

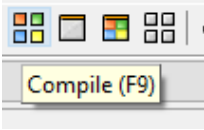


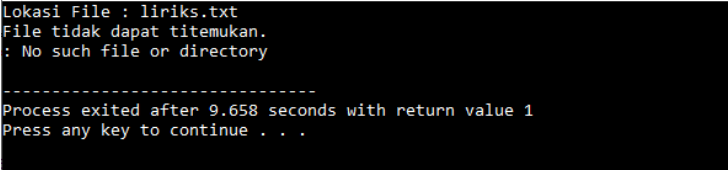
DOKUMEN JOB SHEET(JS)		 POLBAN
No. 37/Doc-Akad/Polban/V/2019		
POLITEKNIK NEGERI BANDUNG		
Program Studi	:	D-IV Teknik Informatika
Mata Kuliah	:	Dasar-Dasar Pemrograman
Kegiatan/Keterampilan	:	Pembuatan aplikasi untuk mencari kata pada file bertipe text
Unit	:	KBK Rekayasa Perangkat Lunak
Objektif Perilaku Siswa (OPS)	:	Mahasiswa dapat membuat program untuk mencari kata pada file bertipe text menggunakan Bahasa Pemrograman C tanpa menghasilkan <i>syntax error</i> , <i>compile error</i> dan <i>runtime error</i> .
Dosen	:	Ani Rahmani, S.Si, MT/ Lukmannul Hakim Firdaus, S.Kom, M.T
TAHAP PERSIAPAN		
Bahan	:	Template File Project (.c), Text Input (.txt)
Peralatan & Perlengkapan	:	Komputer/Laptop, proyektor, IDE DevC++
Referensi	:	<ul style="list-style-type: none"><li>Goodrich, M, 2011. Data Structure &amp; Algorithms, John Wiley&amp;Sons,Inc.</li><li><a href="https://www.tutorialspoint.com/cprogramming/c_file_io.htm">https://www.tutorialspoint.com/cprogramming/c_file_io.htm</a></li></ul>
Dasar Teori	:	<p>Pengolahan terhadap data dalam format teks.</p> <ol style="list-style-type: none"><li>Definisi karakter sebagai akhir dari teks (EOF)</li><li>variabel sebagai penunjuk karakter pada setiap proses pembacaan</li><li>Open File Teks</li><li>Read File</li><li>Text Processing Close File</li></ol>
Petunjuk Bagi Mahasiswa	:	<ul style="list-style-type: none"><li>Siapkan alat dan bahan yang diperlukan untuk praktikum (Download bahan praktikum yang terdapat di moodle).</li><li>Baca dan pelajari jobsheet</li><li>Ikuti petunjuk dari instruktur/dosen</li><li>Tanyakan pada instruktur/dosen hal-hal yang belum dimengerti</li></ul> <p>Laporkan hasil kerja/source code setelah selesai melakukan latihan</p>
Keselamatan Kerja	:	<p>Tidak menyentuh sumber listrik secara langsung</p> <ul style="list-style-type: none"><li>Tidak membawa Thumblers/ Air kemasan ke dalam lab</li></ul>

PROSEDUR PELAKSANAAN		
NO	LANGKAH Pengerjaan dan KEY POINT	ILUSTRASI GAMBAR
1.	Unduh bahan praktikum yang terdapat di moodle dan simpanlah pada drive di laptop. (Key Point : Simpan file lirik.txt satu root dengan file tmp_project.c)	
2.	Buka Aplikasi DevC++	
4.	Buka File "tmp_project.c"	
5.	Pelajari struktur project yang akan menjadi template pembuatan program  (Key Point : bedakan penggunaan function dan prosedur)	<pre> 1  #include&lt;stdio.h&gt; 2  #include&lt;stdlib.h&gt; 3 4  FILE* open_file(char file_name[], char mode[]){ 5      FILE *fp = fopen(file_name, mode); 6 7      if(fp == NULL){ 8          perror("File tidak dapat ditemukan.\n"); 9          exit(EXIT_FAILURE); 10     } 11     return fp; 12 } 13 14 void read_text(FILE* fp){ 15     char ch[512]; 16     printf("Isi file:\n"); 17     while((fgets(ch, 512, fp)) != NULL){ 18         printf("%s", ch); 19     } 20 } 21 22 int main(){ 23 24     return 0; 25 } </pre>
6.	Pada method main, Tulislah baris kode untuk membuka file teks dan menampilkan isinya pada layar.  (Key Point : Penulisan tipe data FILE harus kapital dan berjenis pointer. Setiap file yang sudah di open harus di close)	<pre> 46 int main(){ 47     FILE* file; 48     char filename[30], keyword[20]; 49 50     printf("Lokasi File : "); 51     scanf("%s", &amp;filename); 52 53     file = open_file(filename, "r"); 54     read_text(file); 55 56     fclose(file); </pre>

7.	<p>Pada method main, Tulislah baris kode untuk input pencarian kata.</p> <p><b>(Key Point : Sisipkan open_file() sebelum input pencarian kata)</b></p>	<pre>58 file = open_file(filename, "r"); 59 printf("\nCari Kata : "); 60 scanf("%s", &amp;keyword); 61</pre>
8.	<p>Buatlah procedure dengan nama <b>search_text</b> untuk pencarian kata pada file, yang memiliki parameter input file bertipe FILE* dan keyword bertipe char[].</p> <p><b>(Key Point : Penulisan tipe data FILE harus kapital dan berjenis pointer)</b></p>	<pre>24 void search_text(FILE* f, char key[]){ 25 26 }</pre>
9.	<p>Buatlah <i>looping</i> untuk pencarian kata pada file</p> <p><b>(Key Point : Jangan lupa tambahkan "\n" untuk setiap output yang ditampilkan)</b></p>	<pre>25 char text[512]; 26 27 printf("\nMencari Kata '%s'", key); 28 29 int baris = 1, ditemukan = 0; 30 31 while(fgets(text, 512, f) != NULL){ 32     if(strstr(text, key)){ 33         printf("\nKata %s ditemukan pada baris ke-%d : ", key, baris); 34         printf("\n%s", text); 35         ditemukan++; 36     } 37     baris++; 38 } 39</pre>
10.	<p>Tambahkan library String.h pada bagian atas baris program</p>	<pre>1 #include&lt;stdio.h&gt; 2 #include&lt;stdlib.h&gt; 3 #include&lt;string.h&gt;</pre>
11.	<p>Buatlah baris kode untuk menampilkan pesan jika kata yang dicari tidak ditemukan</p>	<pre>40 if(ditemukan == 0){ 41     printf("\nKata %s tidak ditemukan", key); 42 } 43</pre>
12.	<p>Buatlah pemanggilan procedure search_text() pada method main</p> <p><b>(Key Point : Penulisan nama procedure harus sesuai dengan yang dibuat. Setiap file yang sudah di open harus di close)</b></p>	<pre>62 search_text(file, keyword); 63 64 fclose(file);</pre>

13	<p>Lakukan pengecekan keseluruhan baris kode.</p> <p>(Key Point : Cek seluruh baris kode sebelum di compile, cek titik koma(;), kurung kurawal ({}), dan penamaan variable jangan sampai keliru)</p>	<pre> 1  #include&lt;stdio.h&gt; 2  #include&lt;stdlib.h&gt; 3  #include&lt;string.h&gt; 4 5  FILE* open_file(char file_name[], char mode[]){ 6      FILE *fp = fopen(file_name, mode); 7 8      if(fp == NULL){ 9          perror("File tidak dapat ditemukan.\n"); 10         exit(EXIT_FAILURE); 11     } 12 13     return fp; 14 } 15 16 void read_text(FILE* fp){ 17     char ch[512]; 18     printf("Isi file:\n"); 19     while((fgetc(ch, 512, fp)) != NULL){ 20         printf("%s", ch); 21     } 22 } 23 24 void search_text(FILE* f, char key[]){ 25     char text[512]; 26 27     printf("\nMencari Kata '%s'", key); 28 29     int baris = 1, ditemukan = 0; 30 31     while(fgetc(text, 512, f) != NULL){ 32         if(strstr(text, key)){ 33             printf("\nKata %s ditemukan pada baris ke-%d : ", key, baris); 34             printf("\n%s", text); 35             ditemukan++; 36         } 37         baris++; 38     } 39 40     if(ditemukan == 0){ 41         printf("\nKata %s tidak ditemukan", key); 42     } 43 } 44 45 int main(){ 46     FILE* file; 47     char filename[30], keyword[20]; 48 49     printf("Lokasi File : "); 50     scanf("%s", &amp;filename); 51 52     file = open_file(filename, "r"); 53     read_text(file); 54 55     fclose(file); 56 57     file = open_file(filename, "r"); 58     printf("\nCari Kata : "); 59     scanf("%s", &amp;keyword); 60 61     search_text(file, keyword); 62 63     fclose(file); 64     return 0; 65 } </pre>
14.	<p>Lakukan Compile Program (F9).</p> <p>(Key Point : Pastikan tidak terjadi compile error)</p>	

15.	<p>Lakukan Run Program (F10). Jika berhasil maka akan muncul</p>	
16.	<p>Lakukan Uji coba program yang telah dibuat Inputkan path file yang akan dibuka (Path Valid) <b>Ketik : lirik.txt</b></p> <p>Input kata yang ada pada teks. <b>Ketik : tiba</b></p> <p>Input kata yang tidak ada pada teks. <b>Ketik : rajin</b></p>	  

	<p>Inputkan path file yang akan dibuka (Path Tidak Valid)  <b>Ketik : liriks.txt</b></p>	
<b>EVALUASI</b>		
<ol style="list-style-type: none"> <li>1. Presentasikan langkah-langkah yang sudah dilakukan untuk pembuatan program mencari kata pada file text didepan dosen/kelas.</li> <li>2. Lakukan Pengujian aplikasi <ol style="list-style-type: none"> <li>a) Jika path yang diinputkan valid, maka program akan menampilkan isi file pada layar.</li> <li>b) Jika path yang diinputkan tidak valid, maka program akan menampilkan pesan "File tidak ditemukan".</li> <li>c) Jika kata yang dicari ditemukan, maka program akan menampilkan dibaris berapa saja kata ditemukan</li> <li>d) Jika kata yang dicari tidak ditemukan, maka program akan menampilkan pesan "Kata tidak ditemukan"</li> </ol> </li> </ol>		
<b>CATATAAN</b>		