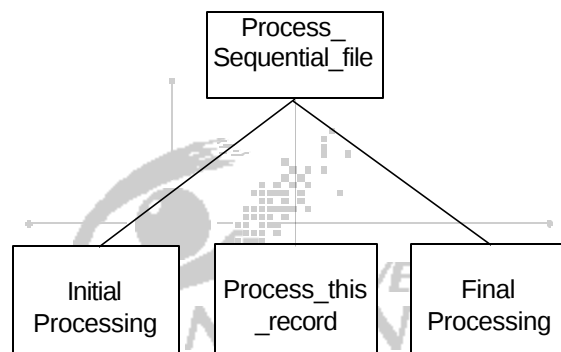


ALGORITMA UNTUK PERSOALAN BISNIS

Tujuan :

- Memperkenalkan algoritma untuk memecahkan masalah bisnis.
- Merancang algoritma untuk memecahkan persoalan bisnis.

Algoritma di atas digambarkan dengan Struktur Chart sbb :



Tiga modul di atas dapat berisi statement yang dibutuhkan untuk memproses data yang dibaca dari file.

Main program dari struktur chart sudah ada sehingga dapat dikembangkan proses dari ketiga modul tersebut.

Struktur Program

Secara umum program yang dilakukan untuk memproses data yang dibaca dari file dapat digambarkan sbb :

Process_sequential_file

Initial processing

Read first record

DO WHILE NOT EOF

Process this record

Read next record

ENDDO

Final processing

END.

Membuat laporan dengan Page Break

[Suatu laporan membutuhkan

- ↳ Judul
- ↳ Baris Judul
- ↳ Kolom Judul
- ↳ Detail dan
- ↳ Total

[Di samping itu suatu halaman laporan hanya menampung n detail, sehingga harus mengatur pemindahan ke halaman baru.

[Untuk itu dibutuhkan konter baris untuk mengecek apakah harus pindah ke halaman baru.

SUATU CONTOH LAPORAN

GLAD RAGS CLOTHING Co.

16/02/96 CURRENT ACCOUNT PAGE 1
BALANCE

CUSTOMER NUMBER	CUSTOMER NAME	CUSTOMER ADDRES	ACCOUNT BALANCE
12345	Sporty's Boutique	The Mall Redfer	300,50
12346	JC Penney	Anggrek Mall	400,50
Total Cust on file			200,00
Total Cust with balance Owning			150,00
Total balance Owning			4.300,00

Algoritma Pemecahan

Main Program

```

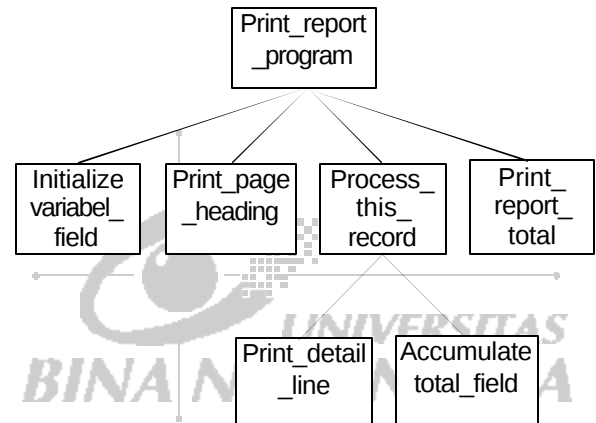
Print_report_program
  Initialize_variable_fields
  Print_page_headings
  Read first record
  DO WHILE NOT EOF
    IF line_count > max_detail_lines THEN
      Print_page_headings
      Set line_count to zero
    ENDIF
    Process_this_record
    Read next record
  ENDDO
  Print_report_totals
END

```

Subordinate Modul

1. Initilize_variable_fields
 - Open file
 - Set accumulators to zero
 - Set page_count to zero
 - Set line_count to zero
 - Set max_detail_lines to designated value
 END.

A. Hierarchy Chart / Struktur Chart



2. Print_page_headings
 - Increment pagecount
 - print heading lines
 - print column heading lines
 - print blank line (if required)
 END.
3. Process_this_record
 - perform necessary calculations (if any)
 - print_detail_line
 - accumulate_total_fields
 END.
4. Print_detail_line
 - build detail line
 - print detail line
 - increment linecount
 END.
5. Accumulate_total_fields
 - increment accumulators are required
 END.
6. Print_report_totals
 - build total line(s)
 - print total line(s)
 - close file
 END.

Single Level Control Break

Diminta membuat laporan dari data file penjualan untuk salesman dengan laporan sbb:

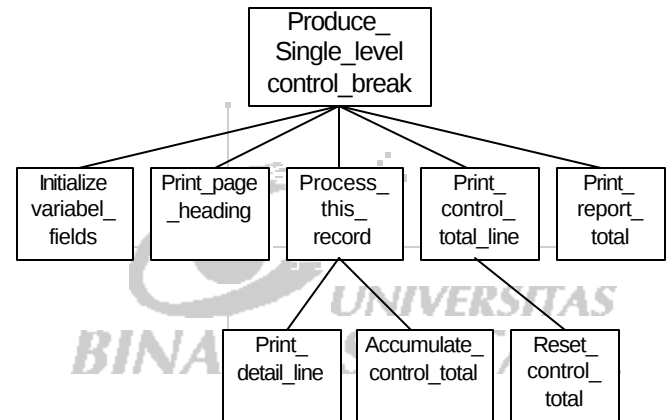
MULTI DISK COMPUTER Co. SALES					
21/03/97		REPORT BY SALES PERSON			PAGE 1
SALES PERSON NUMBER	SALES PERSON NAME	NUMBER	QTY SOLD	PRICE	EXTENTION AMOUNT
1001	Mary Smith	1032	2	10.00	20.00
		1033	2	20.00	40.00
		1044	2	30.00	60.00
		Total for Mary Smith			120.00
1002	Jane Brown	1032	2	10.00	20.00
		1045	1	35.00	35.00
		Total for Jane Brown			55.00
Total Sales					175.00

B.Algoritma**Main Line**

```

Produce_single_level_control_break
  Initialize_variabel_fields
  Print_page_headings
  Read first record
  this_control_field = control_field
  prev_control_field = control_field
  DO WHILE NOT EOF
    IF this_control_field NOT =
      prev_control_field THEN
      print_control_total_line
      prev_control_field =
        this_control_field
    ENDIF
    IF Linecount > max_detail_lines
    THEN
      Print_page_headings
      Set linecount to zero
    ENDIF
    Process_this_record
    Read next record
    this_control_field = control field
  ENDDO
  Print_control_total_line
  Print_report_totals
END.

```

A.Hierarchy Chart**Subordinate Modul**

1. Initialise_variable_fields
Open file sales
Set control total accumulators to zero
Set report total accumulators to zero
Set page_count to zero
Set line_count to zero
Set max_detail_lines to designated value
END.
2. Print_page_headings
Increment page_count
Print heading lines
Print column heading lines
Print blank line
END.
3. Process_this_record
Perform necessary calculation (if any)
Print_detail_line
Accumulate_control_totals
END.

4. Print_control_total_line
 Build control total line
 Print control total line
 Print blank line (if required)
 Increment line_count
 Reset_control_totals
END.
5. Print_report_totals
 Build report total line
 Print report total line
END.
6. Print_detail_line
 Build detail line
 Print detail line
 Increment line_count
END.
7. Accumulate_control_totals
 Increment control total accumulators
END.
8. Reset_control_totals
 Add control total accumulators to report
 total accumulator
 Set control total accumulators to zero
END.