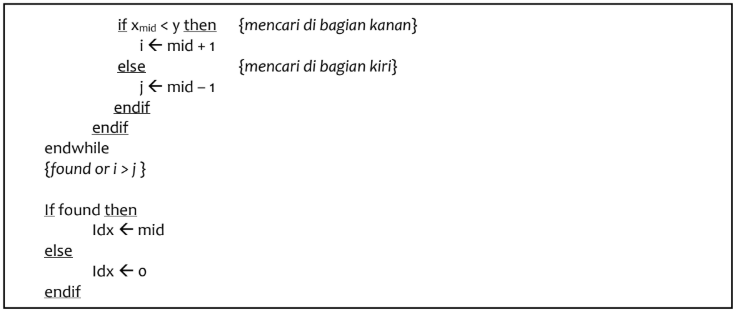
2.Operasi Perbandingan = n + 1

3.Operasi Penjumlahan = n

*Tmax(n)* = 3+n+n+1+n=3n+4

(*Tmin(n)* + *Tmax(n)* )/ 2 = (6+4+3n) / 2 = (10+3n) / 2

*Tavg(n)* = (10+3n) / 2

1. /\*
2. Nama : Ahmad Irfan Fadholi
3. NPM : 14081080034
4. Tgl : 10 Maret 2020
5. Deskripsi : Studi Kasus No 3 - Binary Search\*/
7. #include <iostream>
8. **using** **namespace** std;
10. **int** main(){
11. **int** x[5] = {14, 31, 59, 72, 98};
12. **int** find = 59;
13. **int** mid;
14. **int** n = **sizeof**(x) / **sizeof**(x[0]);
15. **int** index;
16. **int** i = 1;
17. **int** j = n;
18. **bool** found = **false**;
20. **while** (not found && i <= j){
21. mid = (i + j) / 2;
22. **if** (x[mid] == find)
23. found = **true**;
24. **else** **if** (x[mid] < find)
25. i = mid + 1;
26. **else**
27. j = mid - 1;
28. }
30. **if** (found){
31. index = mid;
32. cout << "Index ditemukan pada :  " << index;
33. }
34. **else**{
35. index = 0;
36. cout << "Not Found";
37. }
39. **return** 0;
40. }

Jawab :

1.Operasi Assignment = 6

2.Operasi Perbandingan = 2

*Tmin(n)* = 6 + 2 = 8

Panjang array akan berubah pada setiap iterasi:

•Iterasi 1 = n

•Iterasi 2 = n/2

•Iterasi 3 = n/22

•Iterasi x = n/2k-1 ~ n/2k (-1 diabaikan karena kecil dibanding n/2k)

Panjang array menjadi 1.

Maka,

n/2k= 1

n= 2k

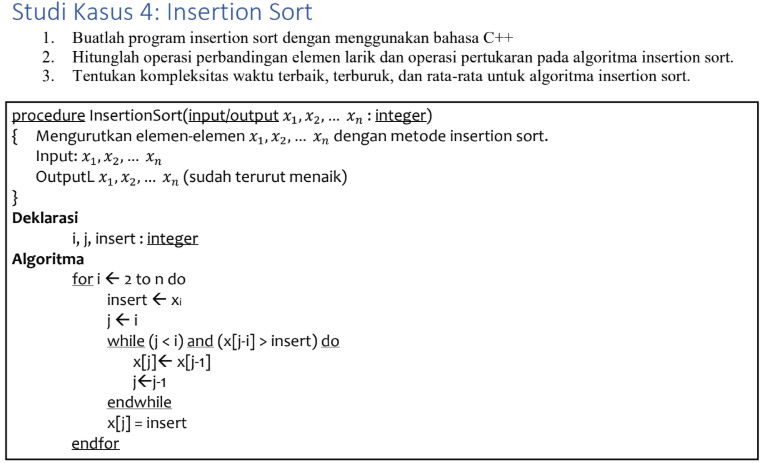
log 2(n) = log 2(2k) = k log 2(2)

k= log 2(n)

*Tmax(n)* = (log 2(n))

(*Tmin(n)* + *Tmax(n)* )/ 2 = (1 + log 2(n)) / 2

*Tavg(n)* = (1 + log 2(n)) / 2



1. /\*
2. Nama : Ahmad Irfan Fadholi
3. NPM : 14081080034
4. Tgl : 10 Maret 2020
5. Deskripsi : Studi Kasus No 4 - Insertion Sort\*/
7. #include <iostream>
8. **using** **namespace** std;
10. **int** x[100], y[100], n;
12. **void** InsertionSort(){
13. **int** temp, i, j;
14. **for** (i = 1; i <= n; i++){
15. temp = x[i];
16. j = i - 1;
17. **while** (x[j] > temp && j >= 0){
18. x[j + 1] = x[j];
19. j--;
20. }
21. x[j + 1] = temp;
22. }
23. }
25. **int** main(){
26. cout << "Insertion Sort\nMasukkan Banyak data : ";
27. cin >> n;
28. **for** (**int** i = 1; i <= n; i++){
29. cout << "Masukkan data ke-" << i << " : ";
30. cin >> x[i];
31. y[i] = x[i];
32. }
33. InsertionSort();
34. cout << "\nHasil Output : ";
36. **for** (**int** i = 1; i <= n; i++){
37. cout << x[i] << " ";
38. }
39. }

Jawab :

1.Operasi Assignment: 2(n-1) + (n-1) = 3n-3

2.Operasi Perbandingan: 2\*((n-1) + (n-1)) = 2\*(2n-2) = 4n-4

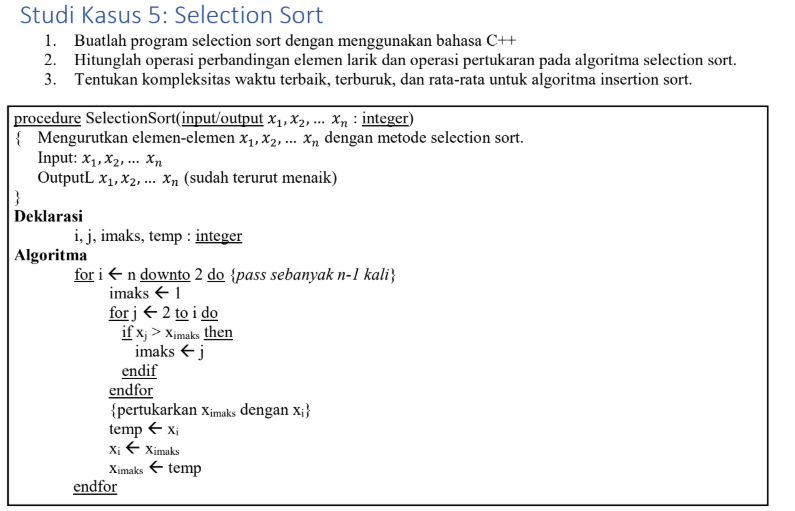
3.Operasi Pertukaran: (n-1) \* n = n2-n

*Tmin(n)* = 3n-3 + 4n-4 + 1 = 7n – 6

*Tmax(n)* = 3n-3 + 4n-4 + n2-n = n2+6n-6

(*Tmin(n)*+ *Tmax(n)* )/ 2 = (7n–6 + n2+6n-6) / 2

*Tavg(n)* = (n2 + 13n - 12) / 2



1. /\*
2. Nama : Ahmad Irfan Fadholi
3. NPM : 14081080034
4. Tgl : 10 Maret 2020
5. Deskripsi : Studi Kasus No 5 - Selection Sort\*/
7. #include <iostream>
8. **using** **namespace** std;
10. **int** x[100], y[100],n;
12. **void** swap(**int** a, **int** b){
13. **int** temp;
14. temp = x[b];
15. x[b] = x[a];
16. x[a] = temp;
17. }
19. **void** SelectionSort(){
20. **int** pos, i, j;
21. **for** (i = 1; i <= n - 1; i++){
22. pos = i;
23. **for** (j = i + 1; j <= n; j++){
24. **if** (x[j] < x[pos])
25. pos = j;
26. }
27. **if** (pos != i)
28. swap(pos, i);
29. }
30. }
32. **int** main(){
33. cout << "Selection Sort \nMasukkan banyak data : ";
34. cin >> n;
36. **for** (**int** i = 1; i <= n; i++){
37. cout << "Masukkan data ke-" << i << " : ";
38. cin >> x[i];
39. y[i] = x[i];
40. }
41. SelectionSort();
42. cout << "\nHasil Output : ";
43. **for** (**int** i = 1; i <= n; i++){
44. cout << " " << x[i];
45. }
46. **return** 0;
47. }

Jawab :

1.Operasi Perbandingan = ½(n^2-n)

2.Operasi Pertukaran = n-1

*Tmin(n)* = (4n-4) + ½(n2-n) +1 ~ n2

*Tmax(n)* = ½(n2-n) + (n-1) ~ n2

(*Tmin(n)* + *Tmax(n)* )/ 2 = (n2+ n2) / 2

*Tavg(n)* = n2