



Praktikum Algoritma dan

Pemrograman Modul Praktikum PAP-

06 Laboratorium Dasar FIK - UDINUS

Modul PAP-06

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:	
int binary_search(int angka, int data[], int jml_data)	
Fungsi ini akan mengembalikan nilai true jika nilai yang dicari melalui parameter angka ada pada parameter data, dan akan mengembalikan nilai false jika angka tidak ditemukan. Algoritma pencarian yang digunakan adalah binary search.	
Contoh pemanggilan fungsi	Contoh pemanggilan fungsi
list1 [] = {1, 2, 3, 4, 5, 6, 7, 8, 9, 10} - binary_search(3, list1, 10) // -> 1	list2 [] = {4, 8, 6, 1, 10, 3, 9, 2, 7, 5} - binary_search(8, list2, 10) // -> 0
1,2,3,4,5,6,7,8,9,10, 1,2,3,4, 3,4, 3	1,2,3,4,5,6,7,8,9,10, 6,7,8,9,10, 6,7, 7, 7
Tugas 2:	
int search_word(char word[], char text[])	
Fungsi ini akan mengembalikan nilai true jika string pada parameter word, terdapat pada string text, dan mengembalikan nilai false jika word tidak terdapat pada text.	
Clue: fungsi akan menghitung panjang karakter dari parameter word. Kemudian secara berulang memotong karakter sejumlah panjang string word untuk dibandingkan dengan string text. Perbandingan dilakukan dengan menggeser karakter demi karakter hingga mendapatkan kata yang diinginkan, atau hingga akhir string.	
Contoh pemanggilan fungsi	
- search_word("Dian", "Universitas Dian Nuswantoro"); -> true - search_word("Algo", "Algoritma"); -> true - search_word("Program", "Pemrograman"); -> false - search_word("Dinus", "Dian Nuswantoro"); -> false	
Tugas 3:	
int is_anagram(char text1[], char text2[])	
Fungsi ini akan mengembalikan nilai true jika string pada parameter text2 merupakan anagram dari string pada parameter text1, dan akan mengembalikan nilai false jika tidak.	



Praktikum Algoritma dan Pemrograman Modul Praktikum PAP-06 Laboratorium Dasar FIK - UDINUS

- Anagram adalah teks/kata/kalimat yang dibentuk dengan merubah posisi huruf dari teks/kata/kalimat lain tanpa menambahkan atau mengurangi jumlah huruf. Contoh: “reactive” merupakan anagram dari “creative”.
- *Clue*: kita bisa menentukan sebuah teks1 merupakan anagram dari teks2 jika frekuensi huruf yang pada teks1 sama persis dengan frekuensi huruf yang muncul pada teks2.

Contoh pemanggilan fungsi

- `is_anagram("the eyes", "they see"); // -> 1`
- `is_anagram("astronomer", "moon starrer"); // -> 1`
- `is_anagram("udinus", "dian nuswantoro"); // -> 0`
- `is_anagram("columbia", "australia"); // -> 1`