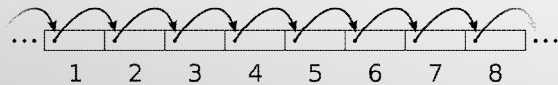
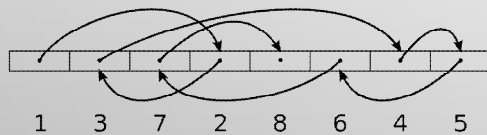


SEQUENTIAL FILE (MEMBUAT & MEMBACA ARSIP BERUNTUN)

Sequential access



Random access



MEMBUAT ARSIP BERUNTUN

- MEMBUAT ARSIP BERUNTUN HANYA MEMBUTUHKAN BEBERAPA DATA YANG DIBUTUHKAN DAN MENENTUKAN REKAMAN *DUMMY*/MARK YANG DIGUNAKAN SEBAGAI PENANDA AKHIR REKAMAN PADA ARSIP BERUNTUN.
- HAL-HAL YANG PERLU DIPERHATIKAN UNTUK MEMBUAT SEBUAH ARSIP BERUNTUN ADALAH SEBAGAI BERIKUT :
 1. MENULISKAN REKAMAN YANG INGIN DITULIS.
 2. MENULIS REKAMAN DUMMY SEBAGAI AKHIR PEMBACAAN REKAMAN

DEKLARASI SEBUAH ARSIP BERUNTUN

Bahasa Algoritmik	Bahasa Pascal	Bahasa C
<pre> type nama_tipe : < nama_struktur1 : tipe_data, nama_struktur2 : tipe_data > nama_variabel_file : SeqFile of tipe_data nama_variabel_rekaman :</pre>	<pre> type nama_tipe = record nama_struktur1 : tipe_data; nama_struktur2 : tipe_data; end; var nama_variabel_file : text; nama_variabel_rekaman :</pre>	<pre> typedef struct{ tipe_data nama_struktur1; tipe_data nama_struktur2; }nama_tipe; FILE *nama_variabel_file; nama_tipe</pre>

DEKLARASI SEBUAH ARSIP BERUNTUN

Bahasa Algoritmik	Bahasa Pascal	Bahasa C
<pre>nama_tipe nama_variabel_file <- OPEN(nama_file , kode_izin_akses) WRITE(nama_variabel_fil e, string_yang_ditulisakan) READ(nama_variabel_file , variabel_penyimpan_isi_ file) CLOSE(arsipMatKul)</pre>	<pre>nama_tipe; begin assign(nama_variabel_fi le, nama_file); rewrite(nama_variabel_f ile); {atau reset(nama_variabel_fil e);} write(nama_variabel_fil e, string_yang_ditulisakan) ; read(nama_variabel_file , variabel_penyimpan_isi_ file); close(nama_variabel_fil e); end.</pre>	<pre>nama_variabel_rekaman; nama_variabel_file = fopen(nama_file, kode_izin_akses); fprintf(nama__variabel_f ile, string_yang_ditulisakan); fscanf(nama_variabel_fil e, string_pemesanan_tempat, &variabel_penyimpan_isi_ file); fclose(nama_variabel_fil e);</pre>

DEKLARASI SEBUAH ARSIP BERUNTUN

<pre>type nilaiMatKul : < nim : <u>string</u>, nama : <u>string</u>, nilai : <u>string</u> > arsipMatKul : <u>SeqFile</u> of nilaiMatKul rekaman : nilaiMatKul arsipMatKul <- <u>OPEN</u>("ArsipMatKul.dat", "w") <u>WRITE</u>(arsipMatKul, rekaman.nim, " ", rekaman.nama, " ",</pre>	<pre>type nilaiMatKul = record nim : string; nama : string; nilai : string; end; var arsipMatKul : text; rekaman : nilaiMatKul; begin assign(arsipMatKul, 'ArsipMatKul.dat'); rewrite(arsipMatKul); {atau reset(arsipMatKul);} write(arsipMatKul, rekaman.nim + ' ' + rekaman.nama + ' ' +</pre>	<pre>typedef struct{ char nim[10]; char nama[100]; char nilai[2]; } nilaiMatKul; FILE *arsipMatKul; nilaiMatKul rekaman; arsipMatKul <- fopen("ArsipMatKul.dat", "w"); fprintf(arsipMatKul, "%s %s %s\n", rekaman.nim, rekaman.nama,</pre>
---	--	--

DEKLARASI SEBUAH ARSIP BERUNTUN

Bahasa Algoritmik	Bahasa Pascal	Bahasa C
rekaman.nilai)	rekaman.nilai);	rekaman.nilai);
<u>READ</u> (arsipMatKul, rekaman.nim, rekaman.nama, rekaman.nilai)	read(arsipMatKul, rekaman.nim);	fscanf(arsipMatKul, "%s %s %s\n", &rekaman.nim, &rekaman.nama, &rekaman.nilai);
<u>CLOSE</u> (arsipMatKul)	close(arsipMatKul); end.	fclose(arsipMatKul);

- BERIKUT ADALAH ALGORITMA ARSIP BERUNTUN YANG BERISI REKAMAN NILAI MATA KULIAH YANG TERDIRI DARI: NOMOR INDUK, NAMA, DAN NILAI MATA KULIAH.

ALGORITMA ARSIP BERUNTUN BERISI REKAMAN

Bahasa Manusia	Bahasa Algoritmik
Mendeklarasikan struktur tipe rekaman, variabel rekaman, variabel yang akan mengacu pada file dan membuka file dengan kode akses penulisan	<pre>type nilaiMatKul : < nim : string, nama : string, nilai : string > rekaman : nilaiMatKul arsipMatKul : SeqFile of nilaiMatKul arsipMatKul <- OPEN("ArsipMatKul.dat", "w")</pre>
Menuliskan ke layar untuk menuliskan rekaman satu per satu	<pre>output("Tuliskan rekaman satu per satu : ")</pre>
membaca nilai rekaman pertama sebagai inisialisasi dari perulangan memasukkan rekaman	<pre>output("masukkan nim : ") input(rekaman.nim)</pre>
membuat proses perulangan untuk memasukkan rekaman satu per satu	<pre>while rekaman.nim <> "XXXXXXXX" do {proses}</pre>

ALGORITMA ARSIP BERUNTUN BERISI REKAMAN

Bahasa Manusia	Bahasa Algoritmik
dan menuliskannya ke file sampai ditemui rekaman <i>dummy</i>	<pre>output("masukkan nama : ") input(rekaman.nama) output("masukkan nilai : ") input(rekaman.nilai) {menuliskan ke file} WRITE(arsipMatKul, rekaman.nim, " ", rekaman.nama, " ", rekaman.nilai) {iterasi} output("masukkan nim : ") input(rekaman.nim) {end while}</pre>
Menuliskan rekaman <i>dummy</i>	<pre>WRITE(arsipMatKul, "XXXXXXXX XXX X")</pre>
Menutup file	<pre>CLOSE(arsipMatKul)</pre>

MEMBACA ARSIP BERUNTUN

- KASUS-KASUS YANG HARUS DIPERHATIKAN PADA SAAT PEMBACAAN SEBUAH ARSIP BERUNTUN ADALAH SEBAGAI BERIKUT :

1. ARSIP KOSONG

ADA KEMUNGKINAN SEBUAH ARSIP BERUNTUN TIDAK BERISI REKAMAN DATA DAN HANYA BERISI REKAMAN DUMMY SEBAGAI TANDA AKHIR MAKA ARSIP BERUNTUN DIANGGAP SEBAGAI ARSIP KOSONG.

CONTOHNYA:

XXXXXXXX XXX X

2. ARSIP BERISI REKAMAN

DIPERLUKAN PROSES UNTUK MEMBACA ARSIP BERUNTUN SAMPAI REKAMAN YANG TERAKHIR.

MEMBACA ARSIP BERUNTUN TANPA PROSES PENCARIAN

- ALGORITMA UNTUK MEMBACA SEBUAH ARSIP BERUNTUN YANG BERISI REKAMAN NILAI MATA KULIAH YANG TERDIRI DARI NOMOR INDUK, NAMA, DAN NILAI MATA KULIAH YANG TELAH DIBUAT:

Bahasa Manusia	Bahasa Algoritmik
Mendeklarasikan struktur tipe rekaman, variabel rekaman, variabel yang akan mengacu pada file dan membuka file dengan kode akses penulisan	<pre><u>type</u> nilaiMatKul : < nim : <u>string</u>, nama : <u>string</u>, nilai : <u>string</u> > rekaman : nilaiMatKul arsipMatKul : <u>SeqFile of</u> nilaiMatKul arsipMatKul <- <u>OPEN</u>("ArsipMatKul.dat", "r")</pre>
Menuliskan ke layar untuk menampilkan rekaman satu per satu	<pre><u>output</u>("isi arsip beruntun adalah : ")</pre>

MEMBACA ARSIP BERUNTUN TANPA PROSES PENCARIAN

Menuliskan ke layar untuk menampilkan rekaman satu per satu	<pre><u>output</u>("isi arsip beruntun adalah : ")</pre>
Membaca nilai rekaman pertama dari file	<pre><u>READ</u>(arsipMatKul, rekaman.nim, rekaman.nama, rekaman.nilai)</pre>
Membuat penanganan jika arsip beruntun merupakan arsip kosong	<pre><u>if</u> rekaman.nim = "XXXXXXXX" <u>then</u> <u>output</u>("arsip kosong") <u>end if</u></pre>
Membuat proses perulangan untuk membaca rekaman satu per satu dan menampilkannya ke layar sampai ditemui rekaman <i>dummy</i>	<pre><u>else</u> <u>repeat</u> {proses} <u>output</u>("nim : ", rekaman.nim) <u>output</u>("nama : ", rekaman.nama) <u>output</u>("nilai : ", rekaman.nilai) <u>output</u>("-----") {iterasi} <u>READ</u>(arsipMatKul, rekaman.nim, rekaman.nama, rekaman.nilai) <u>until</u> rekaman.nim = "XXXXXXXX" <u>end else</u></pre>
Menutup file	<pre><u>CLOSE</u>(arsipMatKul)</pre>

MEMBACA ARSIP BERUNTUN DENGAN PENCARIAN

- ALGORITMA UNTUK MEMBACA SEBUAH ARSIP BERUNTUN YANG BERISI REKAMAN NILAI MATA KULIAH YANG TERDIRI DARI NOMOR INDUK, NAMA, DAN NILAI MATA KULIAH YANG TELAH DIBUAT SEBELUMNYA DENGAN PENCARIAN BERDASARKAN NOMOR INDUK:

Bahasa Manusia	Bahasa Algoritmik
Mendeklarasikan struktur tipe rekaman, variabel rekaman, variabel yang akan mengacu pada file dan membuka file dengan kode akses penulisan	<pre>type nilaiMatKul : < nim : <u>string</u>, nama : <u>string</u>, nilai : <u>string</u> > rekaman : nilaiMatKul arsipMatKul : <u>SeqFile of</u> nilaiMatKul arsipMatKul <- <u>OPEN</u>("ArsipMatKul.dat", "r")</pre>
Mendeklarasikan variabel string yang menyimpan kata yang dicari dan menuliskan ke layar untuk memasukkan kata kunci nama yang akan dicari	<pre>kata_cari : <u>string</u> <u>output</u>("masukkan nomor induk yang dicari : ") <u>input</u>(kata_cari)</pre>

MEMBACA ARSIP BERUNTUN DENGAN PENCARIAN

Membaca nilai rekaman pertama dari file	<pre>READ(arsipMatKul, rekaman.nim, rekaman.nama, rekaman.nilai)</pre>
Membuat penanganan jika arsip beruntun merupakan arsip kosong	<pre>if rekaman.nim = "XXXXXXXX" then output("arsip kosong"); {end if}</pre>
Membuat penanganan jika masukan adalah nilai nomor induk pada rekaman <i>dummy</i> yaitu "XXXXXXXX", karena rekaman ini sebenarnya hanya sebagai penambah maka rekaman dengan nomor induk "XXXXXXXX" dianggap tidak ada	<pre>else if kata_kunci = "XXXXXXXX" then output("tidak ditemukan") {end if}</pre>
Membuat proses perulangan untuk membaca rekaman satu per satu dan mencari nama yang dicari sampai ditemui rekaman <i>dummy</i> atau nama yang dicari telah	<pre>else while (rekaman.nim <> "XXXXXXXX") and (rekaman.nim <> kata_cari) do {iterasi} READ(arsipMatKul, rekaman.nim, rekaman.nama, rekaman.nilai) {end while}</pre>

MEMBACA ARSIP BERUNTUN DENGAN PENCARIAN

Bahasa Manusia	Bahasa Algoritmik
ditemukan	<pre>if rekaman.nim = kata_cari then output("hasil pencarian : ") output("nim : ", rekaman.nim) output("nama : ", rekaman.nama) output("nilai : ", rekaman.nilai) output("-----") {end if} else output("tidak ditemukan") {end else} {end else}</pre>
Menutup file	<pre>CLOSE(arsipMatKul)</pre>



IMPLEMENTASI KEDALAM BAHASA PEMROGRAMAN C++