**Laporan Dasar-Dasar Pemrograman Tugas 4**

Nama: Jestine Vallendra Dwi Putra

NIM : 11220910000026

Compiler: gcc.exe (GCC) 11.2.0

**PEMBAHASAN:**

1. Soal 1:

program ini akan mencari nilai input pada array

Kode No 1:

*// program ini akan mencari nilai input pada array*

*#include <stdio.h>*

*// DEKLARASI KONSTANTA*

*const int ARR[] = {1, 2, 3, 4, 5};*

*// deklarasi variable*

*int input\_int;*

*// deklarasi prosedur*

*void cari\_elemen(int target, int arr\_src[], int arr\_size);*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta masukan dari user*

*printf("Masukkan angka yang ingin dicari dalam array: ");*

*scanf("%i", &input\_int);*

*// memanggil prosedur*

*cari\_elemen(input\_int, ARR, sizeof(ARR) / sizeof(ARR[0]));*

*return 0;*

*}*

*// definisi prosedur*

*void cari\_elemen(int target, int arr\_src[], int arr\_size)*

*{*

*int idx\_pos = -1; // posisi item yang sama, dengan nilai default*

*int i; // index yang digunakan pada loop*

*// loop seisi array untuk mencari nilai yang sama*

*for (i = 0; i < arr\_size; i++)*

*{*

*// jika nilai sama maka perbarui posisi item*

*if (arr\_src[i] == target)*

*{*

*idx\_pos = i;*

*}*

*}*

*if (idx\_pos == -1)*

*{*

*printf("Posisi %i dalam array adalah 0\n", target);*

*}*

*else*

*{*

*printf("Posisi %i dalam array adalah %i\n", target, idx\_pos);*

*}*

*}*

Pseudocode No 1:

// program ini akan mencari nilai input pada array

// DEKLARASI KONSTANTA

const var ARR: int[] = {1, 2, 3, 4, 5}

// deklarasi variable

var input\_int: int

// deklarasi prosedur

fn cari\_elemen(int target, int arr\_src[], int arr\_size): void

// deklarasi algoritma

fn main(): int do

// meminta masukan dari user

input\_int <- read("Masukkan angka yang ingin dicari dalam array: ")

// memanggil prosedur

cari\_elemen(input\_int, arr, sizeof(arr) / sizeof(arr[0]))

return 0;

end fn

// definisi prosedur

fn cari\_elemen(int target, int arr\_src[], int arr\_size): void do

var idx\_pos: int <- -1 // posisi item yang sama, dengan nilai default

var i: int // index yang digunakan pada loop

// loop seisi array untuk mencari nilai yang sama

for i <- 0 to arr\_size i++ do

// jika nilai sama maka perbarui posisi item

if arr\_src[i] == target do

idx\_pos = i

end if

end for

if (idx\_pos == -1) do

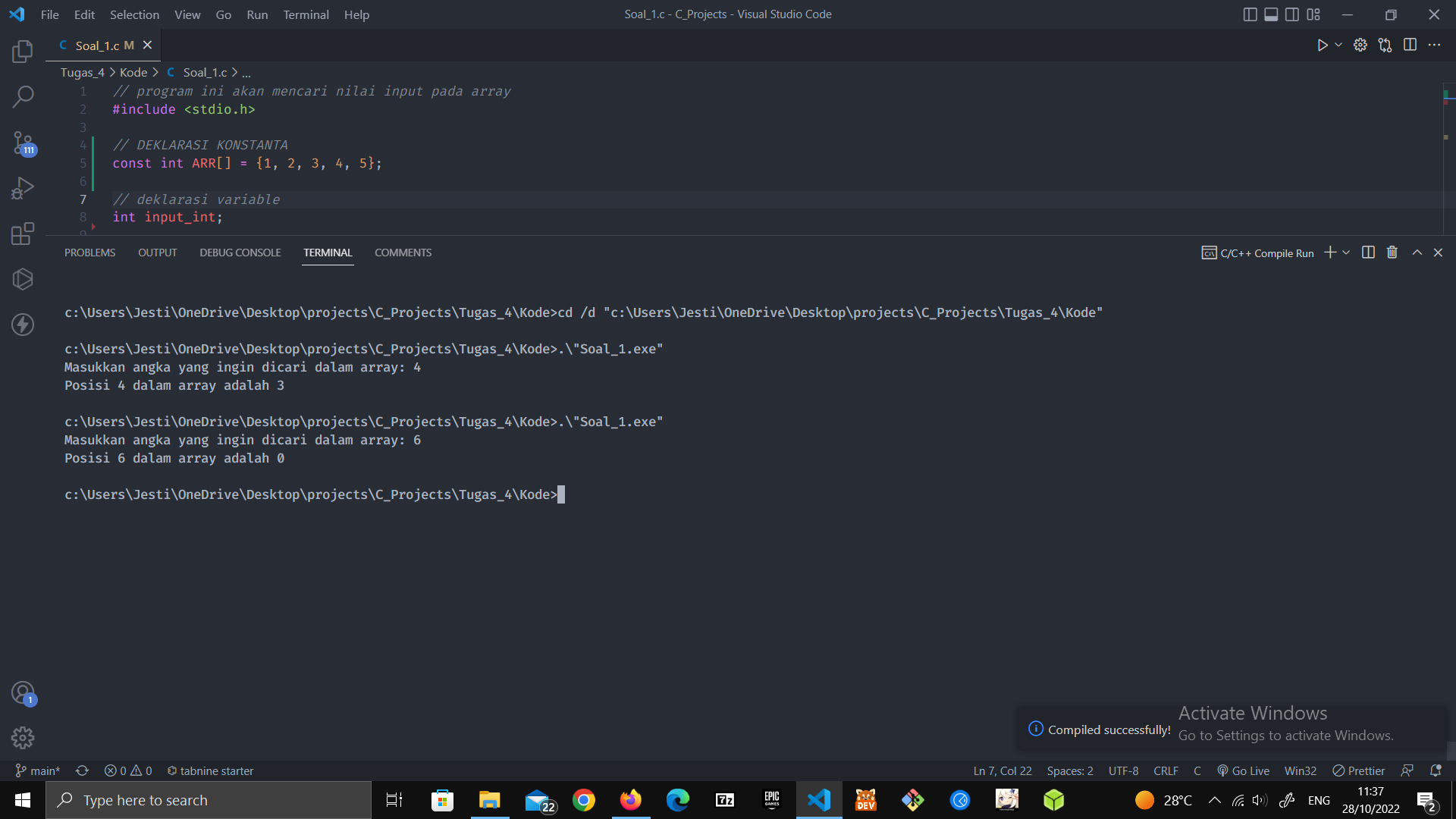
write("Posisi %i dalam array adalah 0\n", target)

else do

write("Posisi %i dalam array adalah %i\n", target, idx\_pos)

end if

end fn



Screenshot Program No 1:

1. Soal 2:

Program untuk menentukan range nilai di dalam larik

Kode No 2:

*// Program untuk menentukan range nilai di dalam larik*

*#include <stdio.h>*

*#include <stdlib.h>*

*// deklarasi variable*

*int arr\_len, i, s;*

*int terbesar, terkecil, range;*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta input dari user*

*printf("Masukan panjang array: ");*

*scanf("%i", &arr\_len);*

*// membuat array*

*int custom\_arr[arr\_len];*

*// memasukkan nilai pada setiap index dalam array*

*for (i = 0; i < arr\_len; i++)*

*{*

*printf("nilai item pada index %i: ", i);*

*scanf("%i", &custom\_arr[i]);*

*}*

*// mengisi nilai default untuk nilai terbesar dan terkecil*

*terbesar = custom\_arr[0];*

*terkecil = custom\_arr[0];*

*// meloop seisi array untuk mencari nilai terbesar dan terkecil*

*for (i = 0; i < arr\_len; i++)*

*{*

*if (custom\_arr[i] > terbesar)*

*{*

*terbesar = custom\_arr[i];*

*}*

*if (custom\_arr[i] < terkecil)*

*{*

*terkecil = custom\_arr[i];*

*}*

*}*

*// menghitung range dan menampilkan hasilnya*

*range = terbesar - terkecil;*

*printf("\nrange nilai di dalam array adalah %i", range);*

*return 0;*

*}*

Pseudocode No 2:

*// Program untuk menentukan range nilai di dalam larik*

*// deklarasi variable*

*var (arr\_len, i, s): int*

*var (terbesar, terkecil, range): int*

*// deklarasi algoritma*

*// meminta input dari user*

*arr\_len <- read("Masukan panjang array:")*

*// membuat array*

*var custom\_arr: int[arr\_len]*

*// memasukkan nilai pada setiap index dalam array*

*for i <- 0 to arr\_len do*

*custom\_arr[i] <- read("nilai item pada index %i : ", i)*

*end for*

*// mengisi nilai default untuk nilai terbesar dan terkecil*

*terbesar <- custom\_arr[0]*

*terkecil <- custom\_arr[0]*

*// meloop seisi array untuk mencari nilai terbesar dan terkecil*

*for i <- 0 i < arr\_len do*

*if custom\_arr[i] > terbesar do*

*terbesar <- custom\_arr[i]*

*end if*

*if custom\_arr[i] < terkecil do*

*terkecil <- custom\_arr[i]*

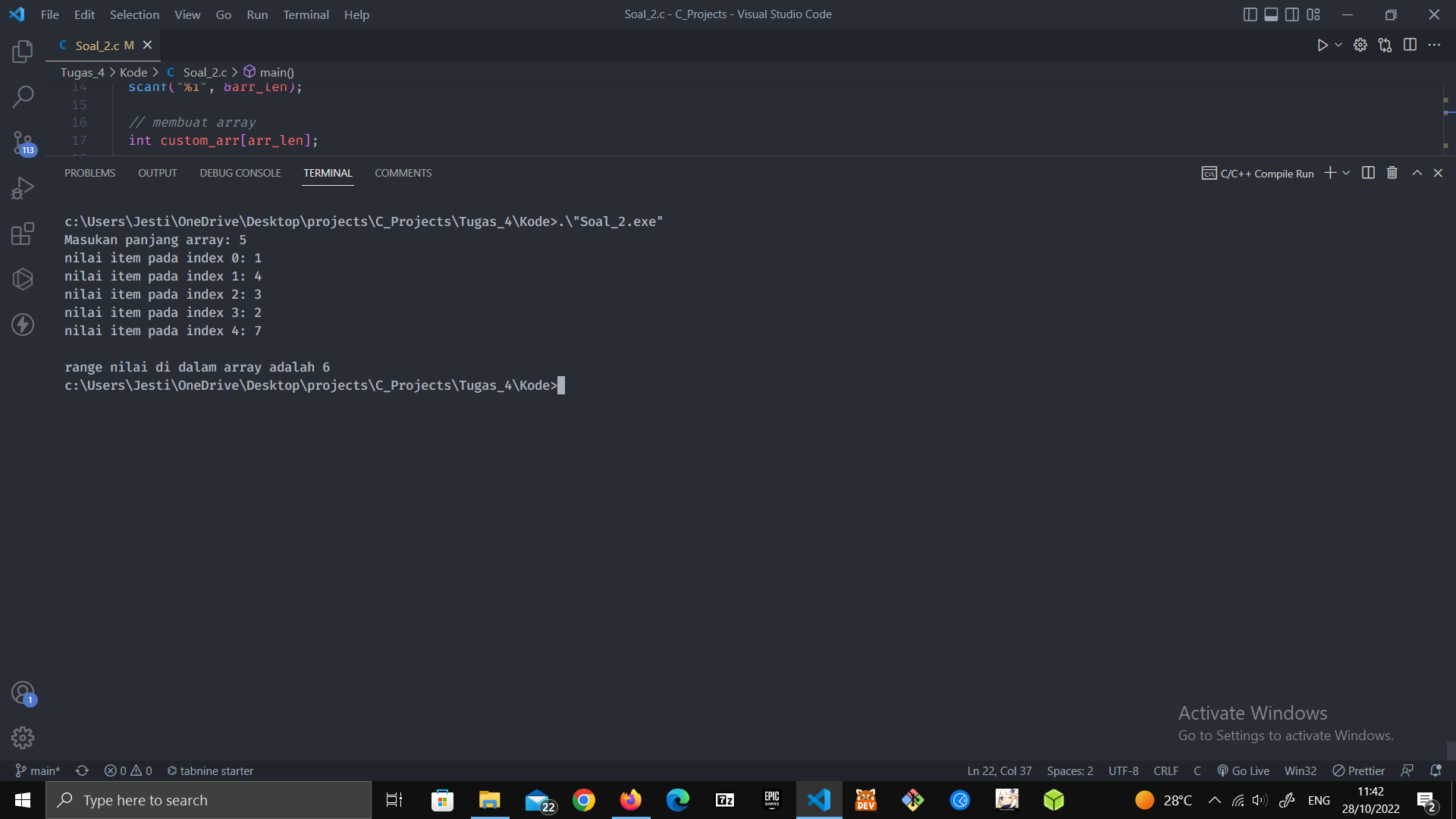
*end if*

*end for*

*// menghitung range dan menampilkan hasilnya*

*range <- terbesar - terkecil*

*write("range nilai di dalam array adalah %i :", range)*



Screenshot Program No 2:

1. Soal 3:

program untuk mencari nilai terbesar dan nilai terbesar kedua

Kode No 3:

*// program untuk mencari nilai terbesar dan nilai terbesar kedua*

*#include <stdio.h>*

*// deklarasi variable*

*int i;*

*int arr\_len, biggest, sec\_biggest;*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta masukan dari user*

*printf("Masukan panjang array: ");*

*scanf("%i", &arr\_len);*

*// deklarasi array dengan panjang custom*

*int custom\_arr[arr\_len];*

*// memasukkan nilai pada setiap index dalam array*

*for (i = 0; i < arr\_len; i++)*

*{*

*printf("nilai item pada index %i: ", i);*

*scanf("%i", &custom\_arr[i]);*

*}*

*// nilai default untuk nilai terbesar*

*biggest = custom\_arr[0];*

*// nilai default untuk nilai terbesar kedua*

*sec\_biggest = custom\_arr[0];*

*// loop seisi array untuk mencari nilai terbesar*

*for (i = 0; i < arr\_len; i++)*

*{*

*if (custom\_arr[i] > biggest)*

*{*

*biggest = custom\_arr[i];*

*}*

*}*

*// loop seisi array untuk mencari nilai terbesar kedua*

*for (i = 0; i < arr\_len; i++)*

*{*

*if (custom\_arr[i] > sec\_biggest && custom\_arr[i] < biggest)*

*{*

*sec\_biggest = custom\_arr[i];*

*}*

*}*

*// menampilkan hasil pencarian nilai terbesar dan nukau terkecil*

*printf("Nilai biggest ke 1 adalah %i \n", biggest);*

*printf("Nilai biggest ke 2 adalah %i", sec\_biggest);*

*return 0;*

*}*

Pseudocode No 3:

*// program untuk mencari nilai terbesar dan nilai terbesar kedua*

*// deklarasi variable*

*var i: int*

*var (arr\_len, biggest, sec\_biggest): int*

*// deklarasi algoritma*

*// meminta masukan dari user*

*arr\_len <- read("Masukan panjang array:")*

*// deklarasi array dengan panjang custom*

*var custom\_arr: int[arr\_len]*

*// memasukkan nilai pada setiap index dalam array*

*for i <- 1 to arr\_len do*

*custom\_arr[i] <- read("nilai item pada index %i : ", i)*

*end for*

*// nilai default untuk nilai terbesar*

*biggest <- custom\_arr[0]*

*// nilai default untuk nilai terbesar kedua*

*sec\_biggest <- custom\_arr[0]*

*// loop seisi array untuk mencari nilai terbesar*

*for i <- 0 to arr\_len do*

*if custom\_arr[i] > biggest do*

*biggest <- custom\_arr[i]*

*end if*

*end for*

*// loop seisi array untuk mencari nilai terbesar kedua*

*for i <- 0 to arr\_len do*

*if custom\_arr[i] > sec\_biggest && custom\_arr[i] < biggest do*

*sec\_biggest <- custom\_arr[i]*

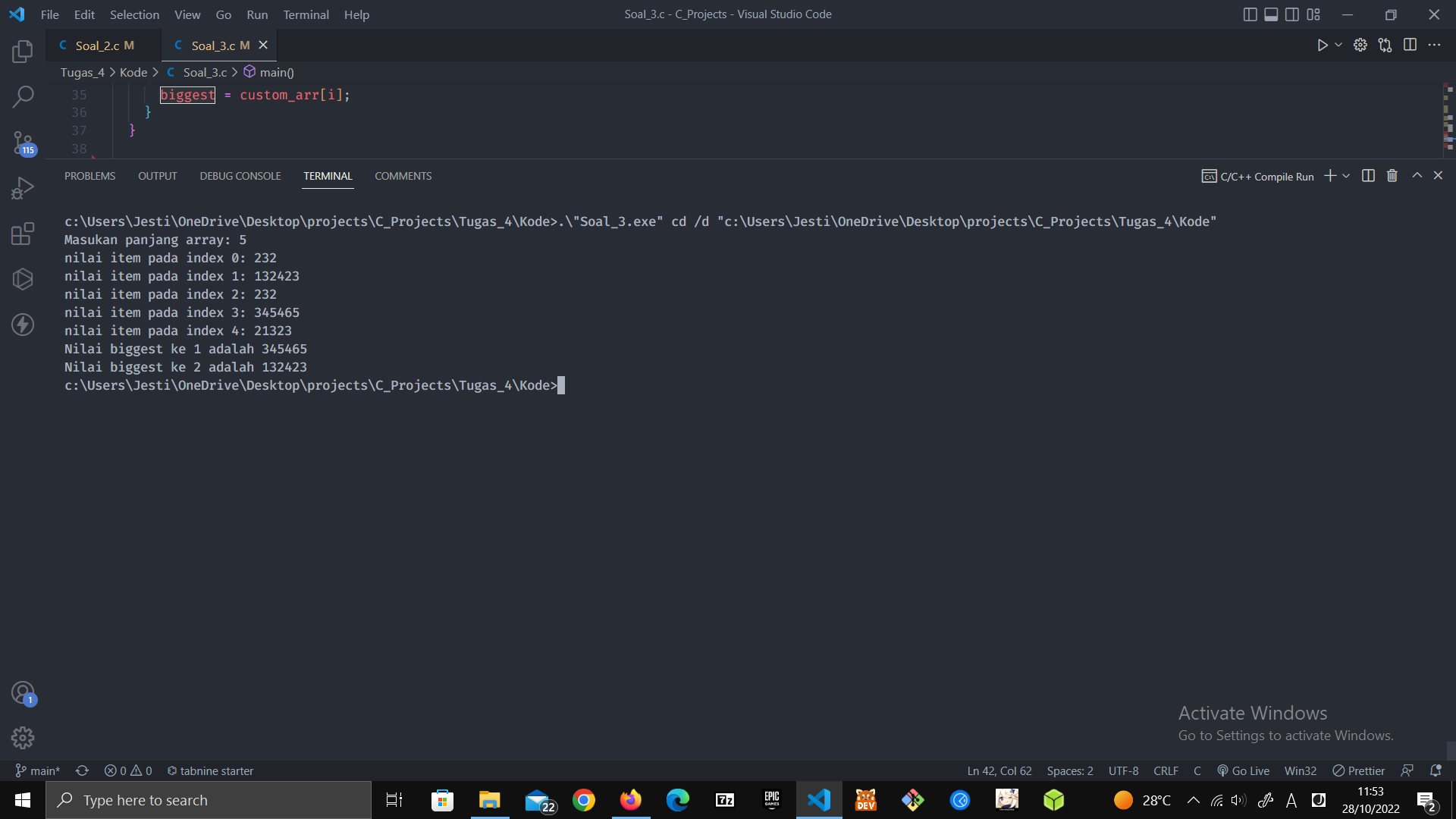
*end if*

*end for*

*// menampilkan hasil pencarian nilai terbesar dan nukau terkecil*

*write("Nilai biggest ke 1 adalah %i", biggest)*

*write("Nilai biggest ke 2 adalah %i", sec\_biggest)*



Screenshot Program No 3

1. Soal 4:

program untuk mencari nilai terkecil pada array

Kode No 4:

*// program untuk mencari nilai terkecil pada array*

*#include <stdio.h>*

*// deklarasi variable*

*int arr\_len, i, nilai\_terkecil;*

*// ALGORITMA*

*int main()*

*{*

*// meminta masukan dari user*

*printf("Masukan panjang array: ");*

*scanf("%i", &arr\_len);*

*// membuat array dengan panjang custom*

*int arr[arr\_len];*

*// meminta user untuk menginput seluruh nilai pada array*

*for (i = 0; i < arr\_len; i++)*

*{*

*printf("nilai item pada index %i: ", i);*

*scanf("%i", &arr[i]);*

*}*

*// mengisi nilai defaukt untuk nilai terkecil*

*nilai\_terkecil = arr[1];*

*// meloop seisi array untuk mencari nilai terkecil*

*for (i = 0; i < arr\_len; i++)*

*{*

*// pengecekan nilai saat ini lebih kecil dari sebelumnya*

*if (arr[i] < nilai\_terkecil)*

*nilai\_terkecil = arr[i];*

*}*

*// mecetak hasil nilai terkecil*

*printf("Nilai terkecil pada array adalah %i", nilai\_terkecil);*

*return 0;*

*}*

Pseudocode No 4:

*// program untuk mencari nilai terkecil pada array*

*// deklarasi variable*

*var arr\_len: int*

*var i: int*

*var nilai\_terkecil: int*

*// ALGORITMA*

*// meminta masukan dari user*

*arr\_len <- read("Masukan panjang array:")*

*// membuat array dengan panjang custom*

*var arr: int[arr\_len]*

*// meminta user untuk menginput seluruh nilai pada array*

*for i <- 0 to arr\_len do*

*arr[i] <- read("nilai item pada index %i: ", i)*

*end for*

*// mengisi nilai defaukt untuk nilai terkecil*

*nilai\_terkecil <- arr[1]*

*// meloop seisi array untuk mencari nilai terkecil*

*for i <- 0 to arr\_len do*

*// pengecekan nilai saat ini lebih kecil dari sebelumnya*

*if arr[i] < nilai\_terkecil do*

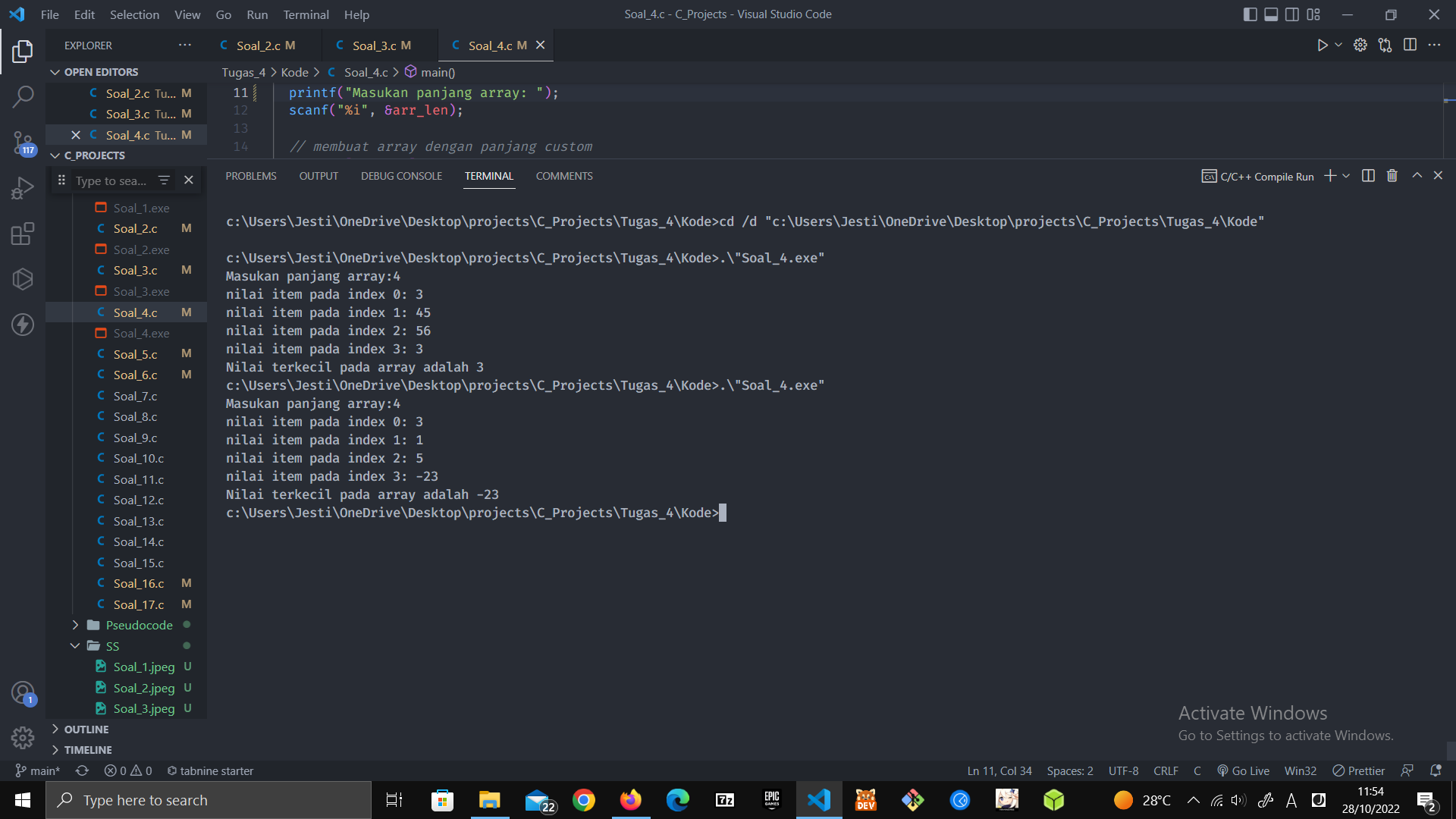
*nilai\_terkecil = arr[i]*

*end if*

*end*

*// mecetak hasil nilai terkecil*

*write("Nilai terkecil pada array adalah %i", nilai\_terkecil)*



Screenshot Program No 4

1. Soal 5:

Program ini akan menukar nilai pada array arr\_1 dan arr\_2

Kode No 5:

*// Program ini akan menukar nilai pada array arr\_1 dan arr\_2*

*#include <stdio.h>*

*// Deklarasi variabel*

*int arr\_1[] = {16, 23, 55, 23, 23, 11, 3};*

*int arr\_2[] = {6, 98, 45, 5, 87, 3, 3};*

*// prosedur*

*void print\_arr(int arr[], int arr\_len);*

*void tukar\_elemen(int i);*

*int main()*

*{*

*printf("sebelum ditukar: \n");*

*print\_arr(arr\_1, sizeof(arr\_1) / sizeof(arr\_1[0]));*

*printf("\n");*

*print\_arr(arr\_2, sizeof(arr\_2) / sizeof(arr\_2[0]));*

*tukar\_elemen(6);*

*printf("\nsesudah ditukar: \n");*

*print\_arr(arr\_1, sizeof(arr\_1) / sizeof(arr\_1[0]));*

*printf("\n");*

*print\_arr(arr\_2, sizeof(arr\_2) / sizeof(arr\_2[0]));*

*printf("\n");*

*return 0;*

*}*

*// definisi prosedur*

*void print\_arr(int arr[], int arr\_len)*

*{*

*// deklarasi variabel index untuk loop*

*int i;*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*if (i == arr\_len - 1)*

*{*

*printf("%i", arr[i]);*

*}*

*else*

*{*

*printf("%i, ", arr[i]);*

*}*

*}*

*printf(" ]");*

*}*

*void tukar\_elemen(int arr\_len)*

*{*

*// deklarasi variable untuk memebantu proses penukaran*

*int i, temp;*

*// proses menukar isi antara 2 array*

*for (i = 0; i < arr\_len; i++)*

*{*

*temp = arr\_1[i];*

*arr\_1[i] = arr\_2[i];*

*arr\_2[i] = temp;*

*}*

*}*

Pseudocode No 5:

*// Program ini akan menukar nilai pada array arr\_1 dan arr\_2*

*// Deklarasi variabel*

*var arr\_1: int[] = [16, 23, 55, 23, 23, 11, 3]*

*var arr\_2: int[] = [6, 98, 45, 5, 87, 3, 3]*

*// prosedur*

*fn print\_arr(int arr[], int arr\_len): void*

*fn tukar\_elemen(int i): void*

*// deklarasi algoritma*

*fn main(): int do*

*write("sebelum ditukar: \n")*

*print\_arr(arr\_1, sizeof(arr\_1) / sizeof(arr\_1[0]))*

*write("\n")*

*print\_arr(arr\_2, sizeof(arr\_2) / sizeof(arr\_2[0]))*

*tukar\_elemen(6)*

*write("\nsesudah ditukar: \n")*

*print\_arr(arr\_1, sizeof(arr\_1) / sizeof(arr\_1[0]))*

*write("\n")*

*print\_arr(arr\_2, sizeof(arr\_2) / sizeof(arr\_2[0]))*

*write("\n")*

*return 0*

*end fn*

*// definisi prosedur*

*fn print\_arr(int arr[], int arr\_len): void do*

*// deklarasi variabel index untuk loop*

*var i: int*

*write("[ ")*

*for i <- 0 to arr\_len do*

*if i == arr\_len - 1 do*

*write("%i", arr[i])*

*else do*

*write("%i, ", arr[i])*

*end if*

*end for*

*write(" ]")*

*end fn*

*fn tukar\_elemen(int arr\_len): void do*

*// deklarasi variable untuk memebantu proses penukaran*

*var (i, temp): int*

*// proses menukar isi antara 2 array*

*for i <- 0 to arr\_len do*

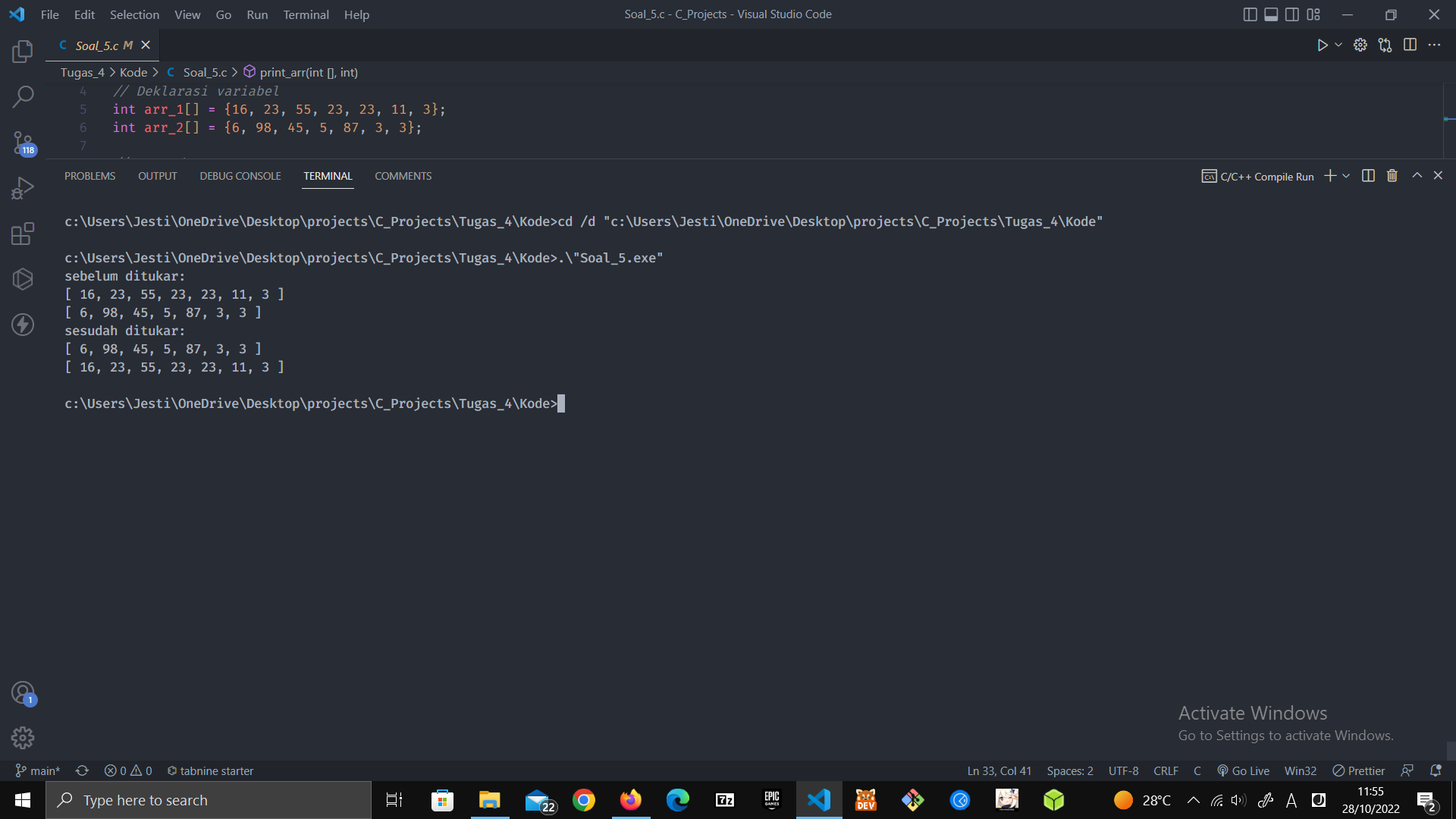
*temp <- arr\_1[i]*

*arr\_1[i] <- arr\_2[i]*

*arr\_2[i] <- temp*

*end for*

*end fn*



Screenshot Program No 5

1. Soal 6:

program ini akan membalikkan elemen char pada arr

Kode No 6:

*// program ini akan membalikkan elemen char pada arr*

*#include <stdio.h>*

*#include <string.h>*

*#include <stdlib.h>*

*// deklarasi variable*

*char input\_str[50];*

*// deklarasi prosedur*

*void balik\_str(char input\_str[]);*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta masukan string dari user*

*printf("Masukkan string: ");*

*fflush(stdin);*

*gets(input\_str);*

*system("cls");*

*printf("Input String: %s\n", input\_str);*

*// call revstr() function*

*balik\_str(input\_str);*

*printf("Reversed String: %s", input\_str);*

*return 0;*

*}*

*// definisi prosedur*

*void balik\_str(char input\_str[])*

*{*

*// deklarasi variabel*

*int i, str\_len = strlen(input\_str), temp;*

*// proses pembalikkan elemen*

*for (i = 0; i < str\_len / 2; i++)*

*{*

*temp = input\_str[i]; // temp akan mengandung nilai ascii karakter*

*input\_str[i] = input\_str[str\_len - i - 1];*

*input\_str[str\_len - i - 1] = temp;*

*}*

*}*

Pseudocode No 6:

*// program ini akan membalikkan elemen char pada arr*

*// deklarasi variable*

*var input\_str: char[50]*

*// deklarasi prosedur*

*fn balik\_str(char input\_str[]): void*

*// deklarasi algoritma*

*fn main(): int do*

*// meminta masukan string dari user*

*input\_str <- read("Masukkan sebuah string: ")*

*write("String sebelum di balik adalah %s\n", input\_str)*

*// call revstr() function*

*balik\_str(input\_str)*

*write(" setelah di balik: %s", input\_str)*

*return 0*

*end fn*

*// definisi prosedur*

*fn balik\_str(char input\_str[]): void do*

*// deklarasi variabel*

*var (i, str\_len <- input\_str.length, temp): int*

*// proses pembalikkan elemen*

*for i <- 0 i to str\_len do*

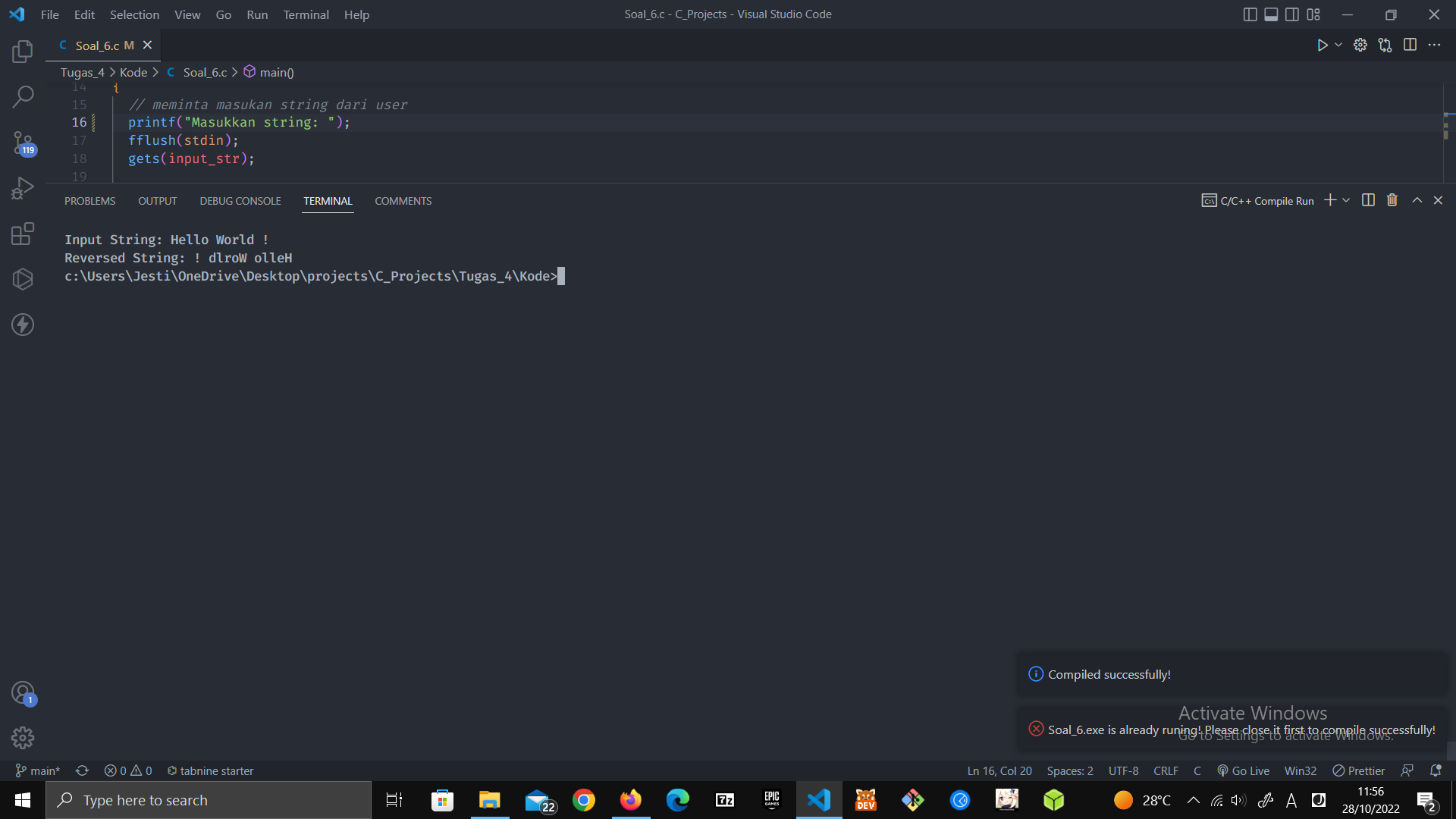
*temp <- input\_str[i] // temp akan mengandung nilai ascii karakter*

*input\_str[i] <- input\_str[str\_len - i - 1]*

*input\_str[str\_len - i - 1] <- temp*

*end for*

*end fn*



Screenshot Program No 6

1. Soal 7:

program ini akan menghitung simpangan baku dari sebuah array

Kode No 7:

*// program ini akan menghitung simpangan baku dari sebuah array*

*#include <stdio.h>*

*#include <math.h>*

*// deklarasi variable*

*int i;*

*int arr[] = {150, 167, 175, 157, 165, 153, 177, 160};*

*int arr\_len = sizeof(arr) / sizeof(arr[0]);*

*int sum = 0;*

*float rata\_rata = 0; //∑ xi/n*

*float simpangan\_total; // ∑(xi - x̄) ^ 2*

*float simpangan\_baku;*

*// deklarasi algoritma*

*int main()*

*{*

*// menambahkan seluruh item pada array*

*for (i = 0; i < arr\_len; i++)*

*{*

*sum = sum + arr[i];*

*}*

*// menghitung rata-rata*

*rata\_rata = sum / arr\_len;*

*// menghitung simpangan total*

*for (i = 0; i < arr\_len; i++)*

*{*

*simpangan\_total = simpangan\_total + pow(arr[i] - rata\_rata, 2);*

*}*

*// menghitung simpangろan baku*

*simpangan\_baku = sqrt(simpangan\_total / arr\_len);*

*// mencetak hasil*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*i == arr\_len - 1 ? printf("%i", arr[i]) : printf("%i, ", arr[i]);*

*}*

*printf(" ]\n");*

*printf("Simpangan Baku bernilai: %.2f\n", simpangan\_baku);*

*}*

Pseudocode No 7:

*// program ini akan menghitung simpangan baku dari sebuah array*

*// deklarasi variable*

*var i: int*

*var j: int*

*var arr: int[] <- [150, 167, 175, 157, 165, 153, 177, 160]*

*var arr\_len: int <- arr.length*

*var sum: int = 0*

*float rata\_rata = 0 //∑ xi/n*

*float simpangan\_total // ∑(xi - x̄) ^ 2*

*float simpangan\_baku*

*// deklarasi algoritma*

*// menambahkan seluruh item pada array*

*for i <- 0 to arr\_len do*

*sum <- sum + arr[i]*

*end for*

*// menghitung rata-rata*

*rata\_rata <- sum / arr\_len*

*// menghitung simpangan total*

*for i <- 0 to arr\_len do*

*simpangan\_total <- simpangan\_total + pow(arr[i] - rata\_rata, 2)*

*end for*

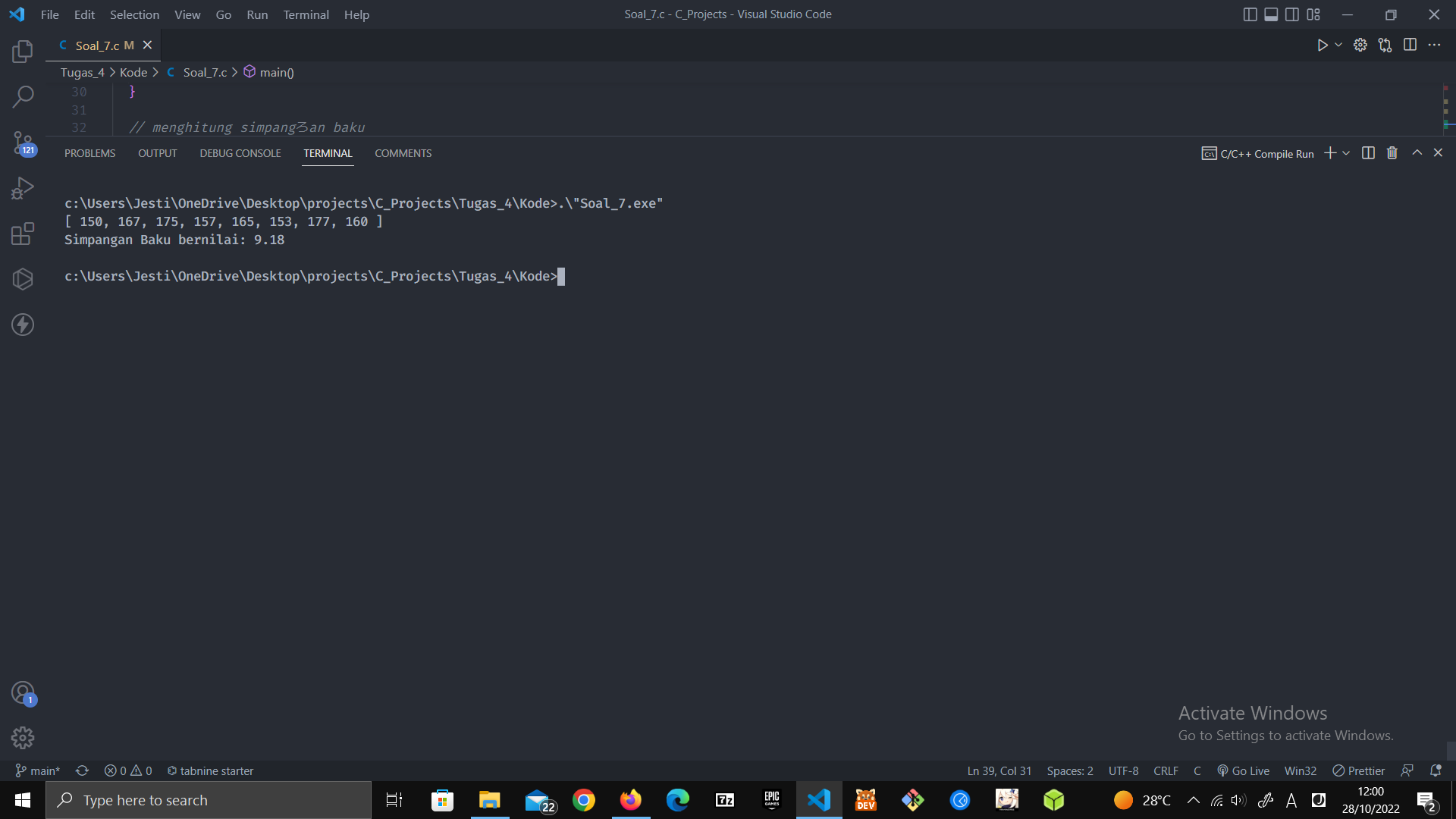
*// menghitung simpangろan baku*

*simpangan\_baku <- sqrt(simpangan\_total / arr\_len)*

*// mencetak hasil*

*write("%arr", arr)*

*write("Simpangan Baku bernilai: %.2f", simpangan\_baku)*



Screenshot Program No 7

1. Soal 8:

program ini berisi prosedur untuk menampilkan elemen terbesar

Kode No 8:

*// program ini berisi prosedur untuk menampilkan elemen terbesar larik A*

*#include <stdio.h>*

*// deklarasi variable*

*int arr[] = {2, 10, 5, 20, 50};*

*int hasil, i;*

*// deklarasi prosedur*

*void print\_terbesar(int arr[], int \*hasil, int arr\_len);*

*// deklarasi algoritma*

*int main()*

*{*

*// memanggil prosedur*

*print\_terbesar(arr, &hasil, sizeof(arr) / sizeof(arr[0]));*

*return 0;*

*}*

*// definisi prosedur*

*void print\_terbesar(int arr[], int \*hasil, int arr\_len)*

*{*

*\*hasil = arr[0];*

*for (i = 1; i < arr\_len; i++)*

*{*

*if (arr[i] > \*hasil)*

*\*hasil = arr[i];*

*}*

*// mencetak hasil*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*i == arr\_len - 1 ? printf("%i", arr[i]) : printf("%i, ", arr[i]);*

*}*

*printf(" ]\n");*

*printf("Hasil terkecil di array adalah %i", \*hasil);*

*}*

Pseudocode No 8:

*// program ini berisi prosedur untuk menampilkan elemen terbesar larik A*

*// deklarasi variable*

*var arr: int[] <- [2, 10, 5, 20, 50]*

*var hasi: int*

*// deklarasi prosedur*

*fn print\_terbesar(arr: int [], \*hasil: int , arr\_len: int): void*

*// deklarasi algoritma*

*fn main(): int do*

*// memanggil prosedur*

*print\_terbesar(arr, &hasil, sizeof(arr) / sizeof(arr[0]))*

*return 0*

*end fn*

*// definisi prosedur*

*fn print\_terbesar(arr: int[], \*hasil: int, arr\_len: int): void do*

*\*hasil <- arr[0]*

*for i <- 1 to arr\_len do*

*if arr[i] > \*hasil do*

*\*hasil <- arr[i]*

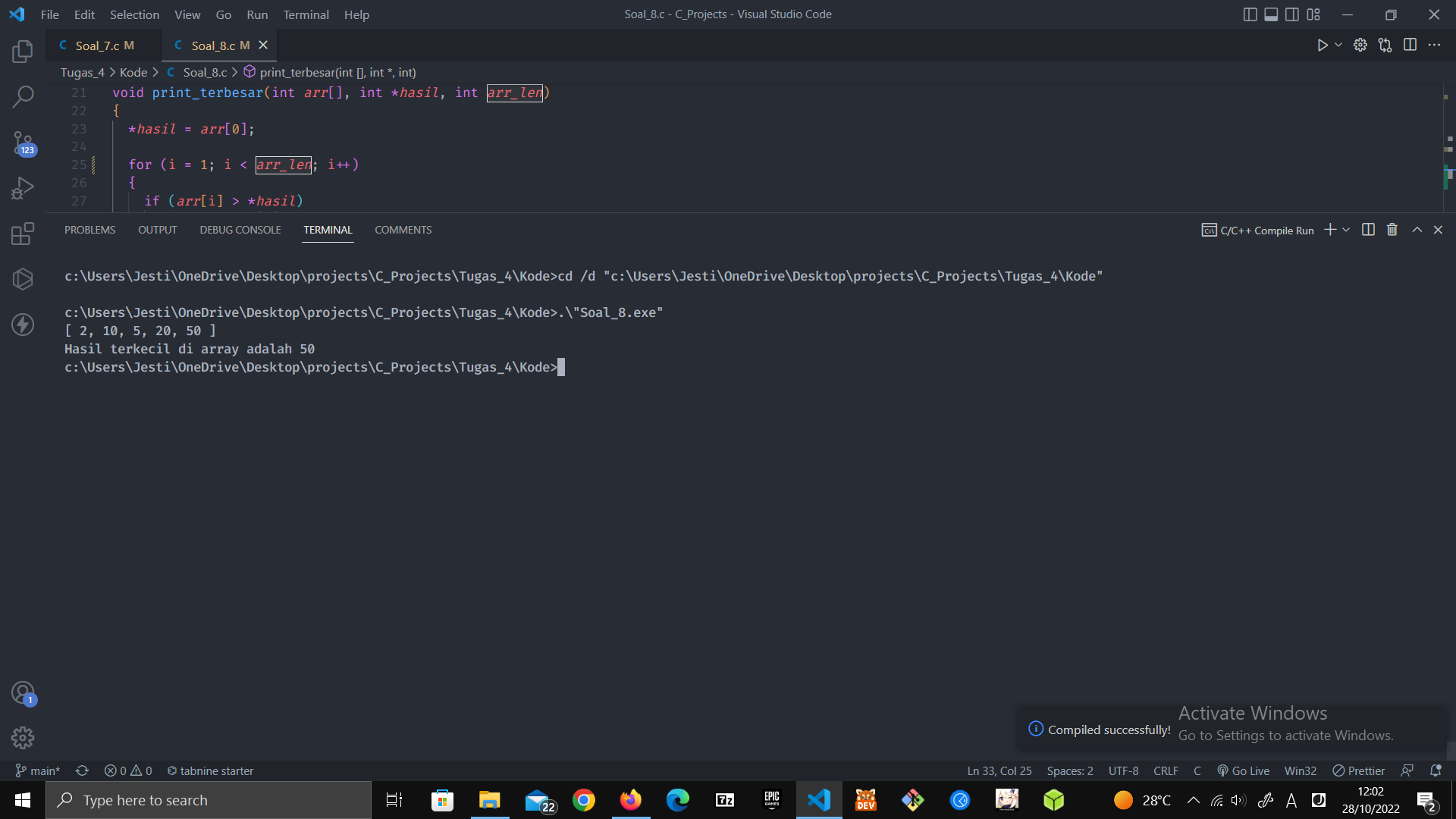
*end if*

*end for*

*write("%arr", arr)*

*write("Hasil terkecil di array adalah %i", \*hasil)*

*end fn*



Screenshot Program No 8

1. Soal 9:

program ini berisi prosedur untuk menampilkan elemen terkecil

Kode No 9:

*// program ini berisi prosedur untuk menampilkan elemen terkecil larik A*

*#include <stdio.h>*

*// deklarasi variable*

*int arr[] = {2, 10, 5, 20, 50};*

*int hasil;*

*int arr\_len = sizeof(arr) / sizeof(arr[0]);*

*// deklarasi prosedur*

*void print\_terkecil(int arr[], int \*hasil, int arr\_len);*

*// deklarasi algoritma*

*int main()*

*{*

*// memanggil prosedur*

*print\_terkecil(arr, &hasil, arr\_len);*

*return 0;*

*}*

*// definisi prosedur*

*void print\_terkecil(int arr[], int \*hasil, int arr\_len)*

*{*

*int i;*

*\*hasil = arr[0];*

*for (i = 1; i < arr\_len; i++)*

*{*

*if (\*hasil > arr[i])*

*\*hasil = arr[i];*

*}*

*// mencetak hasil*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*i == arr\_len - 1 ? printf("%i", arr[i]) : printf("%i, ", arr[i]);*

*}*

*printf(" ]\n");*

*printf("Hasil terkecil di array adalah %i", \*hasil);*

*}*

Pseudocode No 9:

*// program ini berisi prosedur untuk menampilkan elemen terkecil larik A*

*// deklarasi variable*

*var arr: int[] <- [2, 10, 5, 20, 50]*

*var hasil: int*

*var arr\_len: int <- arr.length*

*// deklarasi prosedur*

*fn print\_terkecil(arr: int[], \*hasil: int, arr\_len: int): void*

*// deklarasi algoritma*

*fn main(): void do*

*// memanggil prosedur*

*print\_terkecil(arr, &hasil, arr\_len)*

*return 0*

*end fn*

*// definisi prosedur*

*fn print\_terkecil(arr: int[], \*hasil: int, arr\_len: int): void do*

*\*hasil <- arr[0]*

*for i <- 1 to arr.length do*

*if \*hasil > arr[i] do*

*\*hasil <- arr[i]*

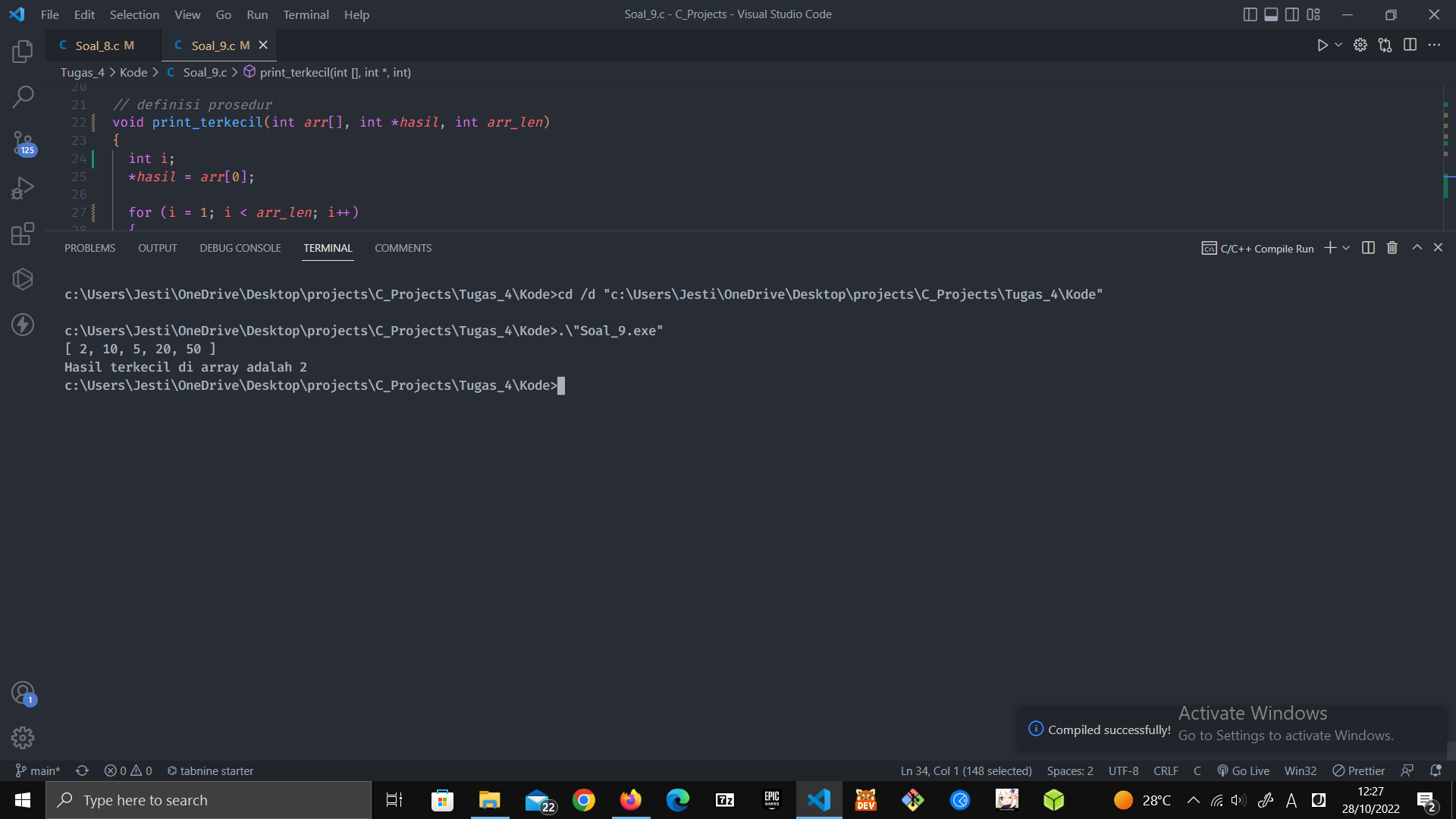
*end if*

*end for*

*write("%arr", arr)*

*write("Hasil terkecil di array adalah %i", \*hasil)*

*end fn*



Screenshot Program No 9

1. Soal 10:

program ini berisi prosedur untuk menampilkan elemen-elemen larik

Kode No 10:

*// program ini berisi prosedur untuk menampilkan elemen-elemen larik A yang lebih kecil dari elemen terkecilnya*

*#include <stdio.h>*

*// deklarasi variable*

*int arr\_1[] = {2, 4, 6, 8};*

*int arr\_2[] = {2, 3, 4, 5, 6};*

*int hasil;*

*// deklarasi prosedur*

*void print\_median(int arr[], int \*hasil, int arr\_len);*

*// deklarasi algoritma*

*int main()*

*{*

*print\_median(arr\_1, &hasil, sizeof(arr\_1) / sizeof(arr\_1[0]));*

*print\_median(arr\_2, &hasil, sizeof(arr\_2) / sizeof(arr\_2[0]));*

*return 0;*

*}*

*// definisi prosedur*

*void print\_median(int arr[], int \*hasil, int arr\_len)*

*{*

*int i;*

*// validasi panjang array untuk mengekesuki rumus yang berbeda*

*\*hasil = arr\_len % 2 == 0*

*? (arr[(arr\_len - 1) / 2] + arr[arr\_len / 2]) / 2*

*: arr[arr\_len / 2];*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*i == arr\_len - 1 ? printf("%i", arr[i]) : printf("%i, ", arr[i]);*

*}*

*printf(" ]\n");*

*printf("Median dari array adalah %i\n", \*hasil);*

*}*

Pseudocode No 10:

*// program ini berisi prosedur untuk menampilkan elemen-elemen larik A yang lebih kecil dari elemen terkecilnya*

*// deklarasi variable*

*var arr\_1: int[] <- [2, 4, 6, 8]*

*var arr\_2: int[] <- [2, 3, 4, 5, 6]*

*var hasil: int*

*// deklarasi prosedur*

*fn print\_median(arr: int[], \*hasil: int, arr\_len: int): void*

*// deklarasi algoritma*

*fn main(): void do*

*print\_median(arr\_1, &hasil, arr\_1.length)*

*print\_median(arr\_2, &hasil, arr\_2.length)*

*return 0*

*end fn*

*// definisi prosedur*

*fn print\_median(arr: int[], \*hasil: int, arr\_len: int): void do*

*var arr\_len: intt <- arr.length*

*// validasi panjang array untuk mengekesuki rumus yang berbeda*

*\*hasil <- arr\_len % 2 == 0*

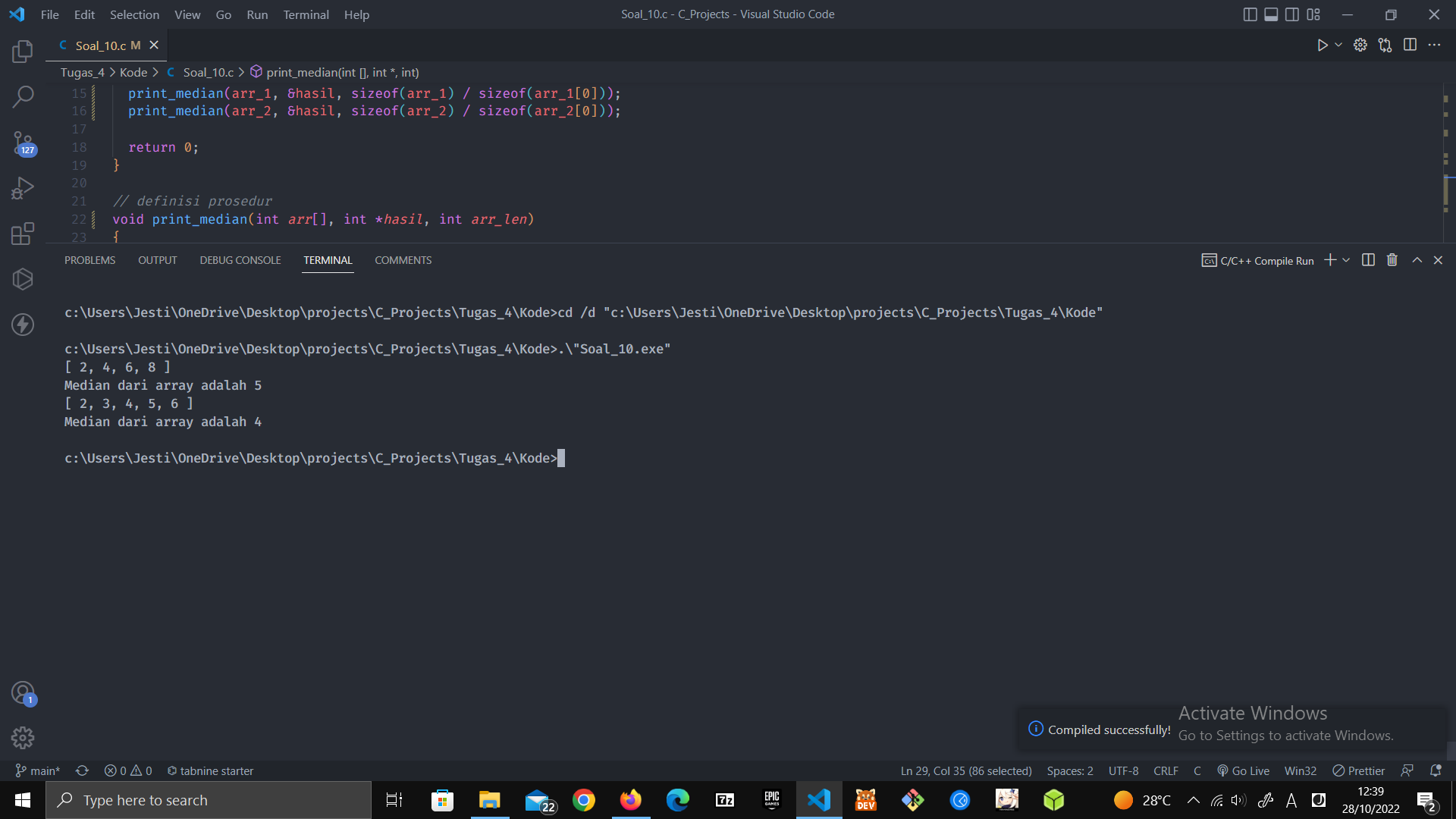
*? (arr[(arr\_len - 1) / 2] + arr[arr\_len / 2]) / 2*

*: arr[arr\_len / 2];*

*write("%arr", arr)*

*write("Median dari array adalah %i", \*hasil)*

*end fn*



Screenshot Program No 10

1. Soal 11:

program ini berisi prosedur untuk menentukan apakah semua isi pada

Kode No 11:

*// program ini berisi prosedur untuk menentukan apakah semua isi pada array bernilai sama*

*#include <stdio.h>*

*#include <stdbool.h>*

*// deklarasi variable*

*int arr\_1[] = {2, 2, 2, 2};*

*int arr\_2[] = {2, 4, 2, 2};*

*// deklarasi prosedur*

*void is\_all\_item\_same(int arr[], int arr\_len);*

*// deklarasi algoritma*

*int main()*

*{*

*is\_all\_item\_same(arr\_1, sizeof(arr\_1) / sizeof(arr\_1[0]));*

*is\_all\_item\_same(arr\_2, sizeof(arr\_2) / sizeof(arr\_2[0]));*

*return 0;*

*}*

*// definisi prosedur*

*void is\_all\_item\_same(int arr[], int arr\_len)*

*{*

*int i;*

*int temp = arr[0]; // penanmpung sementara item pada array*

*bool is\_all\_same = true;*

*// meloop seluruh array item*

*for (i = 0; i < arr\_len; i++)*

*{*

*// jika pada iterasi ini nilainya sama maka lanjut ke iterasi berikutnya*

*if (temp == arr[i])*

*continue;*

*is\_all\_same = false;*

*break;*

*}*

*// menampilkan array*

*printf("[ ");*

*for (i = 0; i < arr\_len; i++)*

*{*

*i == arr\_len - 1 ? printf("%i", arr[i]) : printf("%i, ", arr[i]);*

*}*

*printf(" ]\n");*

*// menampilkan keluaran berdasarkan variable is\_all\_same*

*is\_all\_same*

*? printf("Semua elemen pada array sama !\n")*

*: printf("Semua elemen pada array tidak sama !\n");*

*}*

Pseudocode No 11:

*// program ini berisi prosedur untuk menentukan apakah semua isi pada array bernilai sama*

*// deklarasi variable*

*var arr\_1: int[] <- [2, 2, 2, 2]*

*var arr\_2: int[] <- [2, 4, 2, 2]*

*// deklarasi prosedur*

*fn is\_all\_item\_same(arr: int[], arr\_len: int): void*

*// deklarasi algoritma*

*fn main(): int do*

*is\_all\_item\_same(arr\_1. arr\_1.length)*

*is\_all\_item\_same(arr\_2. arr\_2.length)*

*return 0*

*end fn*

*// definisi prosedur*

*fn is\_all\_item\_same(arr: int[], arr\_len: int): void do*

*var arr\_len: int <- arr.length*

*var temp: int <- arr[0] // penanmpung sementara item pada array*

*var is\_all\_same: boolean <- true*

*// meloop seluruh array item*

*for i <- 0 to arr\_len do*

*// jika pada iterasi ini nilainya sama maka lanjut ke iterasi berikutnya*

*if temp == arr[i] do*

*continue*

*end if*

*is\_all\_same <- false*

*break*

*end for*

*// menampilkan array*

*write("%arr", arr)*

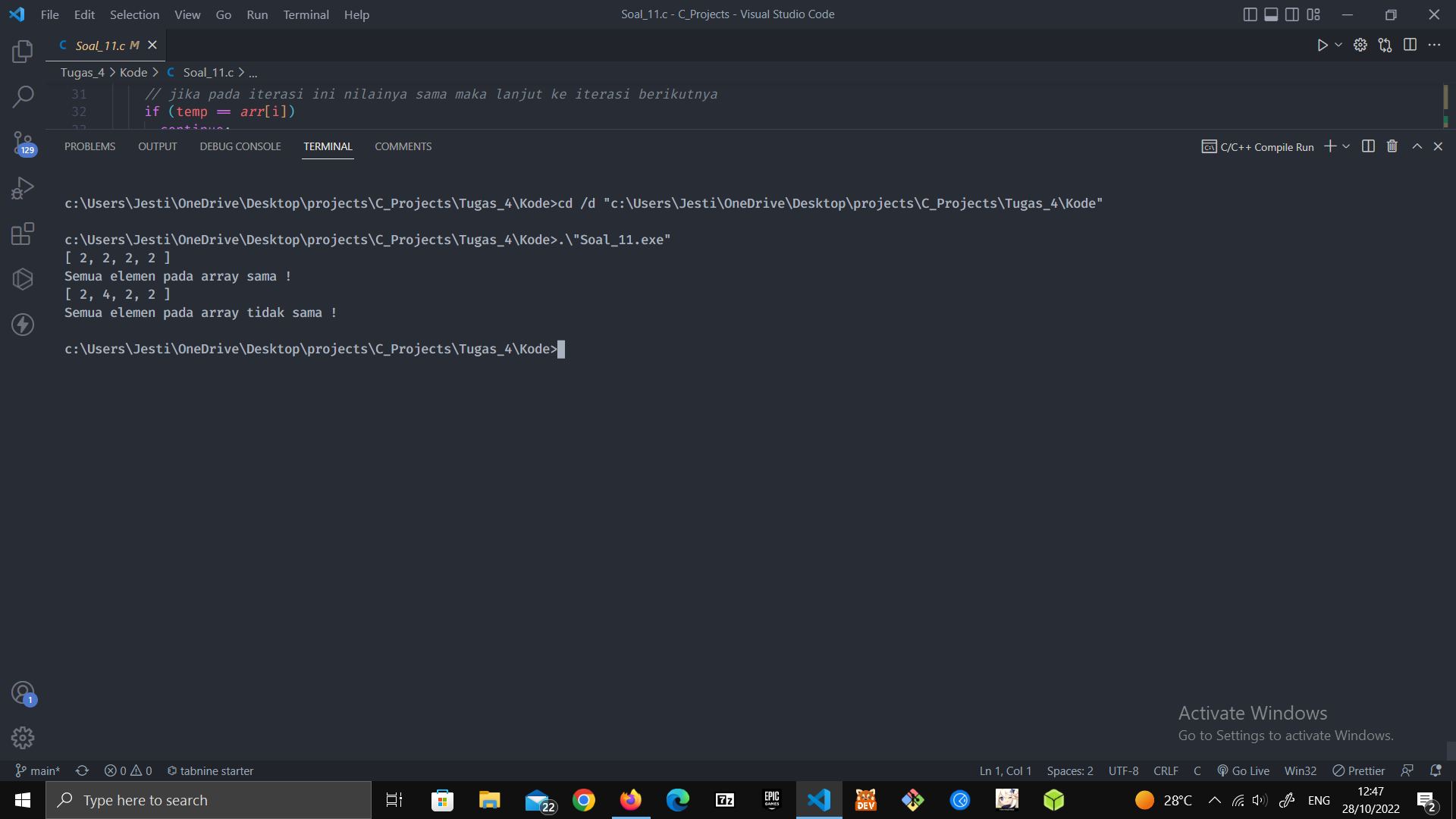
*// menampilkan keluaran berdasarkan variable is\_all\_same*

*is\_all\_same*

*? write("Semua elemen pada array sama !")*

*: write("Semua elemen pada array tidak sama !")*

*end fn*



Screenshot Program No 11

1. Soal 12:

program ini berisi fungsi yang mengembalikan sebuah

Kode No 12:

*// program ini berisi fungsi yang mengembalikan sebuah*

*#include <stdio.h>*

*#include <string.h>*

*// deklarasi variable*

*int start\_cut\_idx, end\_cut\_idx;*

*// deklarasi prosedur*

*void cut\_str(char str[], int start\_idx, int end\_idx);*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta masukan string ke user*

*int arr\_len;*

*printf("Masukkan panjang string: ");*

*scanf("%i", &arr\_len);*

*// membuat string baru*

*char str[arr\_len + 1];*

*// meminta user untuk memasukan input string baru*

*printf("Masukkan sebuah string: ");*

*fflush(stdin);*

*gets(str);*

*// memasukan titik awal pemotongan string*

*printf("Masukkan index titik awal pemotongan string: ");*

*scanf("%i", &start\_cut\_idx);*

*printf("Masukkan index titik akhir pemotongan string: ");*

*scanf("%i", &end\_cut\_idx);*

*cut\_str(str, start\_cut\_idx, end\_cut\_idx);*

*return 0;*

*}*

*// definisi prosedur*

*void cut\_str(char str[], int start\_idx, int end\_idx)*

*{*

*int str\_len = strlen(str);*

*// validasi terhadap input index mulai dan index akhir potongan string*

*if (start\_idx > str\_len - 1 || (end\_idx > str\_len - start\_idx))*

*{*

*printf("%s", "");*

*}*

*else*

*{*

*// loop untuk mencetak hasil potongan*

*for (int i = start\_idx; i < end\_idx; i++)*

*{*

*printf("%c", str[i]);*

*}*

*}*

*}*

Pseudocode No 12:

*// program ini berisi fungsi yang mengembalikan sebuah*

*// deklarasi variable*

*var start\_cut\_idx: int*

*var end\_cut\_idx: int*

*// deklarasi prosedur*

*fn cut\_str(str: char[], start\_idx: int, end\_idx: int : void*

*// deklarasi algoritma*

*fn main(): int do*

*// meminta masukan string ke user*

*var arr\_len: int*

*arr\_len <- read("Masukkan panjang string: ")*

*// membuat string baru*

*var str: char[arr\_len + 1]*

*// meminta user untuk memasukan input string baru*

*str <- read("Masukkan sebuah string: ")*

*// memasukan titik awal pemotongan string*

*start\_cut\_idx <- read("Masukkan index titik awal pemotongan string: ")*

*end\_cut\_idx <- read("Masukkan index titik akhir pemotongan string: ")*

*cut\_str(str, start\_cut\_idx, end\_cut\_idx)*

*return 0*

*end fn*

*// definisi prosedur*

*fn cut\_str(str: char[], start\_idx: int, end\_idx: int : void do*

*var str\_len: int <- str.length*

*// validasi terhadap input index mulai dan index akhir potongan string*

*if start\_idx > str\_len - 1 || (end\_idx > str\_len - start\_idx) do*

*printf("%s", "")*

*else do*

*// loop untuk mencetak hasil potongan*

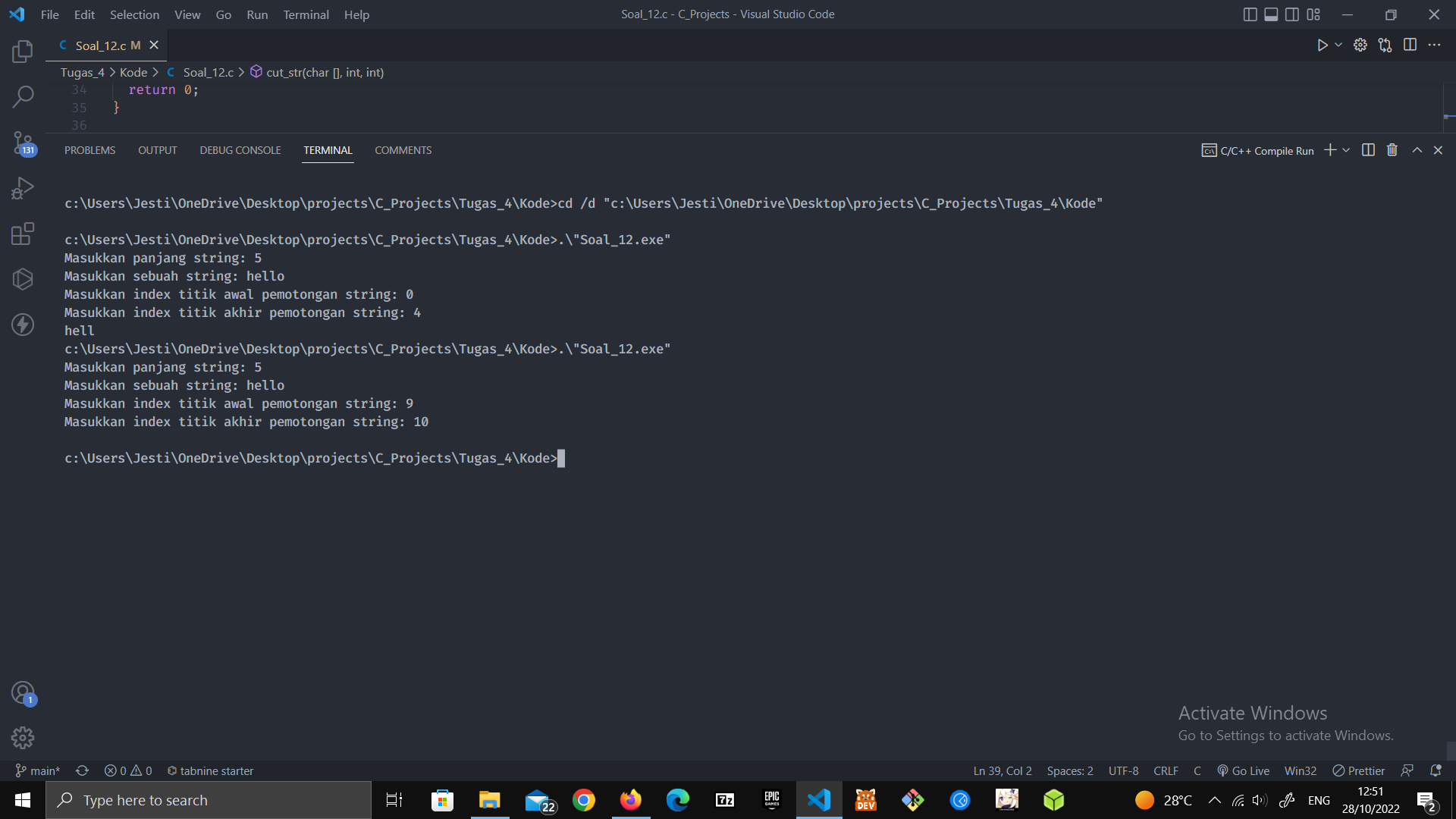
*for i <- start\_idx to end\_idx do*

*printf("%c", str[i])*

*end for*

*end if*

*end fn*



Screenshot Program No 12

1. Soal 13:

program untuk konkatenasi dua buah string

Kode No 13:

*// program untuk konkatenasi dua buah string*

*#include <stdio.h>*

*#include <stdlib.h>*

*#include <string.h>*

*// deklarasi variable*

*int str\_len;*

*// deklarasi algoritma*

*int main()*

*{*

*// meminta masukan dari user*

*printf("masukkan panjang string pertama dan kedua: ");*

*scanf("%i", &str\_len);*

*// deklarasi variable string*

*char str\_1[str\_len];*

*char str\_2[str\_len];*

*char str\_hasil[str\_len \* 2];*

*// meminta masukan string 1 dari user*

*printf("masukan string pertama: ");*

*fflush(stdin);*

*gets(str\_1);*

*// meminta masukan string 2 dari user*

*printf("masukan string kedua: ");*

*fflush(stdin);*

*gets(str\_2);*

*// mencetak hasil konkatenasi 2 string*

*printf("%s%s\n", str\_1, str\_2);*

*return 0;*

*}*

Pseudocode No 13:

*// program untuk konkatenasi dua buah string*

*// deklarasi variable*

*var str\_len: int*

*// deklarasi algoritma*

*// meminta masukan dari user*

*str\_len <- read("masukkan panjang string pertama dan kedua: ")*

*// deklarasi variable string*

*var str\_1: char[str\_len]*

*var str\_2: char[str\_len]*

*var str\_hasil: char[str\_len \* 2]*

*// meminta masukan string 1 dari user*

*str\_1 <- read("masukan string pertama: ")*

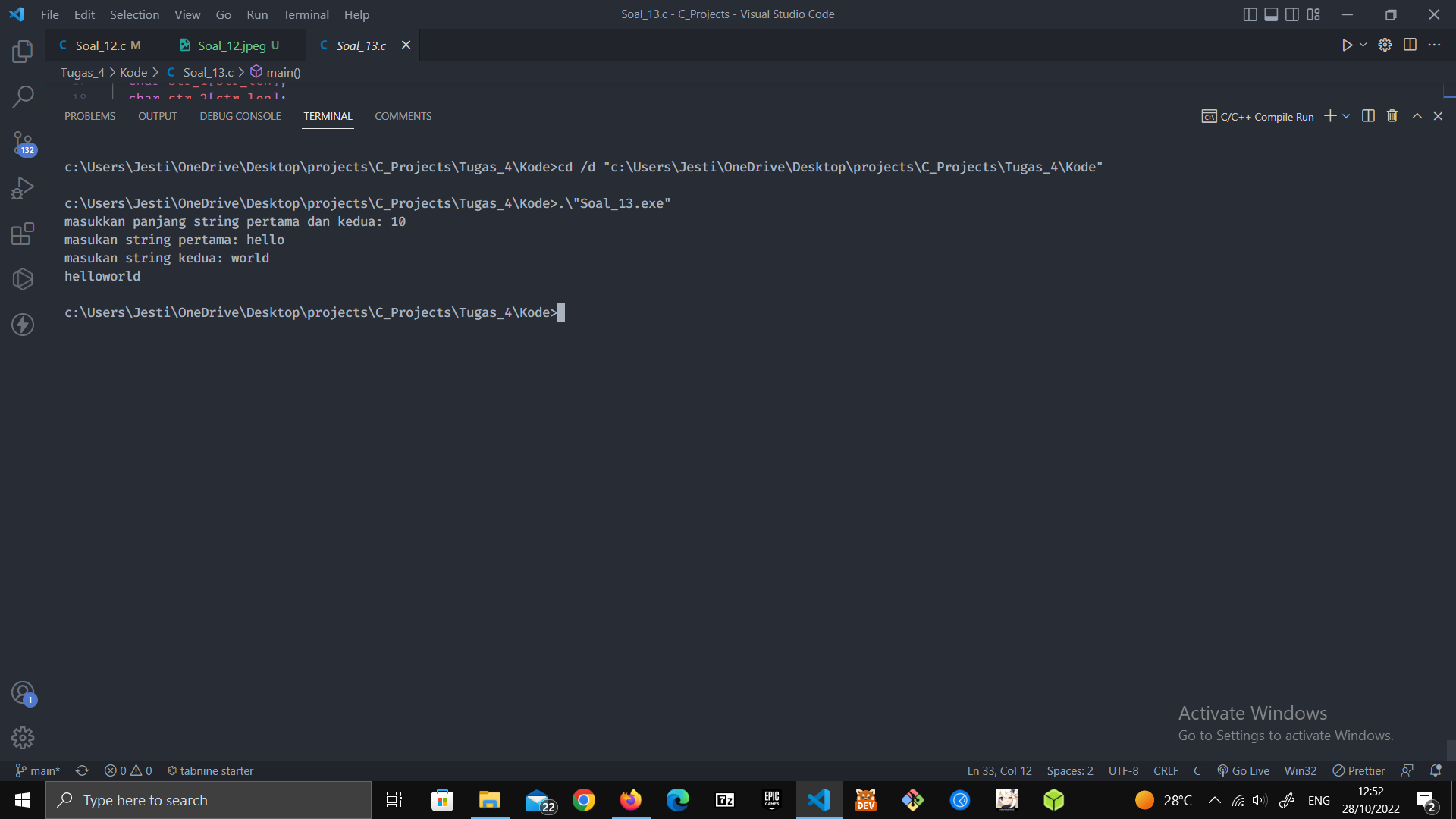
*// meminta masukan string 2 dari user*

*str\_2 <- read("masukan string kedua: ")*

*// mencetak hasil konkatenasi 2 string*

*write("%s%s", str\_1, str\_2)*

*return 0*



Screenshot Program No 13

1. Soal 14:

program untuk mencari nilai terkecil pada array

Kode No 14:

*// program untuk mencari nilai terkecil pada array*

*#include <stdio.h>*

*// deklarasi variable*

*int arr\_len, i, nilai\_terkecil;*

*// ALGORITMA*

*int main()*

*{*

*// meminta masukan dari user*

*printf("Masukan panjang array: ");*

*scanf("%i", &arr\_len);*

*// membuat array dengan panjang custom*

*int arr[arr\_len];*

*// meminta user untuk menginput seluruh nilai pada array*

*for (i = 0; i < arr\_len; i++)*

*{*

*printf("nilai item pada index %i: ", i);*

*scanf("%i", &arr[i]);*

*}*

*// mengisi nilai defaukt untuk nilai terkecil*

*nilai\_terkecil = arr[1];*

*// meloop seisi array untuk mencari nilai terkecil*

*for (i = 0; i < arr\_len; i++)*

*{*

*// pengecekan nilai saat ini lebih kecil dari sebelumnya*

*if (arr[i] < nilai\_terkecil)*

*nilai\_terkecil = arr[i];*

*}*

*// mecetak hasil nilai terkecil*

*printf("Nilai terkecil pada array adalah %i", nilai\_terkecil);*

*return 0;*

*}*

Pseudocode No 14:

*// program untuk menghilangkan setiap karakter c dalam string*

*// deklarasi variabel global*

*var str: char[120]*

*// algoritma*

*fn main(): int do*

*// meminta masukan string dari user*

*str <- read("masukkan sebuah string: ")*

*var str\_len: int <- str.length*

*for i <- 0 to str\_len do*

*// pengecekan character c dalam string*

*if str[i] == 'c' || str[i] == 'C' do*

*// menghapus karakter c dalam string*

*memmove(&str[i], &str[i + 1], strlen(str) - i)*

*str\_len--*

*i--*

*end if*

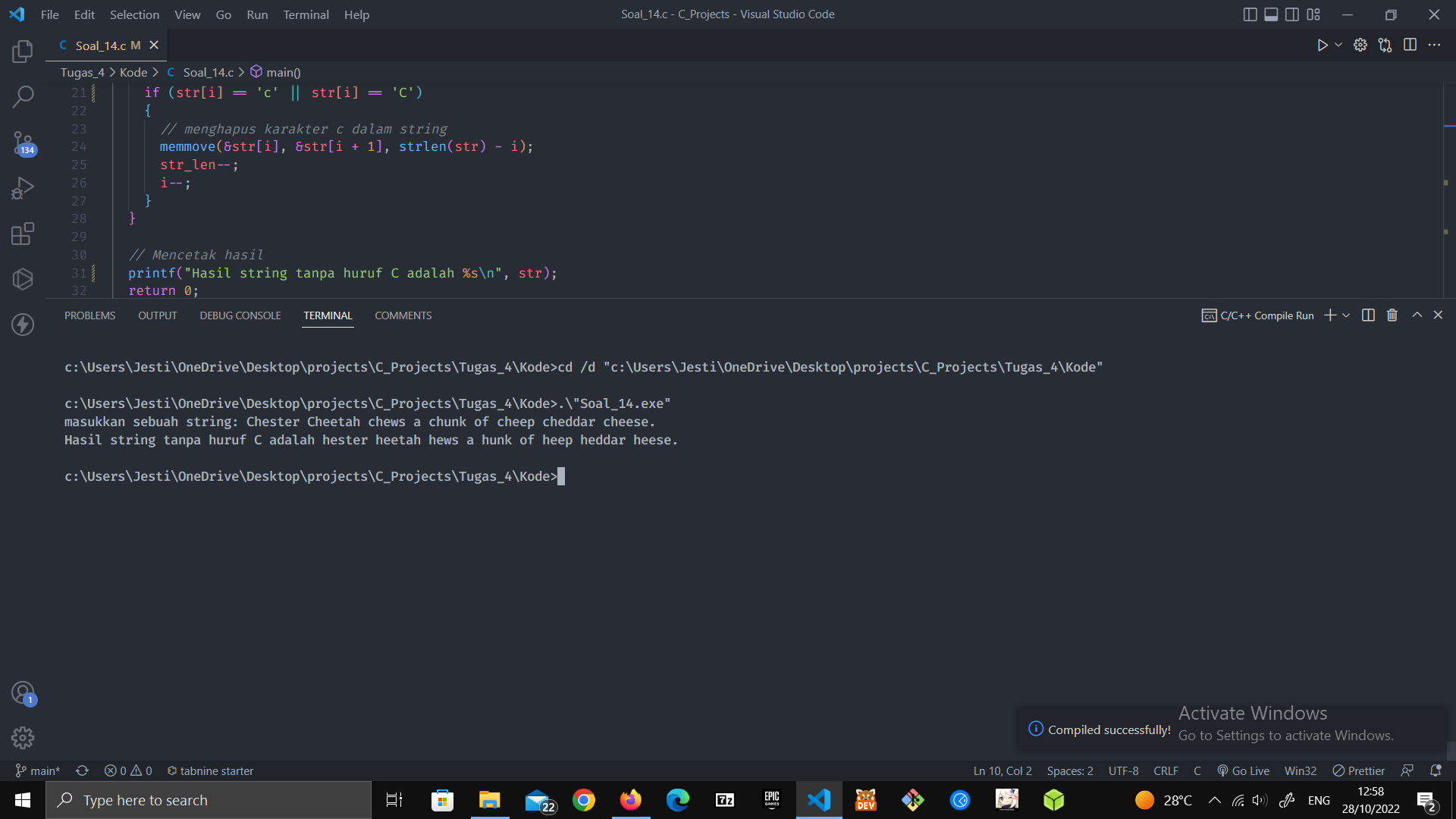
*end for*

*// Mencetak hasil*

*write("Hasil string tanpa huruf C adalah%s", str)*

*return 0*

*end for*



Screenshot Program No 14

1. Soal 15:

program untuk menggeser item pada array secara sirkuler

Kode No 15:

*// program untuk menggeser item pada array secara sirkuler*

#include <stdio.h>

*// deklarasi variable*

int arr\_len, i;

*// deklarasi algoritma*

int main()

{

*// meminta masukan dari user*

printf("masukan panjang array: ");

scanf("%i", &arr\_len);

*// array dengan ukuran custom dari user*

int custom\_arr[arr\_len];

for (i = 0; i < arr\_len; i++)

{

printf("masukan nilai array ke-%i : ", i);

scanf("%i", &custom\_arr[i]);

}

*// untuk mendeklarasi nilai akhir pada array*

int temp = custom\_arr[arr\_len - 1];

for (i = arr\_len - 1; i >= 0; i--)

{

*// mengganti posisi item pada array*

custom\_arr[i] = i != 0 ? custom\_arr[i - 1] : temp;

}

*// mencetakan hasil penukaran item pada array*

printf("Hasil pergerakan sirkuler: \n");

for (i = 0; i < arr\_len; i++)

{

printf("%i ", custom\_arr[i]);

}

return 0;

}

Pseudocode No 15:

*// program untuk menggeser item pada array secara sirkuler*

*// deklarasi variable*

*var (arr\_len, i): int*

*// deklarasi algoritma*

*// meminta masukan dari user*

*arr\_len <- read("masukan panjang array: ")*

*// array dengan ukuran custom dari user*

*var custom\_arr: int[arr\_len]*

*for i <- 0 to arr\_len i++ do*

*custom\_arr[i] <- read("masukan nilai array ke-%i : ", i)*

*end for*

*// untuk mendeklarasi nilai akhir pada array*

*var temp: int <- custom\_arr[arr\_len - 1]*

*for i <- arr\_len - 1 down to equal 0 do*

*// mengganti posisi item pada array*

*custom\_arr[i] <- i != 0 ? custom\_arr[i - 1] : temp*

*end for*

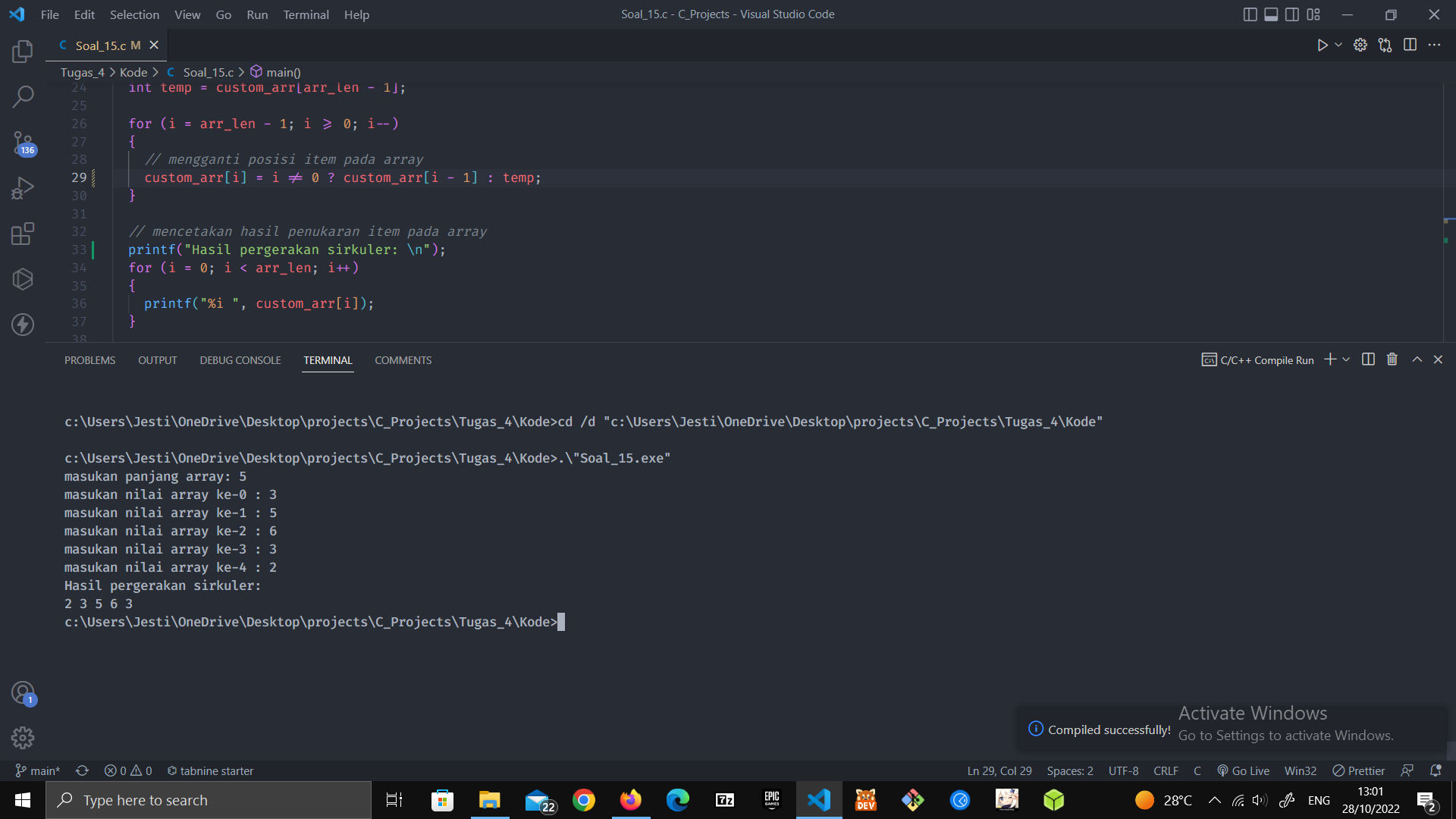
*// mencetakan hasil penukaran item pada array*

*write("Hasil pergerakan sirkuler: \n")*

*for i <- 0 to arr\_len do*

*write("%i ", custom\_arr[i])*

*end for*



Screenshot Program No 15

1. Soal 16:

program untuk menggeser item pada array secara sirkuler

Kode No 16:

*// program untuk mencari nilai modus atau nilai terbanyak dalam array*

*#include <stdio.h>*

*// deklarasi variable*

*int i, j, b = 0;*

*// Array terdefinisi*

*int arr[] = {1, 2, 2, 3, 2, 3, 3};*

*int arr\_len = sizeof(arr) / sizeof(arr[0]);*

*int arr\_item\_sum = 0;*

*// deklarasi algoritma*

*int main()*

*{*

*int hasil\_modus = arr[0];*

*// proses perhitungan modus*

*for (i = 0; i < arr\_len; i++)*

*{*

*// proses perulangan sebanyak arr\_len*

*for (j = 0; j < arr\_len; j++)*

*{*

*if (arr[i] == arr[j])*

*{*

*arr\_item\_sum++;*

*}*

*if (arr\_item\_sum > b)*

*{*

*hasil\_modus = arr[i];*

*b = arr\_item\_sum;*

*}*

*// proses apakah banyaknya nilai yang sekarang sama dengan banyaknya nilai yang sebelumnya?*

*else if (arr\_item\_sum == b)*

*{*

*// proses apakah nilai arr dengan index i lebih besar dari nilai hasil\_modus*

*if (arr[i] > hasil\_modus)*

*{*

*// proses mendeklarasi nilai hasil*

*b = arr\_item\_sum;*

*hasil\_modus = arr[i];*

*}*

*}*

*}*

*}*

*// mencetak hasil moduss*

*printf("%i", hasil\_modus);*

*return 0;*

*}*

Pseudocode No 16:

*// program untuk mencari nilai modus atau nilai terbanyak dalam array*

*// deklarasi variable*

*var (i, j, b <- 0): int*

*// Array terdefinisi*

*var arr: int [] <- [1, 2, 2, 3, 2, 3, 3]*

*var arr\_len: int <- arr.length*

*var arr\_item\_sum: int <- 0*

*// deklarasi algoritma*

*var hasil\_modus: int <- arr[0]*

*// proses perhitungan modus*

*for i <- 0 to arr\_len do*

*// PROSES perulangan sebanyak arr\_len*

*for j <- 0 to arr\_len do*

*if arr[i] == arr[j] do*

*arr\_item\_sum += 1*

*end if*

*if arr\_item\_sum > b do*

*hasil\_modus <- arr[i]*

*b <- arr\_item\_sum*

*// PROSES apakah banyaknya nilai yang sekarang sama dengan banyaknya nilai yang sebelumnya?*

*else if arr\_item\_sum == b do*

*// PROSES apakah nilai arr dengan index i lebih besar dari nilai hasil\_modus*

*if (arr[i] > hasil\_modus) do*

*// PROSES mendeklarasi nilai hasil*

*b <- arr\_item\_sum*

*hasil\_modus <- arr[i]*

*end if*

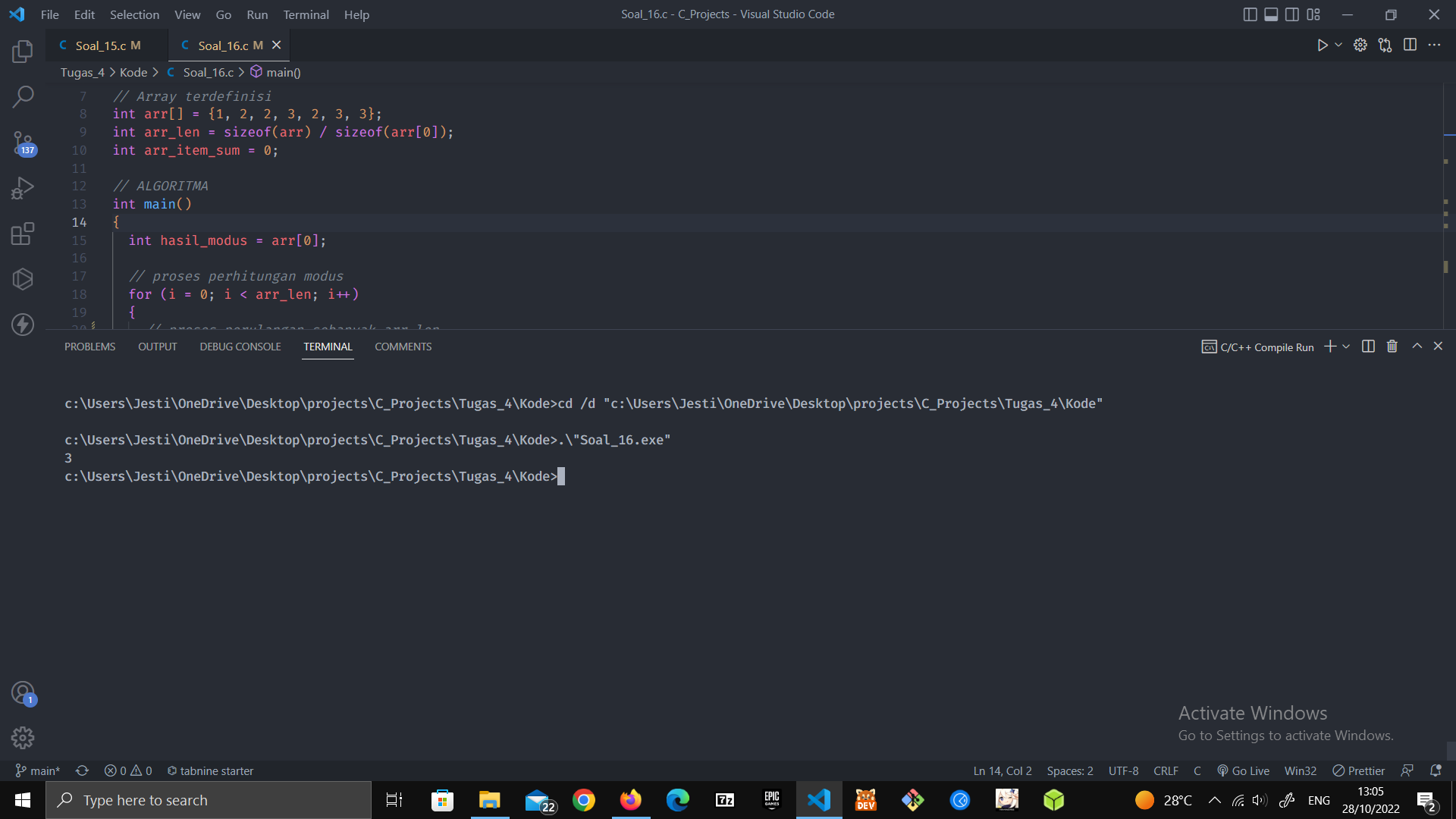
*end if*

*end for*

*end for*

*// mencetak hasil moduss*

*write("%i", hasil\_modus)*



Screenshot Program No 16

1. Soal 17:

program ini berisi element struct mahasiswa

1. membuat fungsi untuk mencetak data mahasiswa dengan IP > 2.0

2. membuat fungsi untuk mencetak nama dan nim mahasiswa dengan IPK tertinggi dan terendah

Kode No 17:

*/\* Program ini berisi element struct mahasiswa*

*1. membuat fungsi untuk mencetak data mahasiswa dengan IP > 2.0*

*2. membuat fungsi untuk mencetak nama dan nim mahasiswa dengan IPK tertinggi dan terendah*

*\*/*

*// deklarasi pustaka*

#include <stdio.h>

#include <stdbool.h>

#include <stdlib.h>

#include <conio.h>

*// deklarasi variable*

typedef struct

{

char nama[120];

long long NIM;

float IPK;

} Mahasiswa;

Mahasiswa list\_mhs[] = {

{"Foo", 11220910000111, 1.5},

{"Bar", 11220910000222, 3.1},

{"John", 11220910000333, 4},

{"Doe", 11220910000444, 2.5},

};

int input\_menu, i;

*// deklarasi prosedur*

void nim\_gt\_dua();

void nim\_highest\_lowest();

*// deklarasi algoritma*

int main()

{

while (true)

{

*// system("cls");*

*// meminta tipe query data mahasiswa*

printf("\nmasukkan tipe fungsi yang ingin di eksekusi !");

printf("\n1. Tampilkan mahasiswa dengan NIM > 2.0");

printf("\n2. Tampilkan 2 mahasiswa dengan NIM tertinggi dan terendah");

printf("\n3. Keluar dari program");

printf("\nInput Anda: ");

scanf("%i", &input\_menu);

*// mengecek input menu user*

if (input\_menu == 1)

nim\_gt\_dua();

else if (input\_menu == 2)

nim\_highest\_lowest();

else if (input\_menu == 3)

break;

else

continue;

}

*// keluar dari program*

printf("\nTerima kasih telah menggunakan program ini !");

return 0;

}

*// definisi fungsi*

void nim\_gt\_dua()

{

int len\_list = sizeof(list\_mhs) / sizeof(list\_mhs[0]);

int hasil\_pos\_kosong = 0;

Mahasiswa hasil[len\_list];

*// mengepush mahasiswa dengan ipk > 2 ke dalam array hasil*

for (i = 0; i < len\_list; i++)

{

if (list\_mhs[i].IPK > 2)

{

hasil[hasil\_pos\_kosong] = list\_mhs[i];

hasil\_pos\_kosong++;

}

}

*// mencetak hasil*

printf("\nMahasiswa dengan IPK > 2.0\n");

for (i = 0; i < hasil\_pos\_kosong; i++)

{

printf("%i. Nama: %s, NIM: %lli, IPK: %.2f\n", i + 1, hasil[i].nama, hasil[i].NIM, hasil[i].IPK);

}

printf("\nPress any button to continue !\n");

getche();

}

void nim\_highest\_lowest()

{

int len\_list = sizeof(list\_mhs) / sizeof(list\_mhs[0]);

Mahasiswa si\_paling\_rajin = list\_mhs[0]; *// mahasiswa dengan IPK tertinggi*

Mahasiswa si\_paling\_skip\_kelas = list\_mhs[0]; *// mahasiswa dengan IPK terendah*

*// meloop seisi array mahasiswa*

for (i = 0; i < len\_list; i++)

{

*// validasi IPK tertinggi*

if (list\_mhs[i].IPK > si\_paling\_rajin.IPK)

{

si\_paling\_rajin = list\_mhs[i];

}

*// validasi IPK terendah*

if (list\_mhs[i].IPK < si\_paling\_skip\_kelas.IPK)

{

si\_paling\_skip\_kelas = list\_mhs[i];

}

}

*// mencetak hasil*

printf("\nMahasiswa dengan IPK Tertinggi\n");

printf("1. Nama: %s, NIM: %lli, IPK: %.2f\n",

si\_paling\_rajin.nama, si\_paling\_rajin.NIM, si\_paling\_rajin.IPK);

printf("Mahasiswa dengan IPK Terendah\n");

printf("2. Nama: %s, NIM: %lli, IPK: %.2f\n",

si\_paling\_skip\_kelas.nama, si\_paling\_skip\_kelas.NIM, si\_paling\_skip\_kelas.IPK);

printf("\nPress any button to continue !\n");

getche();

}

Pseudocode No 17:

*/\* Program ini berisi element struct mahasiswa*

*1. membuat fungsi untuk mencetak data mahasiswa dengan IP > 2.0*

*2. membuat fungsi untuk mencetak nama dan nim mahasiswa dengan IPK tertinggi dan terendah*

*\*/*

*// deklarasi variable*

*structure Mahasiswa{*

*nama: char[120]*

*NIM: long long*

*IPK: float*

*}*

*var list\_mhs[]: Mahasiswa <- [*

*{"Foo", 11220910000111, 1.5},*

*{"Bar", 11220910000222, 3.1},*

*{"John", 11220910000333, 4},*

*{"Doe", 11220910000444, 2.5},*

*]*

*var (input\_menu, i): int*

*// deklarasi prosedur*

*fn nim\_gt\_dua(): void*

*fn nim\_highest\_lowest(): void*

*// deklarasi algoritma*

*fn main(): int do*

*while true do*

*// meminta tipe query data mahasiswa*

*write("masukkan tipe fungsi yang ingin di eksekusi !")*

*write("1. Tampilkan mahasiswa dengan NIM > 2.0")*

*write("2. Tampilkan 2 mahasiswa dengan NIM tertinggi dan terendah")*

*write("3. Keluar dari program")*

*input\_menu <- read("Input Anda: ")*

*// mengecek input menu user*

*if input\_menu == 1 do*

*nim\_gt\_dua()*

*else if input\_menu == 2 do*

*nim\_highest\_lowest()*

*else if input\_menu == 3 do*

*break*

*else do*

*continue*

*end if*

*end while*

*// keluar dari program*

*write("Terima kasih telah menggunakan program ini !")*

*return 0*

*end fn*

*// definisi fungsi*

*fn nim\_gt\_dua(): void do*

*var len\_list: int <- list\_mhs.length*

*var hasil\_pos\_kosong: int <- 0*

*var hasil: Mahasiswa[len\_list]*

*// mengepush mahasiswa dengan ipk > 2 ke dalam array hasil*

*for i <- 0 to len\_list do*

*if list\_mhs[i].IPK > 2 do*

*hasil[hasil\_pos\_kosong] <- list\_mhs[i]*

*hasil\_pos\_kosong++*

*end if*

*end for*

*// mencetak hasil*

*write("Mahasiswa dengan IPK > 2.0")*

*for i <- 0 to hasil\_pos\_kosong do*

*write("%i. Nama: %s, NIM: %lli, IPK: %.2f", i + 1, hasil[i].nama, hasil[i].NIM, hasil[i].IPK)*

*end*

*write("Press any button to continue !")*

*getche()*

*end fn*

*fn nim\_highest\_lowest(): void do*

*var len\_list: int <- list\_mhs.length*

*var si\_paling\_rajin: Mahasiswa <- list\_mhs[0] // mahasiswa dengan IPK tertinggi*

*var si\_paling\_skip\_kelas: Mahasiswa <- list\_mhs[0] // mahasiswa dengan IPK terendah*

*// meloop seisi array mahasiswa*

*for i = 0 to len\_list do*

*// validasi IPK tertinggi*

*if list\_mhs[i].IPK > si\_paling\_rajin.IPK do*

*si\_paling\_rajin <- list\_mhs[i]*

*end if*

*// validasi IPK terendah*

*if list\_mhs[i].IPK < si\_paling\_skip\_kelas.IPK do*

*si\_paling\_skip\_kelas <- list\_mhs[i]*

*end if*

*end fn*

*// mencetak hasil*

*write("Mahasiswa dengan IPK Tertinggi")*

*write("1. Nama: %s, NIM: %lli, IPK: %.2f",*

*si\_paling\_rajin.nama, si\_paling\_rajin.NIM, si\_paling\_rajin.IPK)*

*write("Mahasiswa dengan IPK Terendah")*

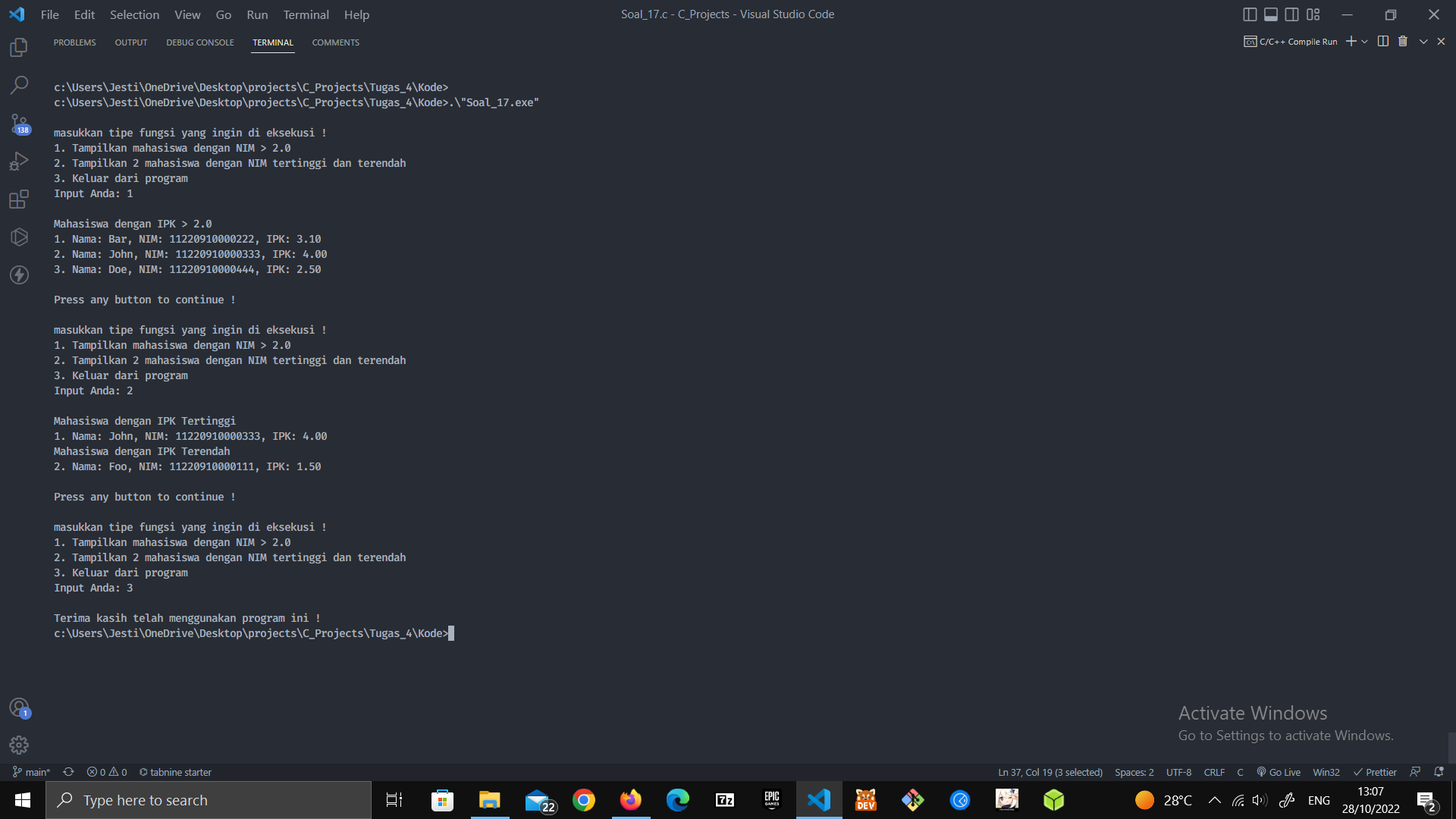
*write("2. Nama: %s, NIM: %lli, IPK: %.2f",*

*si\_paling\_skip\_kelas.nama, si\_paling\_skip\_kelas.NIM, si\_paling\_skip\_kelas.IPK)*

*write("Press any button to continue !")*

*getche()*

*end fn*



Screenshot Program No 17