**LAPORAN PRAKTIKUM**

**ANALISIS ALGORITMA**

**Modul Praktikum 2**

****

**Oleh:**

**Benhard D. H. Tampubolon (140810170042)**

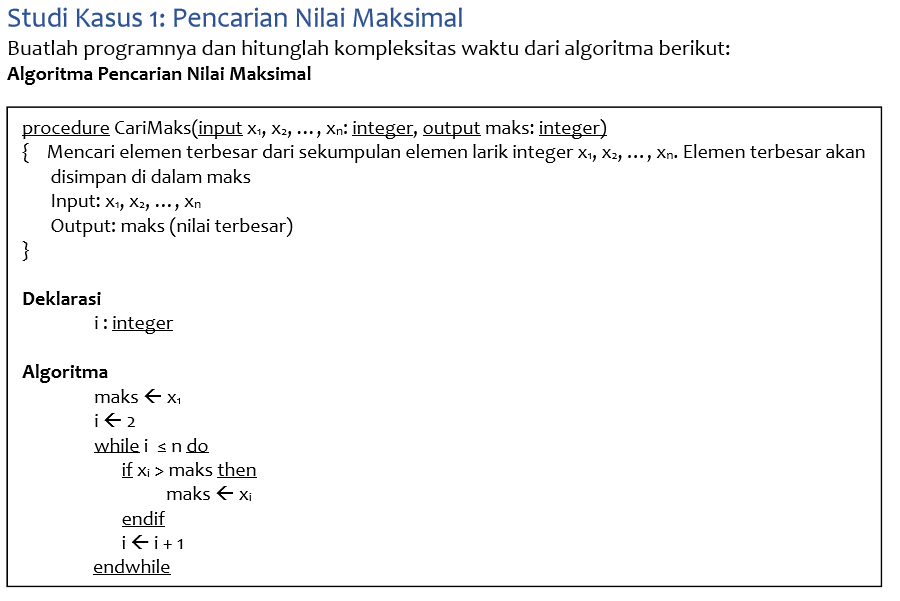
**TAHUN AJARAN 2018/2019**

**PROGRAM STUDI TEKNIK INFORMATIKA**

**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM**

**UNIVERSITAS PADJADJARAN**

Analisis Algoritma



Analisis Algoritma:

1. Operasi pengisian nilai (assignment)

maks ← x1, 1 kali

i ← 2, 1 kali

maks ←x1 n-1 kali

i ← i+1, n-1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + *n* - 1+ *n* - 1 = 2*n*

1. Operasi penjumlahan

k+1, n-1 kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n* - 1

1. Operasi perbandingan

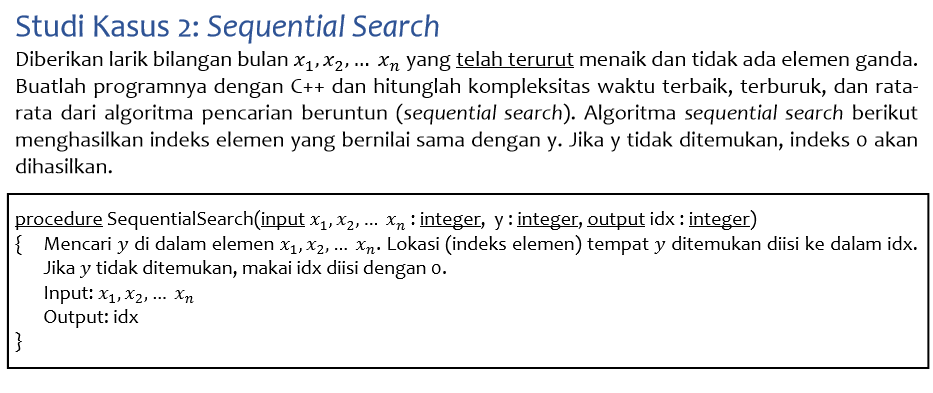
xi>maks, n-1 kali

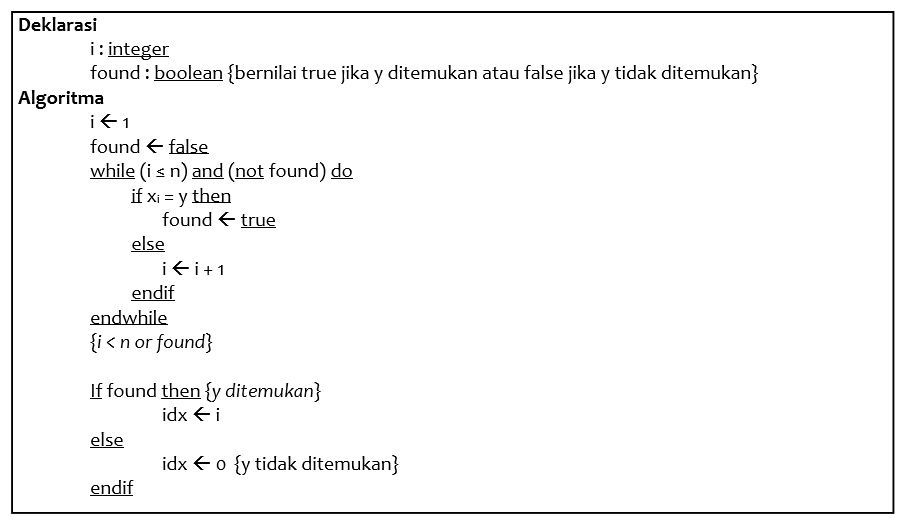
Jumlah seluruh operasi penjumlahan adalah

*t*3 = *n* - 1

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 2*n* + *n* – 1 + *n* – 1 = 4*n* - 2





Analisis Algoritma:

Waktu terbaik:

1. Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← true, 1 kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1= 3

1. Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

xi = y 1 kali

found = true 1 kali

*t*3 = 1+ 1 = 2

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 3 + 2 = 5

Waktu rata-rata:

1. Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← false, 1 kali

found ← true, 1 kali

i ← i+1, (n/2) kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1+ (*n/2)* + 1 = 4 + *n/2*

1. Operasi penjumlahan

i+1, (n/2) kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n/2*

1. Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

xi = y (n+1/2) kali

found = true 1 kali

*t*3 = *n/2* + 1 + 1/2 = (n+3/2)

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 4 + *n/2* + (n/2) + (n+3/2) = (3*n* + 11)/2

Waktu terburuk:

1. Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← false, 1 kali

found ← true, 1 kali

i ← i+1, n-1 kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1+ *n* – 1 + 1 = 3+ *n*

(ii)Operasi penjumlahan

i+1, n-1 kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n*

(iii)Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

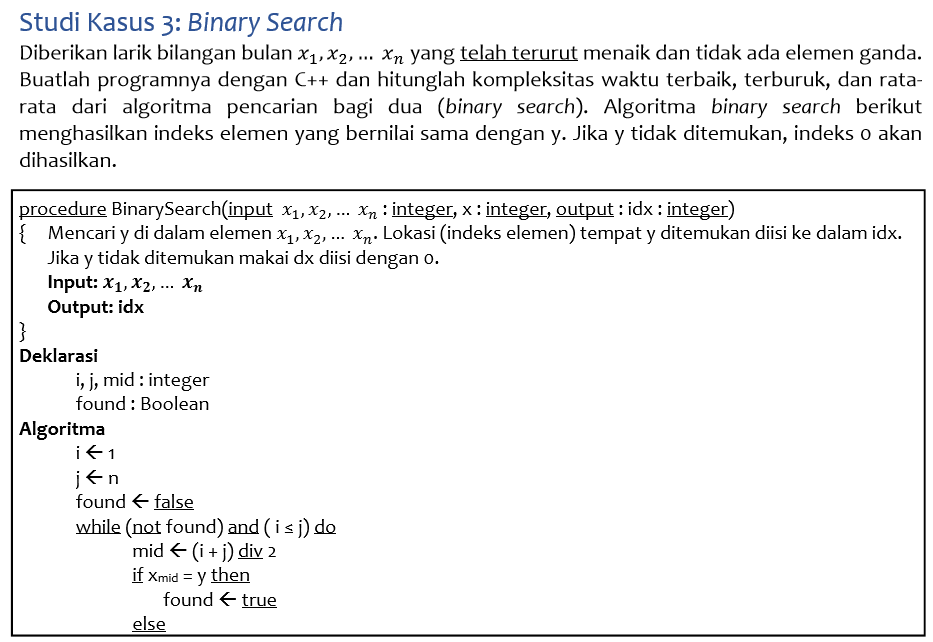
xi = y n kali

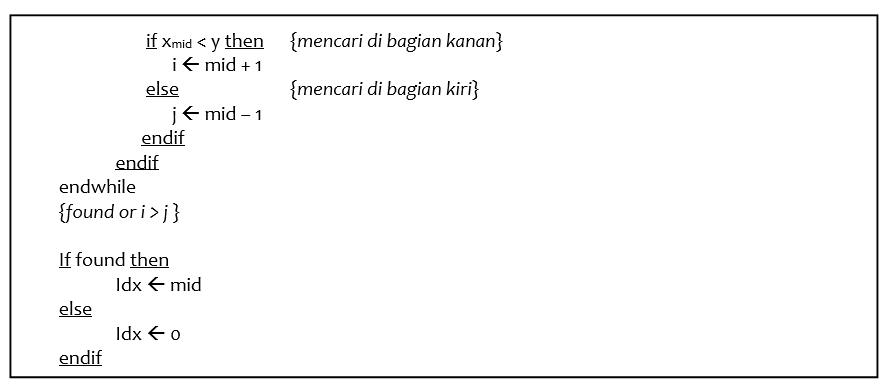
found = true 1 kali

*t*3 = *n* + 1

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 4 + *n-1* + *n-1* + *n* + 1 = 3*n* + 3





Analisis Algoritma:

Waktu terbaik:

(i)Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← true, 1 kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1= 3

(ii)Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

xi = y 1 kali

found = true 1 kali

*t*3 = 1+ 1 = 2

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 3 + 2 = 5

Waktu rata-rata:

(i)Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← false, 1 kali

found ← true, 1 kali

i ← i+1, (n/2) kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1+ (*n/2)* + 1 = 4 + *n/2*

(ii)Operasi penjumlahan

i+1, (n/2) kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n/2*

(iii)Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

xi = y (n+1/2) kali

found = true 1 kali

*t*3 = *n/2* + 1 + 1/2 = (n+3/2)

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 4 + *n/2* + (n/2) + (n+3/2) = (3*n* + 11)/2

Waktu terburuk:

(i)Operasi pengisian nilai (assignment)

i ← 1, 1 kali

found ← false, 1 kali

found ← true, 1 kali

i ← i+1, n-1 kali

idx ← i, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + 1+ *n* – 1 + 1 = 3+ *n*

(ii)Operasi penjumlahan

i+1, n-1 kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n*

(iii)Operasi perbandingan

Jumlah seluruh operasi pembagian adalah

xi = y n kali

found = true 1 kali

*t*3 = *n* + 1

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 4 + *n-1* + *n-1* + *n* + 1 = 3*n* + 3

Analisis Algoritma:

Waktu terburuk

1. Operasi pengisian nilai (assignment)

maks ← x1, 1 kali

i ← 2, 1 kali

maks ←x1 n-1 kali

i ← i+1, n-1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + *n* - 1+ *n* - 1 = 2*n*

1. Operasi penjumlahan

k+1, n-1 kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n* - 1

1. Operasi pembagian

Jumlah seluruh operasi pembagian adalah

Jumlah/n 1 kali

*t*3 = *n* - 1

1. Operasi perbandingan

xi>maks, n-1 kali

Jumlah seluruh operasi penjumlahan adalah

*t*4 = *n* - 1

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 2*n* + *n* – 1 + *n* – 1 = 4*n* - 2

Analisis Algoritma:

1. Operasi pengisian nilai (assignment)

jumlah ← 0, 1 kali

k ← 1, 1 kali

jumlah ←jumlah + ak n kali

k ← k+1, n kali

r ← jumlah/n, 1 kali

Jumlah seluruh operasi pengisian nilai (assignment) adalah

*t*1 = 1 + 1 + *n* + *n* + 1 = 3 + 2*n*

1. Operasi penjumlahan

Jumlah + ak, n kali

k+1, n kali

Jumlah seluruh operasi penjumlahan adalah

*t*2 = *n* + *n* = 2*n*

1. Operasi pembagian

Jumlah seluruh operasi pembagian adalah

Jumlah/n 1 kali

Dengan demikian, kompleksitas waktu algoritma dihitung berdasarkan jumlah operasi aritmatika dan operasi pengisian nilai adalah:

*T*(*n*)= *t*1 + *t*2 + *t*3 = 3*n* + 2*n* + 2*n* + 1 = 4*n* + 4