Mochamad Faizin Ahsan

140810160032

Kelas B

1. Studi Kasus Mencari Maks

int CariMaks(int x[], int n){

int i =2;

int maks = x[1];

while(i<=n){

if (x[i]>maks)

{

maks = x[i];

}

i++;

}

return maks;

}

1. Operasi assignment
   1. Best Case

i 🡨 2 1 kali

maks 🡨 x­­1 1 kali

i 🡨 i+1 n-1 kali

maks 🡨 x1 0 kali

T1 = 1+n

* 1. Worst Case

i 🡨 2 1 kali

maks 🡨 x­­1 1 kali

i 🡨 i+1 n-1 kali

maks 🡨 xi n-1 kali

T1 = 2n

1. Operasi Penjumlahan

i+1 n-1 kali

T2 = n-1

1. Operasi Perbandingan

xi>maks n kali

T3 = n

Maka dari itu waktu kompleksitias untuk algoritma ini dihitung berdasarkan aritmatika dan operasi pengisian nilai:

T(min) = T1+T2+T3 = 1+n + n-1 + n = 3n

T(max) = T1+T2+T3 = 2n + n-1 + n = 4n-1

1. Studi Kasus Sequential Search

int SequentialSearch(int n,int y,int x[]){

int i = 0,index=0;

bool found = false;

while((i<n)&&(found == false)){

if (x[i] == y)

{

found = true;

}else{

i++;

}

}

if (found == true)

{

index = i;

}else{

index = 0;

}

return index;

}

1. Best Case

X0 = y;

Tmin(n) = 1

1. Worst Case

Bila an = y atau y tidak ditemukan

Tmax(n) = n

1. Average Case
   1. Assignment

i🡨1 1 kali

found 🡨 false 1 kali

found 🡨 true n-i kali

i 🡨 i+1 n-i kali

idx 🡨 I 1 kali

T1 = 2n-2i-3

* 1. Penjumlahan

i+1 n-i

T2 = n-i

* 1. Perbandinga

Xi == y n-i kali

If found 1 kali

T3 = n-i-3

Maka

TAvg = 2n-2i-3 + n-i + n-i-3 = 4n-4i-6

Jika i diabaikan maka Tavg = 4n-6 = 2n-3

1. Binary

void binarySearch(){

const int Ar[10] = {1,2,3,4,5,6,7,8,9,10}; // untuk proses ascending

int tar;

cout<<"masukan data yang dicari : ";

cin>>tar;

int awal=0, akhir=10, tengah;

while (awal <= akhir)

{ tengah = (awal + akhir)/2;

if (tar > Ar[tengah] ) // descending ubah tanda > menjadi <

{ awal = tengah + 1; }

else if (tar < Ar[tengah]) // descending ubah tanda < menjadi >

{akhir= tengah - 1;}

else {awal = akhir +1;

}

}

if (tar == Ar[tengah])

{cout<<" Data ditemukan, Ke- "<<tengah+1<<endl;

}

else {

cout<<"target tidak ditemukan "<<endl;

}

}

Best Case

Asignment

i🡨 1 1 kali

j🡨 n 1 kali

found 🡨 false 1 kali

mid 🡨 (i+j)/2 1 kali

found 🡨 true 1 kali

idx 🡨 mid 1 kali

Penjumlahan

I+j 1 kali

Pembagian

I+j / 2 1 kali

Perbandingan

If Xmid = y 1 kali

If found 1 kali

Tmin(n) = 10

Worst Case

Asignment

i🡨 1 1 kali

j🡨 n 1 kali

found 🡨 false 1 kali

mid 🡨 (i+j)/2 n kali

found 🡨 true n kali

i 🡨 mid+1 n kali

j🡨mid -1 n kali

idx 🡨 mid n kali

Penjumlahan

I+j n kali

Mid +1 n kali

Mid -1 n kali

Pembagian

I+j / 2 n kali

Perbandingan

If Xmid = y n kali

If xmid < y n kali

If xmid > y n kali

If found 1 kali

If notfound 1 kali

Tmax(n) = 5+12n