

Operasi Aritmatika

STATEMENT ARITMATIKA & COMPUTE

Kemampuan untuk mengadakan perhitungan pada program COBOL tidak begitu penting dibandingkan dgn bahasa program yang lain seperti FORTRAN, BASIC, ALGOL atau PL/I sehingga kemampuannya tidak begitu luwes dibandingkan bahasa-bahasa tersebut.

COBOL mempunyai 4 verb dasar utk membuat statement operasi ARITMATIKA :

- a. ADD (penjumlahan)
- b. SUBTRACT (pengurangan)
- c. MULTIPLY (perkalian)
- d. DIVIDE (pembagian)

Keempat verb dasar tersebut dilengkapi dgn verb untuk perhitungan yg lebih rumit: COMPUTE verb.

ADD Statement

Digunakan utk menjumlahkan 2 atau lebih operand numerik, dan menyimpan hasilnya.

Ada 3 bentuk statement ADD yaitu :

BENTUK 1

ADD nama-data-1 nama-data-2
 literal-1 literal-2 **TO** nama-data-m [**ROUNDED**]
 [; ON **SIZE ERROR** statement imperative]

BENTUK 2

ADD nama-data-1 nama-data-2 nama-data-3
 literal-1 literal-2 literal-3 **GIVING** nama-data-m [**ROUNDED**]
 [; ON **SIZE ERROR** statement imperative]

BENTUK 3

ADD CORRESPONDING
 CORR identifier-i TO identifier-e [**ROUNDED**]
 [; ON **SIZE ERROR** statement imperative]

PERBEDAAN : **BENTUK TO**

apabila beberapa nilai akan dijumlahkan & salah satunya diisi dgn hasil penjumlahan.

Contoh : **8 12**

ADD A TO B

Misal pada awalnya A mempunyai nilai 20

B mempunyai nilai 4

Maka setelah dilakukan operasi diatas, nilai A tetap 20, nilai B menjadi 24

BENTUK GIVING

Apabila beberapa nilai akan dijumlahkan, dan hasilnya akan disimpan dalam field data item yg lain.

Contoh : **8 12**

ADD A, B GIVING C

Misal pada awalnya A=6 B=4 C=2

Maka setelah dilakukan operasi diatas, nilai A tetap 6, B tetap 4 dan C menjadi 10

KETERANGAN :

1. Field penerima hasil HARUS berbentuk suatu nama data tidak boleh literal
2. Bentuk TO atau GIVING HARUS ada, tetapi tidak boleh digunakan secara bersama- sama.

3. Semua nama data yg digunakan didalam operasi aritmatika HARUS berbentuk data numeric dengan PICTURE YG BELUM DIEDIT, kecuali operand dari field penerima boleh berbentuk numerik dgn PICTURE EDITING
4. ROUNDED Option digunakan bila hasil perhitungan ingin dibulatkan. Dibulatkan ke ATAS bila pecahan terakhir bernilai 5 atau lebih.
5. Bila ON SIZE ERROR Option digunakan maka :
Bila hasil perhitungan utk digit-digit bilangan utuh (high-order digits) tidak bisa masuk seluruhnya ke field penerima, nilai ini tdk disimpan di storage dan program akan memproses imperative statement yg mengikutinya.

Bila ON SIZE ERROR Option tdk digunakan maka :hasilnya walaupun tersimpan, tetapi terpotong (sudah tentu hasilnya menjadi salah).

LATIHAN :

1. BENARKAH BENTUK -BENTUK STATEMENT ADD DIBAWAH INI ?

A.	8	12 ADD A TO B.	B.	8	12 ADD A, B TO C.
C.	8	12 ADD A , B GIVING C.	D.	8	12 ADD A TO 10.
E.	8	12 ADD A, B.	F.	8	12 ADD A TO B GIVING C
G.	Pd WORKING-STORAGE SECTION 8 12 77 A PIC 99 VALUE 55. 01 NILAI. 02 B PIC \$ZZ9. 02 C PIC \$ZZ9.		H.	WORKING-SORAGE SECTION 8 12 SOAL. ADD 10 TO A. ADD 12 TO A. ADD A , 1 TO B.	
	Pd PROCEDURE DIVISION. 8 12 ADD A , B GIVING C				

2. ISILAH SEL - SEL KOSONG DIBAWAH INI !

a. Bila diberi perintah ADD A , 15 TO B

	A	B
Sebelum	0	0
Sesudah		

c. Bila diberi perintah ADD A , 10 GIVING B

	A	B
Sebelum	0	15
Sesudah		

SUBTRACT Statement

Digunakan utk operasi pengurangan

BENTUK STATEMENT SUBTRACT

<u>SUBTRACT</u>	nama-data-1 literal-1	nama-data-2 literal-2
	nama-data-n <u>FROM</u> literal-n	<u>GIVING</u> nama-data-m
	<u>[ROUNDED]</u> [;ON <u>SIZE ERROR</u> imperative statement]	

KETERANGAN :

1. Bila akan mengurangkan suatu nilai dgn nilai - nilai lain, dan hasil pengurangan disimpan pada nilai yg dikurangkan maka digunakan FROM
2. Bila akan mengurangkan suatu nilai dgn nilai lain, dan hasil pengurangan disimpan pada nama data lain maka digunakan FROM dan GIVING
Secara bersama - sama

LATIHAN :

1. BENARKAH BENTUK -BENTUK STATEMENT SUBTRACT DIBAWAH INI

A	8	12	B	8	12
		SUBTRACT A FROM B.			SUBTRACT A , 2 FROM C
C	8	12	D	8	12
		SUBTRACT A FROM C.			SUBTRACT A , 10 FROM 100 GIVING C.
E	8	12	F	8	12
		SUBTRACT A GIVING B			SUBTRACT A,5 FROM D GIVING C

2. ISILAH SEL - SEL KOSONG DIBAWAH INI !

- a. Bila diberi perintah SUBTRACT A FROM B

	A	B
Sebelum	5	80
Sesudah		

MULTIPLY Statement

Digunakan utk operasi perkalian

BENTUK STATEMENT MULTIPLY 1

```
nama-data-1  
MULTIPLY literal-1 BY nama-data-2  
[; ON SIZE ERROR imperative statement ]
```

BENTUK STATEMENT MULTIPLY 2

```
nama-data-1 nama-data-2  
MULTIPLY literal-1 BY literal-2  
GIVING nama-data-3 [ROUNDED]  
[; ON SIZE ERROR imperative statement ]
```

KETERANGAN :

1. Bila hasil perkalian akan disimpan pada salah satu field dari numerik data pengali, maka digunakan bentuk BY
2. Bila hasil perkalian akan disimpan ditempat lain, maka digunakan bentuk BY dan bentuk optional GIVING bersama-sama.
3. Konstanta / literal pengali letaknya sebelum BY bila tidak menggunakan GIVING
4. Setelah GIVING tidak boleh literal.

LATIHAN :

1. BENARKAH BENTUK-BENTUK STATEMENT MULTIPLY DIBAWAH INI ?

A.	8	12	B.	8	12
		MULTIPLY A FROM B.			MULTIPLY A FROM B
C.	8	12	D.	8	12
		MULTIPLY A BY B.			MULTIPLY 5 BY A.
E.	8	12	F.	8	12
		MULTIPLY A BY 5.			MULTIPLY A FROM 3 GIVING C.
G.	8	12	H.	8	12
		MULTIPLY A GIVING B.			MULTIPLY A BY B GIVING 15.

2. ISILAH SEL - SEL KOSONG DIBAWAH INI !

a. Bila diberi perintah MULTIPLY A BY 15 GIVING B

	A	B
Sebelum	5	10
Sesudah		

b. Bila diberi perintah MULTIPLY 5 BY A.

	A
Sebelum	35
Sesudah	

DIVIDE Statement

Digunakan utk operasi pembagian

BENTUK 1 :

nama-data-1
<u>DIVIDE</u> literal-1 <u>INTO</u> nama-data-2 [<u>ROUNDED</u>]
[; ON <u>SIZE ERROR</u> imperative statement]

BENTUK 2 :

nama-data-1 nama-data-2
DIVIDE literal-1 **INTO** literal-2 **GIVING** nama-data-3
ROUNDED [; ON **SIZE ERROR** imperative statement]

BENTUK 3 :

nama-data-1 nama-data-2
DIVIDE literal-1 **BY** literal-2 **GIVING** nama-data-3
ROUNDED [; ON **SIZE ERROR** imperative statement]

BENTUK 4 :

```
DIVIDE nama-data-1 nama-data-2  
literal-1 INTO literal-2 GIVING nama-data-3  
[ROUNDED]  
REMAINDER nama-data-4  
[; ON SIZE ERROR imperative statement ]
```

BENTUK 5 :

```
nama-data-1 nama-data-2  
DIVIDE literal-1 BY literal-2 GIVING nama-data-3 [ROUNDED]  
REMAINDER nama-data-4  
[; ON SIZE ERROR imperative statement ]
```

KETERANGAN :

1. Untuk MS COBOL bentuk ke 4 dan ke 5 tidak tersedia. tetapi pada COBOL Standard ANSI 74 tersedia.
2. Bentuk BY harus diikuti dengan GIVING.
3. Bentuk INTO TIDAK HARUS / OPTIONAL -----> diikuti dgn GIVING

LATIHAN :

1. BENARKAH BENTUK -BENTUK STATEMENT DIVIDE DIBAWAH INI ?

A.	8	12	DIVIDE A FROM B.	B.	8	12	DEVIDE A INTO B
C.	8	12	DIVIDE A BY B.	D.	8	12	DIVIDE B INTO A
E.	8	12	MULTIPLY A BY 5.	F.	8	12	DIVIDE A BY B GIVING C
G.	8	12	DIVIDE A GIVING B.	H.	8	12	DIVIDE A BY 5 GIVING C

2. ISILAH SEL - SEL KOSONG DIBAWAH INI !

a. Bila diberi perintah DIVIDE A BY 15 GIVING B

	A	B
Sebelum	25	4
Sesudah		

COMPUTE verb

Digunakan utk operasi perhitungan yang lebih rumit.

Misal :

RUMUS : $Y = A + B - (C/D) \times E$

Apabila menggunakan statemen aritmatika maka :

8 12

hitung.

ADD A TO B.

DIVIDE D INTO C.

MULTIPLY C BY E.

SUBTRACT E FROM B GIVING Y

Sedangkan bila kita menggunakan statemen COMPUTE

8 12
hitung.

COMPUTE Y = A + B - (C / D) * E.

BENTUK :

COMPUTE nama-data-1 [**ROUNDED**] = ungkapan aritmatika
[; ON **SIZE ERROR** imperative statement]

OPERATOR ARITMATIKA

Operator aritmatika :

Simbol yg digunakan untuk membentuk suatu ungkapan aritmatika

Operator Aritmatika	Maksud	Jenjang
**	Pemangkatan	1
*	Perkalian	2
/	Pembagian	2
+	Penjumlahan	3
-	Pengurangan	3

UNGKAPAN ARITMATIKA

Ungkapan aritmatika :

Kombinasi dari literal numerik , nama data, operator aritmatika dan kurung buka , kurung tutup.

Contoh :

Maksud	Penulisan
$A + B$	$A + B$
$\frac{A+B}{C+D}$	$(A + B) / (C + D)$
$\frac{A^2}{C+D}$	$A ** 2 / (B + C)$

KETERANGAN :

1. Penulisan Statement COMPUTE untuk setiap elemen harus dipisahkan dengan blank dan digunakan OPERATOR ARITMATIKA (Bab 5.6)
2. Penulisan Aoperative Arithmetic harus diperhatikan jenjangnya. (Bab 5.6)