

Pemrograman Dasar



**SEKILAS JENIS-JENIS OPERATOR
OPERATOR PENUGASAN
OPERATOR ARITMATIKA
OPERATOR MAJEMUK**

Operator dan Operand



- *Operator* merupakan simbol yang biasa dilibatkan dalam program untuk melakukan suatu operasi atau manipulasi. Misalnya untuk
 - Menjumlahkan dua buah nilai
 - Memberikan nilai ke suatu variabel
 - Membandingkan dua buah nilai
- Misalkan pada pernyataan $c = a + b$;
- Maka c , a , dan b merupakan *operand*, sedangkan $=$ dan $+$ merupakan *operator*.

Unary, binary dan ternary



- Berdasarkan jumlah *operand* yang terlibat, operator dikelompokkan menjadi:

Sifat	Keterangan	Contoh
unary	melibatkan satu operan	a++ -a
binary	melibatkan dua operan	2*5 10-6
ternary	melibatkan tiga operan	a ? b : c

- *to be continued*

Operator penugasan



- *Operator* penugasan / *assignment*, yaitu ‘ = ‘
- *Operator* penugasan sederhana, menugaskan / menyalin nilai dari *operand* di sebelah kanan ke *operand* di sebelah kiri.
- Contoh:
 - $a = b;$
 - $\text{bayar} = \text{jumlah} * \text{harga};$
 - $\text{hasil} = a + b;$

Operator penugasan



- *Operator* penugasan dapat juga digunakan seperti berikut:
 - `a = b = c = d = e = 1;`
 - `a = 2 + (b = 5);` //nilai a akan menjadi 7.
- Pernyataan `a=b=c=d=e=1;` ekuivalen dengan:
 - `e = 1;`
 - `d = e;`
 - `c = d;`
 - `b = c;`
 - `a = b;`
- *Operator* “ = “ dikerjakan dari kanan ke kiri.

Operator aritmatika



- *Operator* aritmatika (*arithmetic*) berfungsi untuk melakukan operasi aritmatika. *Operator* aritmatika antara lain:

Operator aritmatika

Misalkan nilai $a = 10$, $b = 20$

Operator	Keterangan	Contoh	Hasil
+	menjumlahkan dua operan	$a + b$	30
-	mengurangkan operan kedua dari operan pertama	$a - b$	-10
*	mengalikan dua operan	$a * b$	200
/	membagi operan pertama dengan operan kedua	b / a	2
%	sisa hasil bagi operan pertama dibagi operan kedua	$10 \% 3$	1
++	<i>increment</i> menambah satu ke operan	$a++$	11
--	<i>decrement</i> mengurangkan satu dari operan	$b--$	19
-	<i>minus</i> membalik tanda operan	$-a$	-10

Latihan modulus



- Berapakah hasil dari operasi berikut:
 - $20 \% 3$
 - $9 \% 5$
 - $14 \% 6$
 - $5 \% 6$
 - $12 \% 3$
 - $10 \% 20$

operator bitwise



- *Operator bitwise* adalah operator yang mengoperasikan nilai dalam tingkat bit/biner.

operator	asm equivalent	description
&	AND	Bitwise AND
	OR	Bitwise Inclusive OR
^	XOR	Bitwise Exclusive OR
~	NOT	Unary complement (bit inversion)
<<	SHL	Shift Left
>>	SHR	Shift Right

- Disambung di belakang.

Operator majemuk



- *Operator majemuk (compound assignment)*

Operator	Contoh	Keterangan
+=	x += 2;	Kependekan dari x = x + 2;
-=	x -= 2;	Kependekan dari x = x - 2;
*=	x *= 2;	Kependekan dari x = x * 2;
/=	x /= 2;	Kependekan dari x = x / 2;
%=	x %= 2;	Kependekan dari x = x % 2;
<<=	x <<= 2;	Kependekan dari x = x << 2;
>>=	x >>= 2;	Kependekan dari x = x >> 2;
&=	x &= 2;	Kependekan dari x = x & 2;
=	x = 2;	Kependekan dari x = x 2;
^=	x ^= 2;	Kependekan dari x = x ^ 2;

Latihan



- Buat program yang menerima input dua bilangan (a dan b) dan kemudian menampilkan hasil penjumlahan, pengurangan ($a-b$), perkalian, pembagian (a/b) dan sisa bagi ($a\%b$).

Latihan



- Buat program untuk melakukan operasi-operasi aritmatika berturut-turut: penjumlahan, pengurangan, perkalian, pembagian, sisa bagi. Masing-masing terhadap hasil operasi sebelumnya. Hanya boleh menggunakan 2 variabel saja.

```
c:\Documents and Settings\fandi\My Documents\Visual Studio 2008\Projects\latihan\Debug\lat... - □ ×
Masukkan suatu bilangan:5
Jumlahkan bilangan tersebut dengan:7
Hasilnya:12
Hasil sebelumnya dikurangkan dengan:3
Hasilnya:9
Hasil sebelumnya dikalikan dengan:5
Hasilnya:45
Hasil sebelumnya dibagi dengan:3
Hasilnya:15
Hasil sebelumnya sisa baginya terhadap:6
Hasilnya:3
ya
```

operator perbandingan



- *Operator perbandingan (hasil: true atau false)*
- *Comparison operators/Relational operators*

Operator	Keterangan	Contoh	Hasil
==	Sama dengan	$a == b$	FALSE
!=	Tidak sama dengan	$a != b$	TRUE
>	Lebih besar dari	$a > b$	FALSE
<	Lebih kecil dari	$a < b$	TRUE
>=	Lebih besar atau sama dengan	$a >= b$	FALSE
<=	Lebih kecil atau sama dengan	$a <= b$	TRUE

- To be continued

operator logika



- *Operator logika (Logical operator)*
- Berguna untuk menggabungkan dua kondisi

Operator	Indonesia	Logika
&&	DAN	AND
	ATAU	OR
!	BUKAN	NOT

- Hasilnya juga *true* atau *false*

operator logika



kondisi1	kondisi2	&&	
TRUE	TRUE	TRUE	TRUE
TRUE	FALSE	FALSE	TRUE
FALSE	TRUE	FALSE	TRUE
FALSE	FALSE	FALSE	FALSE

- To be continued

Prioritas *operator*



- Jika dalam satu baris yang sama terdapat banyak operator, maka prioritasnya:

Operator	Urutan pengerjaan
++ --	←
! ~ -	←
* / %	→
+ -	→
<< >>	→
< > <= >=	→
== !=	→
& ^	→
&&	→
? :	→
= *= /= %=	→
+= -= <<= >>=	→
&= ^= =	→

Latihan



- Berapakah hasil dari operasi aritmatika berikut:
 - $2 + 3 - 5$
 - $7 * 3 + 5$
 - $4 + 2 * 7 + 4$
 - $5 + 3 * 3 + 2 * 1 + 6$
 - $6 / 2 + 3 * 6$

Menggunakan tanda kurung



- Jika perlu, kita dapat menggunakan tanda kurung untuk mengubah prioritas operasi.
- Misalkan pernyataan $2 + 5 * 7$ akan menghasilkan nilai 37. Sedangkan kita menginginkan operasi penjumlahan dilakukan paling pertama. Maka kita dapat menggunakan tanda kurung: $(2+5) * 7$

Latihan



- Buat program yang menghitung persamaan berikut:
 - $L = \pi r^2$ r diinput
 - $x = \frac{a+b}{c+d}$ a, b, c dan d diinput
 - $I = I_1 + \frac{V_2}{R_2}$ I dalam mA (harus dijadikan A)
 - V dalam Volt (oke)
 - R dalam K Ω (harus dijadikan Ω)
 - I dalam A.

Perhitungan matematika



- Terkadang, perhitungan sederhana saja tidak memenuhi kebutuhan kita. Terkadang kita perlu untuk menggunakan fungsi seperti misalnya sinus, cosinus, tangen dan akar.
- Untuk itu, kita perlu memanggil library math.h dengan menyertakan baris program `#include <math.h>`

Perhitungan matematika

Fungsi	Keterangan
sin	menghitung nilai sinus
cos	menghitung nilai cosinus
tan	menghitung nilai tangen
asin	mengembalikan nilai yang sinusnya adalah nilai argumen
acos	mengembalikan nilai yang cosinusnya adalah nilai argumen
atan	mengembalikan nilai yang tangennya adalah nilai argumen
log	menghitung nilai logaritma
log10	menghitung nilai logaritma basis 10
exp	menghitung nilai pangkat bilangan natural
sqrt	menghitung nilai akar
Fungsi	Keterangan
abs	nilai mutlak / absolut / positif
ceil	pembulatan ke atas
floor	pembulatan ke bawah
min	mencari nilai terkecil
max	mencari nilai terbesar

Perhitungan matematika



- Contoh:

```
#include <iostream>
#include <conio.h>
#include <math.h>
using namespace std;
void main() {
    float bilangan; //fungsi sqrt hanya terima pecahan
    cout<<"Program menghitung akar bilangan"<<endl;
    cout<<"Masukkan bilangan yang ingin dihitung:";
    cin>>bilangan;
    cout<<"Akar bilangan tersebut:"<<sqrt(bilangan);
    _getch();
}
```

Perhitungan matematika



- Contoh:

```
#include <iostream>
#include <conio.h>
#include <math.h>
using namespace std;
void main() {
    const long double PI = 3.14159265358979323846;
    float sudut; //fungsi sqrt hanya terima pecahan
    cout<<"Program menghitung sin,cos,tan sudut"<<endl;
    cout<<"Masukkan sudut yang ingin dihitung:";
    cin>>sudut;
    cout<<"sinus sudut tersebut:"<<sin(sudut)<<endl;
    cout<<"cosinus sudut tersebut:"<<cos(sudut)<<endl;
    cout<<"tangen sudut tersebut:"<<tan(sudut)<<endl;
    _getch();
}
```

Thank You for your attention



小事不做，大事難成。

Jika enggan mengerjakan hal kecil, maka kita pun akan sulit menyelesaikan tugas.



- Thanks to:

- Pemrograman C++ by Abdul Kadir
- Yoannita S.Kom & Hermawan S.Kom → Sumber slide
- <http://www.tzuchi.or.id/perenungan.php?page=1> → Renungan

operator bitwise



- *Operator bitwise* adalah operator yang mengoperasikan nilai dalam tingkat bit/biner.

operator	asm equivalent	description
&	AND	Bitwise AND
	OR	Bitwise Inclusive OR
^	XOR	Bitwise Exclusive OR
~	NOT	Unary complement (bit inversion)
<<	SHL	Shift Left
>>	SHR	Shift Right

operator bitwise



- Dalam menggunakan operator biner, perlu diketahui penggunaan basis bilangan:
- Basis 10 (desimal): angka terbesar 9 (yang kita pakai sehari-hari)
- Basis 2(biner): angka terbesar 1
- Basis 8(oktal): angka terbesar 7
- Basis 16(heksadesimal): angka terbesar 16
- Contoh bilangan desimal, biner, oktal dan heksa:
 - 11001010 (biner)
 - 0312 (oktal)
 - xCA (heksa)
 - 202 (desimal)

operator bitwise



- Mengkonversi basis:
 - Angka 202 desimal ke:
 - Biner: 202 dibagi 2 sampai jadi nol.
 - Ambil dari bawah ke atas:
 - 11001010
- | | |
|------------------|---|
| 202 bagi 2 sisa: | 0 |
| 101 bagi 2 sisa: | 1 |
| 50 bagi 2 sisa: | 0 |
| 25 bagi 2 sisa: | 1 |
| 12 bagi 2 sisa: | 0 |
| 6 bagi 2 sisa: | 0 |
| 3 bagi 2 sisa: | 1 |
| 1 bagi 2 sisa: | 1 |
| <hr/> | |
| 0 | |

operator bitwise



- Mengkonversi basis:
- Angka 202 desimal ke:
 - Oktal: 202 dibagi 8 sampai jadi nol.
 - Ambil dari bawah ke atas:
 - 312 ditulis 0312

$$\begin{array}{r} 202 \text{ bagi } 8 \text{ sisa: } 2 \\ 25 \text{ bagi } 8 \text{ sisa: } 1 \\ 3 \text{ bagi } 8 \text{ sisa: } 3 \\ \hline 0 \end{array}$$

operator bitwise



- Mengkonversi basis:
- Angka 202 desimal ke:
 - Heksadesimal: 202 dibagi 16 sampai jadi nol.
 - Ambil dari bawah ke atas:
 - CA ditulis xCA

$$\begin{array}{r} 202 \text{ bagi } 16 \text{ sisa: } 10 = A \\ 12 \text{ bagi } 16 \text{ sisa: } 12 = C \\ \hline 0 \end{array}$$

operator bitwise



- Mengkonversi basis:
- Angka 11001010 biner ke:
 - Desimal

1	1	0	0	1	0	1	0
2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
128	64	32	16	8	4	2	1
128	64	0	0	8	0	2	0
Total:							202

operator bitwise



- Mengkonversi basis:
- Angka 11001010 biner ke:
 - Oktal

0	1	1	0	0	1	0	1	0
	3		1			2		

- Heksadesimal

1	1	0	0	1	0	1	0
	C			A			

operator bitwise



- **Contoh:**

A	11101011
B	01100100

A & B	01100000
A B	11101111
A ^ B	10001111
~A	00010100
~B	10011011
A<<2	1110101100
B<<2	110010000
A>>2	00111010
B>>2	00011001