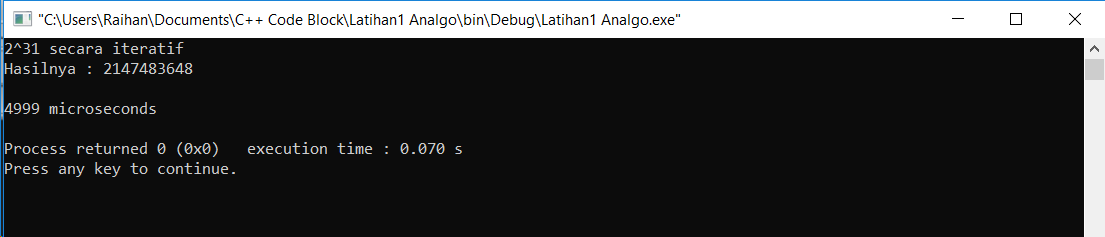
Analisis Run Time

Latihan 1

## 2n dengan iteratif

File .cpp untuk bahasa C++

|  |
| --- |
| #include <iostream>  #include <chrono>  using namespace std;  using namespace std::chrono;  int main()  {  // Begin  high\_resolution\_clock::time\_point t1 = high\_resolution\_clock::now();  int n = 31; // Manual input  cout<<"2^" <<n <<" secara iteratif \n";  long long int result = 2;  for(int i=n; i>1; i--){  result \*= 2;  }  cout<<"Hasilnya : " <<result <<endl<<endl;  // End  high\_resolution\_clock::time\_point t2 = high\_resolution\_clock::now();  auto duration = duration\_cast<microseconds>( t2 - t1 ).count();  cout<<duration <<" microseconds" <<endl;  return 0;  } |

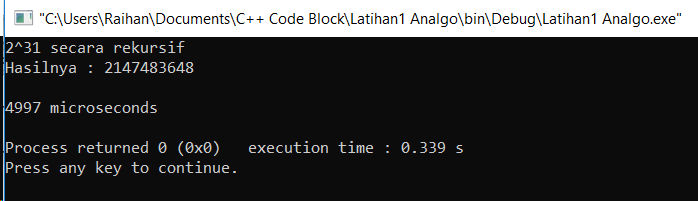


Runtime : 0,004999 detik

## 2n dengan rekursif

File .cpp untuk bahasa C++

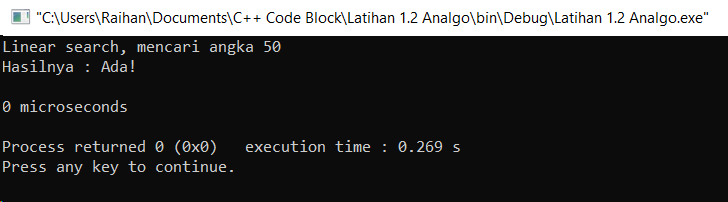
|  |
| --- |
| #include <iostream>  #include <chrono>  using namespace std;  using namespace std::chrono;  long long int rekursif(int n){  if (n == 1)  return 2;  return 2 \* rekursif(n - 1);  }  int main()  {  // Begin  high\_resolution\_clock::time\_point t1 = high\_resolution\_clock::now();  int n = 31; // Manual input  cout<<"2^" <<n <<" secara iteratif \n";  cout<<"Hasilnya : " <<rekursif(n) <<endl<<endl;  // End  high\_resolution\_clock::time\_point t2 = high\_resolution\_clock::now();  auto duration = duration\_cast<microseconds>( t2 - t1 ).count();  cout<<duration <<" microseconds" <<endl;  return 0;  } |



## Pencarian Linear

File .cpp untuk bahasa C++

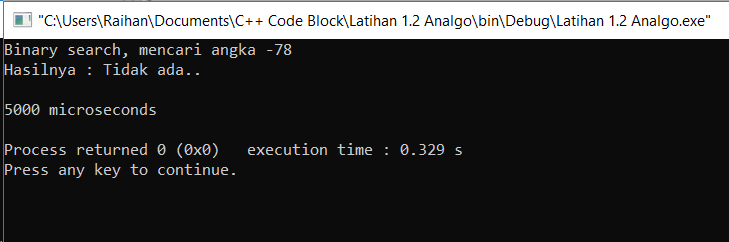
|  |
| --- |
| #include <iostream>  #include <chrono>  using namespace std;  using namespace std::chrono;  int main()  {  // Sorted ASCENDING Array  int n = 200; // Jumlah data  int arr[n];  for(int i=0; i<n; i++){  arr[i] = i + 1;  }  // Begin  high\_resolution\_clock::time\_point t1 = high\_resolution\_clock::now();  int size = sizeof(arr) / sizeof(arr[0]);  int searchKey = 30;  cout<<"Linear search, mencari angka " <<searchKey <<endl;  bool flag = false;  for(int i=0; i<size; i++){  if (n == arr[i])  flag = true;  }  cout<<"Hasilnya : "  <<(flag == true ? "Ada!" : "Tidak ada..") <<endl<<endl;  // End  high\_resolution\_clock::time\_point t2 = high\_resolution\_clock::now();  auto duration = duration\_cast<microseconds>( t2 - t1 ).count();  cout<<duration <<" microseconds" <<endl;  return 0;  } |



## Pencarian Biner

File .cpp untuk bahasa C++

|  |
| --- |
| #include <iostream>  #include <chrono>  using namespace std;  using namespace std::chrono;  int main()  {  // Sorted ASCENDING Array  int n = 100; // Jumlah data  int arr[n];  for(int i=0; i<n; i++){  arr[i] = i + 1;  }  // Begin  high\_resolution\_clock::time\_point t1 = high\_resolution\_clock::now();  int size = sizeof(arr) / sizeof(arr[0]);  int searchKey = -78;  cout<<"Binary search, mencari angka " <<searchKey <<endl;  bool flag = false;  int left = 0,  right = size - 1;  while (left <= right) {  int mid = left + (right - left) / 2;  if (arr[mid] == searchKey)  flag = true;  if (arr[mid] < searchKey)  left = mid + 1;  else  right = mid - 1;  }  cout<<"Hasilnya : "  <<(flag == true ? "Ada!" : "Tidak ada..") <<endl<<endl;  // End  high\_resolution\_clock::time\_point t2 = high\_resolution\_clock::now();  auto duration = duration\_cast<microseconds>( t2 - t1 ).count();  cout<<duration <<" microseconds" <<endl;  return 0;  } |



## Analisis

|  |  |  |  |
| --- | --- | --- | --- |
| Percobaan ke- | n | Iteratif (ms) | Rekusif (ms) |
| 1 | 31 | 0 | 0 |
| 2 | 31 | 3008 | 1569 |
| 3 | 31 | 3510 | 4993 |
| 4 | 31 | 4902 | 5001 |
| 5 | 40 | 4999 | 0 |
| 6 | 40 | 5009 | 5019 |
| 7 | 40 | 5001 | 4999 |
| 8 | 50 | 0 | 5016 |
| 9 | 50 | 5002 | 4511 |
| 10 | 50 | 5004 | 0 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Percobaan ke- | Jumlah data | Pencarian data ke- | Linear search (ms) | Binary search (ms) |
| 1 | 200 | 30 | 3049 | 3008 |
| 2 | 200 | 100 | 1860 | 2999 |
| 3 | 200 | 178 | 2076 | 3015 |
| 4 | 200 | 200 | 1833 | 1024 |
| 5 | 1000 | 100 | 3007 | 11417 |
| 6 | 1000 | 500 | 8033 | 3033 |
| 7 | 1000 | 1000 | 3010 | 3008 |
| 8 | 3000 | 300 | 1532 | 6713 |
| 9 | 3000 | 1500 | 6058 | 6540 |
| 10 | 3000 | 3000 | 13029 | 3011 |

Keterangan :

ms = microsecond  
warna biru pada tabel melambangkan nilai yang lebih cepat (ms)

Running time program **tidak akan selalu sama** tiap kali program di-*run,* bahkan untuk skenario yang sama sekalipun*.* Kita tidak tahu apa yang sedang CPU kerjakan selain menjalankan program ini, terkadang sangat cepat hingga kalkulasi runtimenya bernilai 0 terkadang juga lambat karena performa kecepatannya harus dialihkan ke tugas lain terlebih dahulu.

Menurut hasil tes yang dilaksanakan, Algotitma rekursif lebih cepat daripada iteratif dan Binary search lebih cepat daripada Linear search. Keduanya memiliki kelebihan dan kelemahan tersendiri, sesuai skenario yang diberikan terntunya.

## Spesifikasi

Program yang digunakan adalah Code::Blocks 17.12

Compiler GNU GCC  
Bahasa C++11

Spesifikasi komputer :

