



Modul PAP-05

Deklarasikan fungsi-fungsi di bawah ini ke dalam **pustaka.h**, serta buat realisasi fungsi tersebut pada file **pustaka.c**, kemudian buat uji cobalah semua fungsi dengan membuat program pemanggil pada file **main.c**.

Tugas 1:

void compress_array(int data[], int jml_data)

Fungsi ini akan menghilangkan semua index array yang bernilai 0 dan berada di tengah array yang bernilai lebih dari 0. Fungsi ini akan menggeser semua array yang berada di sebelah kanan index yang bernilai 0 sehingga tidak ada lagi nilai 0 di tengah array.

Deklarasi Array	Contoh pemanggilan fungsi
<pre>int arr1[] = { 3, 5, 38, 0, 44, 47 }; int arr2[] = { 3, 0, 44, 38, 0, 5, 47 }; int arr3[] = { 2, 15, 0, 0, 0, 26, 27, 36 }; int arr4[] = { 15, 36, 27, 0, 0, 2, 26 };</pre>	<pre>compress_array (arr1, 6) // -> arr1[] = { 3, 5, 38, 44, 47 } compress_array (arr2, 7) // -> arr2[] = { 3, 44, 38, 5, 47 } compress_array (arr3, 8) // -> arr3[] = { 2, 15, 26, 27, 36 } - compress_array (arr4, 7) - // -> arr4[] = { 15, 36, 27, 2, 26 }</pre>

Tugas 2:

void selection_sort1(int data[], int jml_data)

Fungsi ini akan mengurutkan larik integer yang diberikan pada parameter data dan mencetak ke layar setiap iterasi pengurutan yang dilakukan.

Pada fungsi ini algoritma pengurutan yang digunakan adalah selection sort tanpa menggunakan pengecekan urut. Sehingga meskipun di tengah iterasi data telah terurut, fungsi ini akan tetap melanjutkan proses hingga semua iterasi selesai dijalankan.

Deklarasi Array	Contoh pemanggilan fungsi
<pre>int arr1[] = { 3, 44, 38, 5, 47 }; int arr2[] = { 15, 36, 27, 2, 26 };</pre>	<pre>- selection_sort1(arr1, 5) // -> {3, 5, 38, 44, 47} 3 44 38 5 47 3 5 38 44 47 3 5 38 44 47 3 5 38 44 47 3 5 38 44 47 - selection_sort1(arr2, 5) // -> {2, 15, 26, 27, 36} 2 36 27 15 26 2 15 27 36 26</pre>



Praktikum Algoritma dan Pemrograman

Modul Praktikum PAP-05

Laboratorium Dasar FIK - UDINUS

	<pre>2 15 26 36 27 2 15 26 27 36 2 15 26 27 36</pre>
Tugas 3:	
void selection_sort2(int data[], int jml_data)	
<p>Fungsi ini akan mengurutkan larik integer yang diberikan pada parameter data dan mencetak ke layar setiap iterasi pengurutan yang dilakukan.</p> <p>Pada fungsi ini algoritma pengurutan yang digunakan adalah selection sort dengan menggunakan pengecekan urut. Sehingga jika di tengah iterasi data telah terurut, maka fungsi akan berhenti tanpa menyelesaikan iterasi lagi.</p>	
Deklarasi Array	Contoh pemanggilan fungsi
<pre>int arr1[] = { 3, 44, 38, 5, 47 }; int arr2[] = { 15, 36, 27, 2, 26 };</pre>	<pre>- selection_sort2(arr1, 5) // -> {3, 5, 38, 44, 47} 3 44 38 5 47 3 5 38 44 47 - selection_sort2(arr2, 5) // -> {2, 15, 26, 27, 36} 2 36 27 15 26 2 15 27 36 26 2 15 26 36 27 2 15 26 27 36</pre>
Tugas 4:	
int get_median(int data[], int jml_data)	
<p>Fungsi ini akan mengembalikan nilai median dari sebuah larik yang diberikan. Jika larik pada parameter belum terurut, maka fungsi ini akan melakukan pengurutan terlebih dahulu sebelum melakukan proses pencarian median.</p>	
Deklarasi Array	Contoh pemanggilan fungsi
<pre>int arr1[] = { 3, 44, 38, 5, 47 }; int arr2[] = { 15, 36, 27, 2, 26 };</pre>	<pre>- get_median(arr1, 5) // -> 38 - get_median(arr2, 5) // -> 26</pre>