1. Langkah-Langkah Percobaan
2. Langkah-langkah Percobaan Sequential Search
3. Buatlah packages baru pada Netbeans dengan nama TestSearching
4. Buat class Searching , kemudian deklarasikan variabel berikut ini:

|  |
| --- |
| public class Searching {  public int[] data;  public int jumData;  } |

1. Buatlah konstruktor dengan parameter Data[] dan jmlData kemudian Isi konstruktor tersebut dengan kode berikut!

|  |
| --- |
| public Searching(int[] Data, int jmlData){  this.jumData = jmlData;  data = new int[jmlData];  for (int i = 0; i < jumData; i++) {  data[i] = Data[i];  }  } |

1. Buatlah method FindSeqSearch bertipe integer dengan parameter cari bertipe integer. Kemudian Deklarasikan isi method Find dengan algoritma pencarian data menggunakan teknik sequential searching.

|  |
| --- |
| public int FindSeqSearch(int cari){  int posisi = -1;  for (int j = 0; j < jumData; j++) {  if (data[j] == cari) {  posisi = j;  break;  }  }  return posisi;  } |

1. Buatlah method TampilData bertipe void dan Deklarasikan isi dari method TampilData

|  |
| --- |
| public void TampilData(){  for (int i = 0; i < jumData; i++) {  System.out.print(data[i]+ " ");  }  System.out.println("");  } |

1. Buatlah class baru dengan nama SearchingMain tambahkan method main dan kemudian buat array dengan nama data kemudian isi array tersebut!

|  |
| --- |
| public class SearchingMain {  public static void main(String[] args) {  int data[] = {10,30,40,50,60,70,80,90};  } |

1. . Buatlah objek baru dengan nama pencarian yang merupakan instansiasi dari class Searching, kemudian isi parameternya!

|  |
| --- |
| Searching pencarian = new Searching(data, 8); |

1. Lakukan pemanggilan method TampilData pada class Searching.

|  |
| --- |
| System.out.println("Isi Array : ");  pencarian.TampilData(); |

1. Deklarasikan variabel cari bertipe integer dan isi nilainya dengan nilai yang ingin dicari

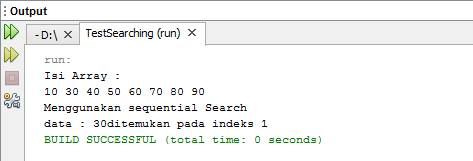
|  |
| --- |
| int cari = 30; |

1. Deklarasikan variabel posisi bertipe integer yang memanggil method FindSeqSearch yang terdapat pada class Searching

|  |
| --- |
| System.out.println("Menggunakan sequential Search");  int posisi = pencarian.FindSeqSearch(cari); |

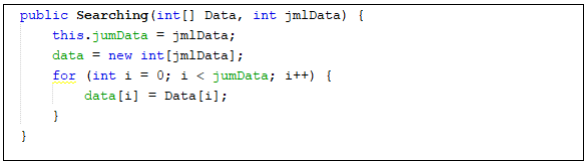
1. Lakukan pengecekan untuk kondisi data yang dicari.

|  |
| --- |
| if (posisi != -1) {  System.out.println("data : " + cari + "ditemukan pada indeks " + posisi);  }  else{  System.out.println("data " + cari + "tidak ditemukan");  } |

1. Jalankan dan Amati Hasilnya

**PERTANYAAN**

1. Jelaskan fungsi kode program berikut ini!



Jawab :

Kode program tersebut adalah method dengan nama Searching dengan 2 parameter yaitu Data bertipe array integer dan jmlData betipe integer. Kemudian variabel jumdata bernilai sama dengan jmlData. Kemudian variabel data yang bertipe array integer nilainya akan sama dengan jumlah isi array dengan jmlData. Perulangan berfungsi untuk mengidentifikasi nilai array. Dimana data[i] akan berulanga sebanyak 8 kali yang dimulai dari 0, kemudian Data[i] akan terisi nilai yang sama dengan nilai array

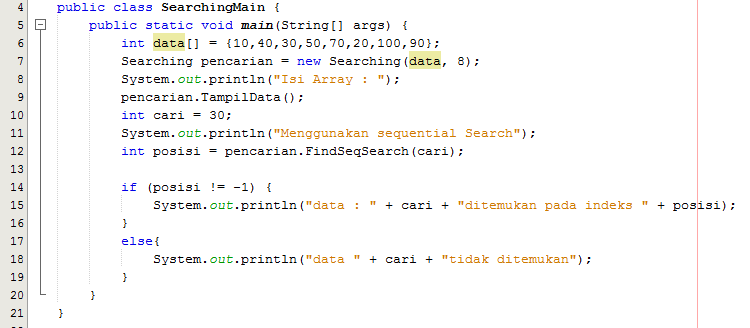
1. Jelaskan fungsi break pada kode program dibawah ini!

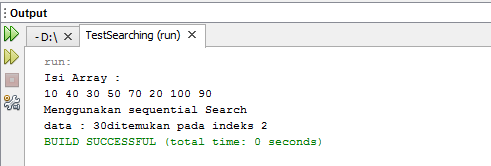
Jawab :

Fungsi break digunakan untuk menghentikan pencarian apabila data yang dicari telah ditemukan

1. Ubahlah array int data[] ={10,40,30,50,70,20,100,90}. Apakah program masih dapat berjalan? Apakah hasil yang dikeluarkan benar? Mengapa demikian!

Jawab :

Program masih bisa berjalan serta hasil yang dikeluarkan adalah bear. Hal ini dikarenakan cara yang digunakan adalah ssequential search sehigga pencarian akan dilakukan secara berurutan dari depan ke paling belakang. Dalam program data yang dicari adalah 30, maka program akan melakukakan pencarian dari elemen pertama hingga ditemukan data yang bernilai 30. Sampai pada akhirnya ditemukan elemen 30 pada indeks ke dua



1. Langkah-Langkah Percobaan Binary Search
2. Pada percobaan 1.4.1 (sequential search) tambahkan method FindBinarySearch bertipe integer pada class Searching. Kemudian Deklarasikan isi method FindBinarySearch dengan algoritma pencarian data menggunakan teknik binary searching.

|  |
| --- |
| public int FindBinarySearch(int cari, int left, int right){  int mid;  if (right >= left) {  mid = (left + right) / 2;  if (cari == data[mid]) {  return (mid);  }  else if(data[mid] > cari){  return FindBinarySearch(cari,left, mid-1);  }  else{  return FindBinarySearch(cari, mid+1, right);  }  }  return -1;  } |

1. Tambahkan method Tampilpoisisi pada class Searching.

|  |
| --- |
| public void Tampilposisi(int x, int pos){  if (pos != -1) {  System.out.println("data " + x + " ditemukan pada indeks ke " + pos);  }  else{  System.out.println("data " + x + " tidak ditemukan");  }  } |

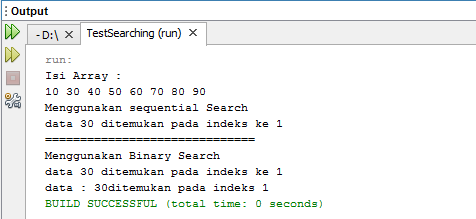
1. Kembalikan nilai int data menjadi berikut ini Pada class SearchingMain

|  |
| --- |
| int data[] = {10,30,40,50,60,70,80,90}; |

1. Hapus kondisi yang ada di fungsi main kemudian panggil method Tampilposisi dan memanggil method FindBinarySearch terdapat pada class Searching

|  |
| --- |
| pencarian.Tampilposisi(cari, posisi);  System.out.println("==============================");  System.out.println("Menggunakan Binary Search");  posisi = pencarian.FindBinarySearch(cari, 0, data.length -1);  pencarian.Tampilposisi(cari, posisi); |

1. Jalankan dan amati hasilnya.



PERTANYAAN

1. Tunjukkan pada kode program yang mana proses divide dijalankan!

Jawab :

|  |
| --- |
| mid = (left + right) / 2; |

1. Tunjukkan pada kode program yang mana proses conquer dijalankan!

Jawab :

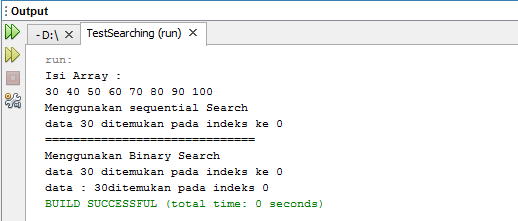
|  |
| --- |
| else if(data[mid] > cari){  return FindBinarySearch(cari,left, mid-1);  }  else{  return FindBinarySearch(cari, mid+1, right); |

1. Jika array int data [] ={100,90,80,70,60,50,40,30} dan elemen yang dicari adalah 30. Bagaimana hasil dari binary search? Apakah sesuai? Jika tidak sesuai maka ubahlah kode program binary seach agar hasilnya sesuai

Jawab :

Data tidak sesuai. Karena metode pencarian yang digunakan adalah Binary Search maka yang harus dilakukan adalah dengan mengurutkan data di dalam array terlebih dahulu.

|  |
| --- |
| int data[] = {30,40,50,60,70,80,90,100}; |



1. Ubahlah array int data[] ={10,40,30,50,70,20,100,90}. Apakah program masih dapat berjalan? Mengapa demikian!

Jawab :

Untuk pencarian dengan Binary Search pencarian untuk nilai 30 tidak dapat ditemukan karena nilai di dalam array belum terurut dari nilai terkecil ke nilai tertinggi (ascending)

1. Modifikasilah program diatas yang mana data array dapat di inputkan atau bersifat dinamis!

Jawab :

1. Langkah-langkah Percobaan Merge Sort
2. Buatlah Package baru pada NetBeans dengan nama MergeSortTest
3. Tambahkan class MergeSort pada package tersebut

|  |
| --- |
| package MergeSortTest;  public class MergeSorting {  public void mergesort(int[] data){  sort(data, 0, data.length -1);  }    public void merge(int[] data, int left, int middle, int right){  int[] temp = new int[data.length];  for (int i = left; i <= right; i++) {  temp[i] = data[i];  }  int a = left;  int b = middle + 1;  int c = left;    while (a <= middle && b <= right) {  if (temp[a] <= temp[b]) {  data[c] = temp[a];  a++;  } else {  data[c] = temp[b];  b++;  }  c++;  }  int s = middle - a;  for (int i = 0; i <= s; i++) {  data[c + i] = temp[a + i];  }  }  private void sort(int data[], int left, int right){  if (left < right) {  int middle = (left + right) / 2;  sort(data, left, middle);  sort(data, middle + 1, right);  merge(data, left, middle, right);  }  }    public void printArray(int arr[]){  int n = arr.length;  for (int i = 0; i < n; i++) {  System.out.println(arr[i]+ " ");  }  System.out.println();  }    } |

1. Buatlah method sort

|  |
| --- |
| package MergeSortTest{  public class SortMain {  public static void main(String[] args) {  int data[] = {10,40,30,50,70,20,100,90};  System.out.println("sorting dengan merge sort");  MergeSorting mSort = new MergeSorting();  System.out.println("dataa awal");  mSort.printArray(data);  mSort.mergesort(data);  System.out.println("Setalah diurutkan");  mSort.printArray(data);  }  } |

