**LAPORAN**



**INSERTION SORT**

Nama : Diaz Adha Asri Prakoso

NIM : 0102518007

Program Studi : Teknik Informatika

**UNIVERSITAS AL AZHAR INDONESIA**

**JAKARTA**

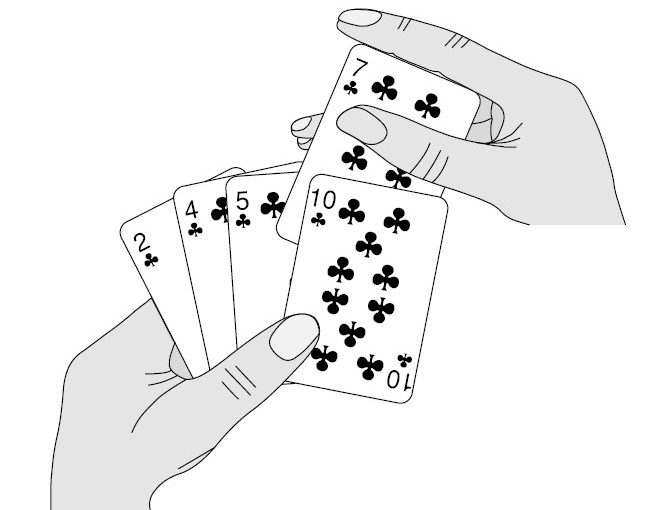
**2019**

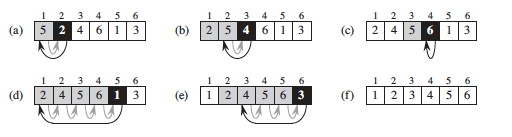
1. **Definisi Insertion Sort**

Insertion sort adalah sebuah algoritma pengurutan yang membandingkan dua elemen data pertama, yang kemudian di urutkan, lalu mengecek elemen data berikutnya satu persatu dan membandingkannya dengan elemen data yang telah diurutkan.

1. **Cara Kerja**

Anggap saja insertion sort seperti permainan kartu. Saat kita memegang beberapa kartu, kita mulai membandingkan satu persatu kartu mulai dari ujung paling kiri. Satu per satu kartu di bandingkan dan nominal terkecil akan diletakan di sebelah kiri, begitu terus hingga didapat urutan yang benar dari kiri ke kanan.

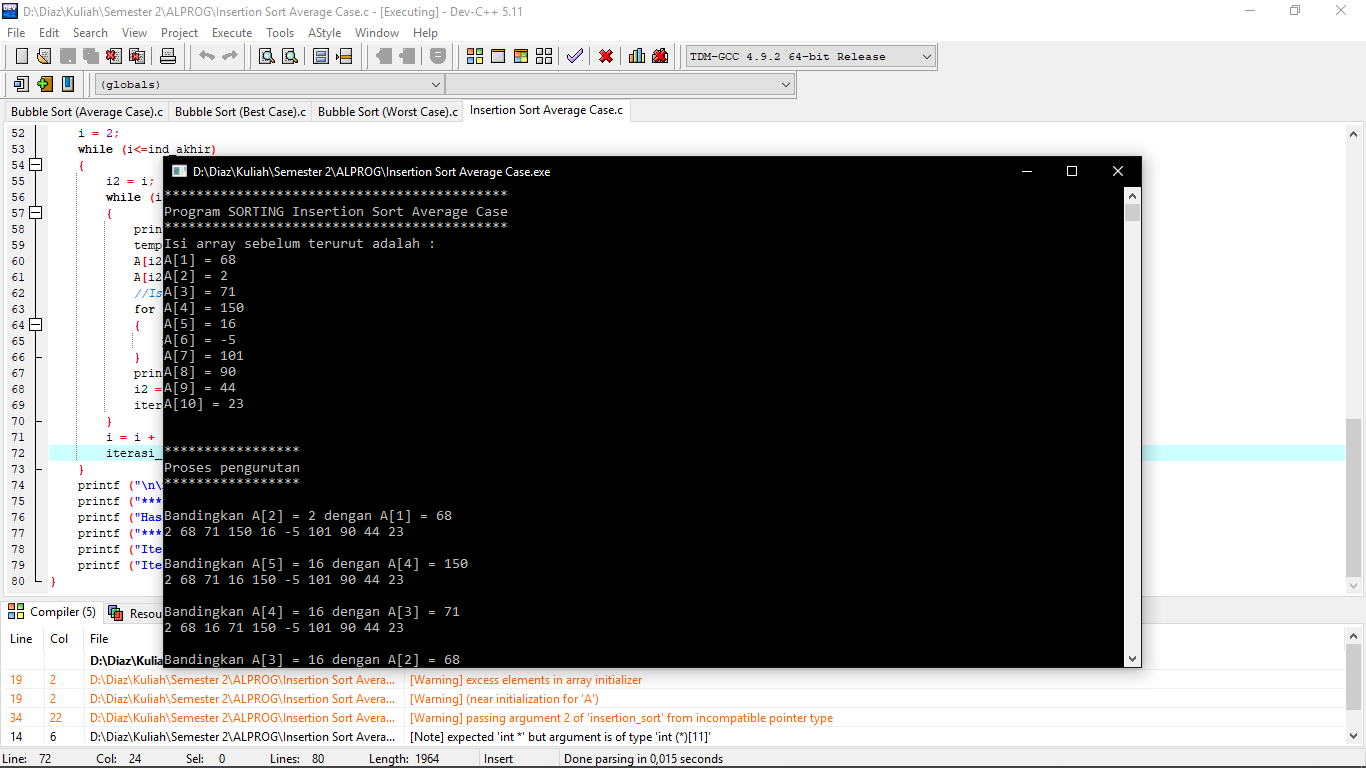


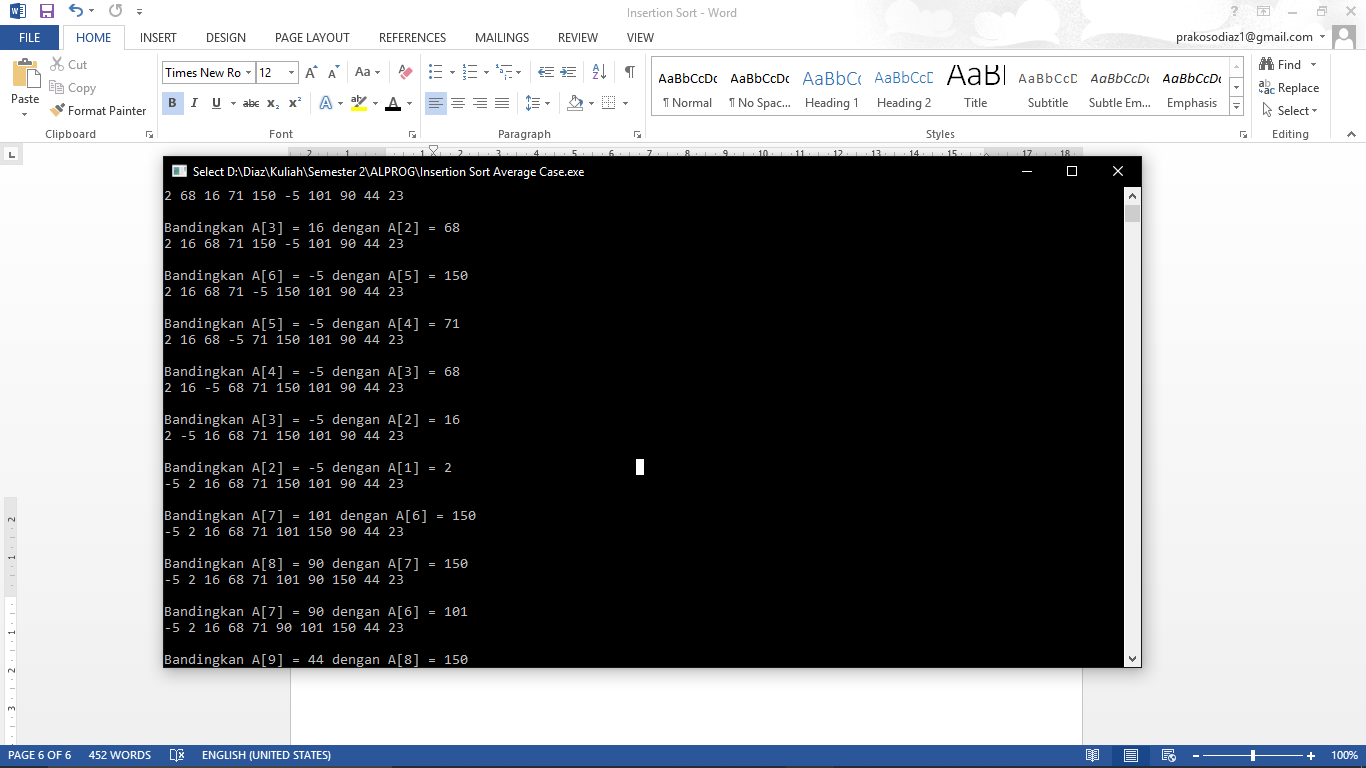


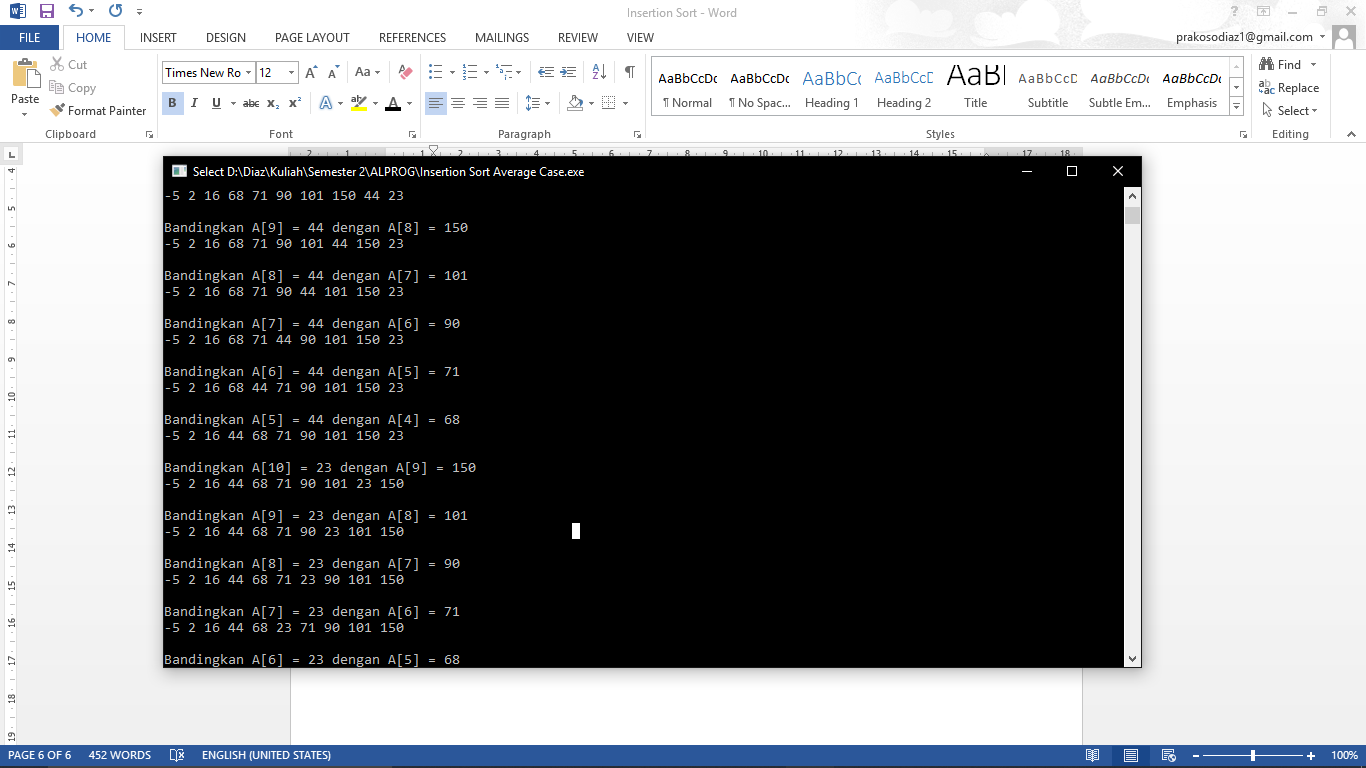
1. **Pseudo Code**

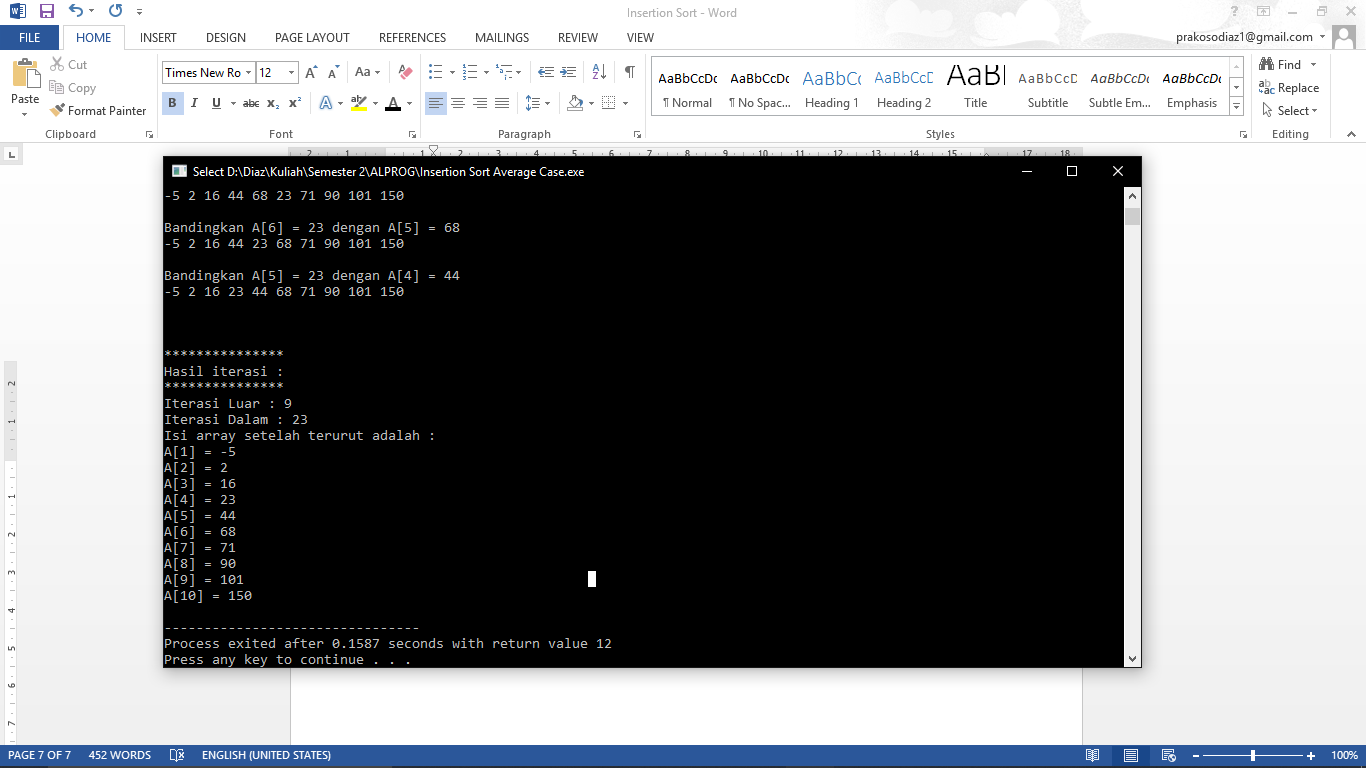
|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43  44  45  46  47  48  49  50  51  52  53  54  55  56  57  58  59  60  61  62  63  64  65  66  67  68  69  70  71  72  73  74  75  76  77 | #include <stdio.h>  #include <stdlib.h>  /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*  Deskripsi : Mengurutkan array dari elemen ke-1 hingga akhir menggunakan metode insertion sort  IS : Array telah berisi nilai  FS : Array telah terurut  Dibuat oleh :  Nama : Diaz Adha Asri Prakoso  Tanggal : 25 Mei 2019  \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/  void insertion\_sort (int a, int A[100]);  void main (void)  //KAMUS  { int x;  int A[11] = {999, 68, 2, 71, 150, 16, -5, 101, 90, 44, 23, -9999};  //ALGORITMA  printf ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");  printf ("Program SORTING Insertion Sort Average Case \n");  printf ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");  printf ("Isi array sebelum terurut adalah : ");  printf ("\n");  for (x=1;x<=10;x++)  {  printf ("A[%d] = %d\n", x, A[x]);  }  printf ("\n\n\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");  printf ("Proses pengurutan \n");  printf ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n\n");  insertion\_sort (10, &A);    printf ("Isi array setelah terurut adalah : ");  printf ("\n");  for (x=1;x<=10;x++)  {  printf ("A[%d] = %d\n", x, A[x]);  }  }  void insertion\_sort (int a, int A[100])  {//KAMUS  int ind\_awal, ind\_akhir, i, i2, temp, iterasi\_luar, iterasi\_dalam, x;    ind\_awal = 1;  ind\_akhir = a;  iterasi\_dalam = 0;  iterasi\_luar = 0;  i = 2;  while (i<=ind\_akhir)  {  i2 = i;  while (i2>ind\_awal && A[i2]<A[i2-1])  {  printf ("Bandingkan A[%d] = %d dengan A[%d] = %d\n", i2, A[i2], i2-1, A[i2-1]);  temp = A[i2-1];  A[i2-1] = A[i2];  A[i2] = temp;  //Isi array setelah ditukar  for (x=1;x<=10;x++)  {  printf ("%d ", A[x]);  }  printf ("\n\n");  i2 = i2 - 1;  iterasi\_dalam++;  }  i = i + 1;  iterasi\_luar++;  }  printf ("\n\n");  printf ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");  printf ("Hasil iterasi : \n");  printf ("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* \n");  printf ("Iterasi Luar : %d\n", iterasi\_luar);  printf ("Iterasi Dalam : %d\n", iterasi\_dalam);  } |

**# Hasil Run :**









**Referensi :** <http://student.blog.dinus.ac.id/setiyono/>