

1 D4 TEKKOM B

EXPRESSION (OPERATOR & OPERAND) & STATEMENT



Nama	: M. Fierlyan I.
NRP	: 3224600051
Kelas	: 1 D4 Teknik Komputer B
Mata Kuliah	: Praktikum Pemrograman Dasar
Dosen	: Ir Sigit Wasista M.Kom
Hari/Tgl Praktikum	: Kamis, 5 September 2024

Percobaan 2

2.1 JUDUL: EXPRESSION (OPERATOR & OPERAND) & STATEMENT

2.2 TUJUAN: Mempelajari dan mengamati penggunaan operator aritmatika seperti +, -, *, / dan % (modulus) dan operator unary seperti - (minus), ++ (increment) dan - (decrement) yang dipadukan dengan tipe variable dasar yaitu int (bilangan bulat) dan float (presisi tunggal), double (presisi ganda) dan char.

2.3 TEORI: Suatu Expression adalah terdiri atas dua bagian yaitu operator dan operand, yang dikombinasikan menjadi satu. Operator adalah symbol yang digunakan untuk melakukan proses operasi satu atau beberapa operand. Operand adalah bagian yang paling sederhana dari Expression. Operand dapat berupa konstanta seperti 339, 55, 132 dan seterusnya atau berupa variable seperti **x**, **y**, jumlah, selisih dan sebagainya. Statement (statement) adalah unsur dasar pembentuk suatu program. Suatu program terdiri dari beberapa statement, dimana komputer akan melakukan tugas tertentu sesuai dengan urutan statement. Ada 3 jenis statement, yaitu: expression statement, compound statement, dan control statement. Suatu expression statement adalah suatu expression yang diikuti dengan tanda titik koma [:] (semi colon). Suatu compound statement (dikenal dengan block statement) adalah dua atau lebih statement yang dikelompokkan menjadi satu dengan cara memberi batas tanda kurung awal dan tanda kurung akhir, sehingga tidak perlu diakhiri dengan tanda titik koma pada akhir dari compound. Control statement adalah statement yang mengendalikan langkah-langkah program, contohnya for loop, while loop, dan if-else (yang akan dilakukan pada percobaan berikutnya). Symbolic Constant adalah suatu nama dimana digunakan untuk menggantikan suatu nilai tertentu, sehingga akan lebih mudah dalam pembacaan suatu program, contohnya #define PI 3.14, untuk membedakan dengan variable maka nama ditulis dengan huruf besar seperti PI.

2.4 PROGRAM PERCOBAAN

2.4.1 Mengamati tipe variabel yang digunakan untuk menyimpan suatu nilai sesuai dengan tipe dari masing-masing nilai yang diberikan pada suatu variabel, kemudian mencetak isi variabel tersebut dan melihat hasilnya. Misalnya bilangan bulat, maka dapat ditampung menggunakan tipe variabel integer yang mempunyai jangkauan dari -32768 sampai 3276, bilangan pecahan dengan presisi tunggal dapat disimpan pada tipe variabel dengan tipe float yang mempunyai jangkauan dari 3.4×10^{-32} sampai 3.4×10^{38} (7 angka presisi), bilangan pecahan dengan presisi ganda dapat disimpan pada tipe variabel dengan tipe double yang mempunyai jangkauan 3.4×10^{-4932} sampai dengan 3.4×10^{4932} (19 angka presisi) dan tipe variabel char dengan jangkauan nilai dari -128 sampai 127 (satu karakter).

```
/* Nama File : EXPRESS1.C */

void main()
{
    int bulat = 32767;
    float pecahan = 339.1234567;
    double ganda = 3.4567890e+11;
    char karakter = 'W';

    printf("Variabel bilangan bulat      = %d\n\n", bulat);
    printf("Variabel bilangan pecahan    = %d\n\n", pecahan);
    printf("Variabel bilangan pecahan2   = %d\n\n", ganda);
    printf("Variabel bilangan karakter = %d\n\n", karakter);

    getch();
}
```

```
#include <stdio.h>
#include <conio.h>

void main() {
    int bulat = 32767;
    float pecahan = 339.1234567;
    double ganda = 3.4567890e+11;
    char karakter = 'W';

    printf("Variabel bilangan bulat = %d\n\n", bulat);
    printf("Variabel bilangan pecahan = %f\n\n", pecahan);
    printf("Variabel bilangan pecahan2 = %e\n\n", ganda);
    printf("Variabel karakter = %c\n\n", karakter);

    getch();
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

Variabel bilangan bulat = 32767

Variabel bilangan pecahan = 339.123444

Variabel bilangan pecahan2 = 3.456789e+11

Variabel bilangan pecahan2 = 3.456789e+11

Variabel karakter = W

█

2.4.2 Mengamati hasil dari suatu persamaan aritmatika, dimana nilai-nilai operannya dimasukkan melalui keyboard serta hirarki dari operator.

```
/* Nama File : EXPRESS2.C */

void main()
{
    int a, b, c, d, hasil;

    printf("\nMasukkan nilai a b = ");
    scanf("%d %d", &a, &b);
    printf("\nMasukkan nilai c d = ");
    scanf("%d %d", &c, &d);
    printf("a = %d, b = %d, c = %d, d = %d", a, b, c, d);

    hasil = a - b;
    printf("\nHasil dari: a - b adalah %d\n", hasil);

    hasil = c + d;
    printf("\nHasil dari: c + d adalah %d\n", hasil);

    hasil = b * c;
    printf("\nHasil dari: b * c adalah %d\n", hasil);

    hasil = a / c;
    printf("\nHasil dari: a / c adalah %d\n", hasil);

    hasil = a + b * c;
    printf("\nHasil dari: a + b * c adalah %d\n", hasil);

    hasil = a * b + c * d;
    printf("\nHasil dari: a * b + c * d adalah %d\n", hasil);

    getch();
}
```

```
#include <stdio.h>
#include <conio.h>

void main() {
    int a, b, c, d, hasil;

    printf("\nMasukan nilai a b = ");
    scanf("%d %d", &a, &b);
    printf("\nMasukan nilai c d = ");
    scanf("%d %d", &c, &d);
    printf("a = %d, b = %d, c = %d, d = %d\n", a, b, c, d);

    hasil = a - b;
    printf("\nHasil dari a - b = %d\n", hasil);

    hasil = c + d;
    printf("\nHasil dari c + d = %d\n", hasil);

    hasil = b * c;
    printf("\nHasil dari b * c = %d\n", hasil);

    hasil = a / c;
    printf("\nHasil dari a / c = %d\n", hasil);

    hasil = a + b * c;
```

```

printf("\nHasil dari a + b * c = %d\n", hasil);

hasil = a * b + c * d;
printf("\nHasil dari a * b + c * d = %d\n", hasil);

getch();

}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

Masukan nilai c d = 3 4
a = 1, b = 2, c = 3, d = 4

Hasil dari a - b = -1

Hasil dari c + d = 7

Hasil dari b * c = 6

Hasil dari a / c = 0

Hasil dari a + b * c = 7

Hasil dari a * b + c * d = 14
PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2>

```

- 2.4.3 Mengamati perbedaan penggunaan tipe variabel integer dan tipe variabel float pada proses pembagian, bila pembagian dilakukan menggunakan tipe integer maka pecahan akan dibuang. Perhatikan hasil dari pembagian isi variabel a dibagi dengan 5, isi variabel a dibagi dengan isi variabel b dan isi variabel c dibagi dengan d. Perhatikan juga penggunaan operator unari minus (-).

```

/* Nama File : EXPRESS3.C */

void main()
{
    int a = 12;
    int b = 8;
    int hasil;
    float c = 11.0;
    float d = 7.0;

    printf("6 + a / 5 * b = %d\n\n", 6 + a / 5 * b);
    printf("a / b * b = %d\n\n", a / b * b);
    printf("c / d * d = %f\n\n", c / d * d);
    printf("-a = %d\n", -a);

    getch();
}

```

```
#include <stdio.h>

void main() {
    int a = 12;
    int b = 8;
    int hasil;
    float c = 11.0;
    float d = 7.0;

    printf("6 + a / 5 * b = %d\n\n", 6 + a / 5 * b);
    printf("a / b * b = %d\n\n", a / b * b);
    printf("c / d * d = %f\n\n", c / d * d);
    printf("-a = %d\n\n", -a);
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

6 + a / 5 * b = 22

a / b * b = 8

c / d * d = 11.000000

-a = -12

PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2> 
```

- 2.4.4 Untuk menghitung sisa dari hasil bagi maka telah disediakan operator modulus yaitu (%), dimana hasil dari operasi dengan operator ini akan menghasilkan sisa pembagian. Perlu diperhatikan bahwa dikarenakan operator ini adalah sama dengan operator format, sehingga bila operator modulus ingin ditampilkan, maka penulisan operator ini harus ditulis dua kali (%%)

```

/* Nama File : EXPRESS4.C */

void main()
{
    int a = 14, b = 2, c = 3, d = 4;

    printf("a %% b = %d\n\n", a % b);
    printf("a %% c = %d\n\n", a % c);
    printf("a %% d = %d\n\n", a % d);
    printf("a / d * d + a %% d = %d\n", a / d * d + a % d);

    getch();
}

```

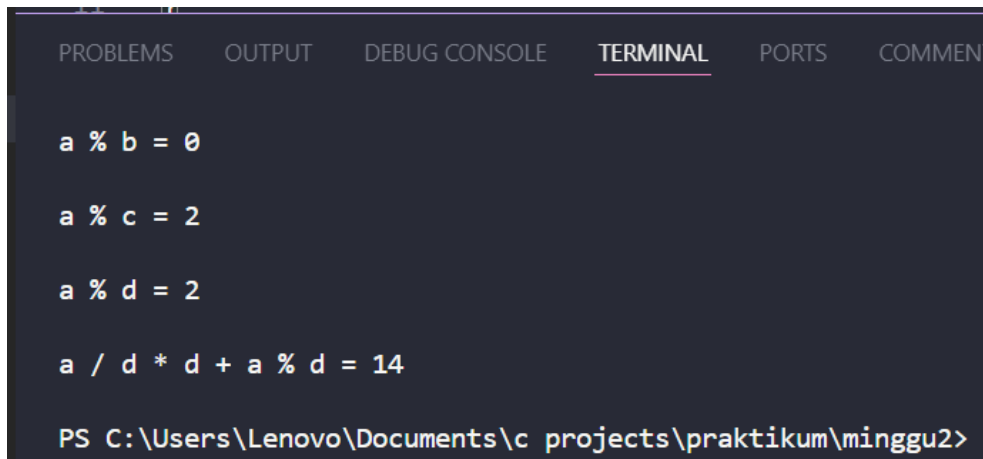
```

#include <stdio.h>

void main() {
    int a = 14, b = 2, c = 3, d = 4;

    printf("a %% b = %d\n\n", a % b);
    printf("a %% c = %d\n\n", a % c);
    printf("a %% d = %d\n\n", a % d);
    printf("a / d * d + a %% d = %d\n\n", a / d * d + a % d);
}

```



```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENT

a % b = 0

a % c = 2

a % d = 2

a / d * d + a % d = 14

PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2>

```

- 2.4.5 Suatu variabel dapat menampung isi dari suatu tipe variabel yang berbeda asalkan tipe variabel tersebut lebih besar atau sama dengan variabel sumber. Contoh tipe variabel float dapat menampung isi dari tipe variabel integer dan tidak sebaliknya.

```

/* Nama File : EXPRESS5.C */

void main()
{
    float p1 = 123.45678, f2;
    int b1, b2 = -150;

    b1 = p1;          /* konversi pecahan ke bulat */
    printf("%f disimpan ke int menghasilkan %d\n\n", p1, b1);

    p1 = b2;          /* konversi bulat ke pecahan */
    printf("%d disimpan ke float menghasilkan %f\n\n", b2, p1);

    p1 = b2 / 100;     /* bulat dibagi bulat */
    printf("%d dibagi 100 menghasilkan %f\n\n", b2, p1);

    p2 = b2 / 100.0;   /* bulat dibagi pecahan */
    printf("%d dibagi 100 menghasilkan %f\n\n", b2, p2);

    getch();
}

```

```

#include <stdio.h>

void main() {
    float p1, p2 = 123.45678, f2;
    int b1, b2 = -150;

    b1 = p1;
    printf("%f disimpan ke int menghasilkan %d\n\n", p1, b1);

    p1 = b2;
    printf("%d disimpan ke float menghasilkan %f\n\n", b2, p1);

    p1 = b2 / 100;
    printf("%d dibagi 100 disimpan menghasilkan %f\n\n", b2, p1);

    p2 = b2 / 100.0;
    printf("%d dibagi 100 disimpan menghasilkan %f\n\n", b2, p2);
}

```



```
PROBLEMS 1 OUTPUT DEBUG CONSOLE TERMINAL PORTS CONSOLE

0.000000 disimpan ke int menghasilkan 0

-150 disimpan ke float menghasilkan -150.000000

-150 dibagi 100 disimpan menghasilkan -1.000000

-150 dibagi 100 disimpan menghasilkan -1.500000

PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2>
```

2.4.6 Operator increment (++) dan operator decrement (--) hanya dapat digunakan pada suatu variabel, dan tidak dapat digunakan pada suatu konstanta. Penggunaan dari operator ini akan berakibat perubahan pada variabel, yaitu ditambah satu (++) atau dikurangi satu (--), contoh $x++$ adalah sama dengan $x=x+1$ dan $x--$ adalah sama dengan $x=x-1$. Sedangkan posisi dari tanda (prefix atau postfix) juga berpengaruh pada hasil suatu expression.

```
/* Nama File : EXPRESS6.C

void main()
{
    int x, y, z;

    x=80; y=x++; z=++x;
    printf("Nilai dari x, y, z adalah %d, %d, %d\n\n", x, y, z);

    y=x--; z=--x;
    printf("Nilai dari x, y, z adalah %d, %d, %d", x, y, z);

    getch();
}
```

```
#include <stdio.h>

void main() {
    int x, y, z;

    x=80; y=x++; z=++x;
    printf("Nilai dari x, y, z adalah %d, %d, %d\n\n", x, y, z);

    y=x--; z=--x;
    printf("Nilai dari x, y, z adalah %d, %d, %d", x, y, z);
}
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL  PORTS  COMMENT

Nilai dari x, y, z adalah 82, 80, 82

Nilai dari x, y, z adalah 80, 82, 80
PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2>
```

2.4.7 Format tampilan dari suatu variabel dengan tipe pecahan dapat digunakan tiga macam format tampilan yaitu menggunakan karakter khusus `%e`, `%f`, dan `%g`. Disini diberikan program untuk mencoba hasil yang ditampilkan dari ketiga karakter format tersebut.

```
/* Nama File: EXPRESS7.C */

#include <stdio.h>

void main()
{
    float x;
    printf("Masukkan nilai pecahan yg akan ditampilkan : ");
    scanf("%f", &x);

    printf("format e  => %e\n", x);
    printf("format f  => %f\n", x);
    printf("format g  => %g\n", x);

    getch();
}
```

```
#include <stdio.h>

void main() {
    float x;
    printf("Masukkan nilai pecahan yang akan ditampilkan : ");
    scanf("%f", &x);

    printf("format e => %e\n", x);
    printf("format f => %f\n", x);
    printf("format g => %g\n", x);
}
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

Masukkan nilai pecahan yang akan ditampilkan : 2.5
format e => 2.500000e+00
format f => 2.500000
format g => 2.5
PS C:\Users\Lenovo\Documents\c projects\praktikum\minggu2>
```

2.5 TUGAS-TUGAS

2.5.1 Buatlah program untuk mengkonversikan suatu nilai derajat Fahrenheit (F) ke derajat Celcius (C) menggunakan formula dibawah ini:

$$C = \frac{5}{9}(F - 32)$$

Contoh tampilan:

```
Masukkan derajat Fahrenheit = 212 ↵
212 derajat Fahrenheit adalah = 100 derajat Celcius
```

```
#include <stdio.h>

int main() {
    float fahrenheit, celsius;

    printf("Masukkan derajat Fahrenheit = ");
    scanf("%f", &fahrenheit);

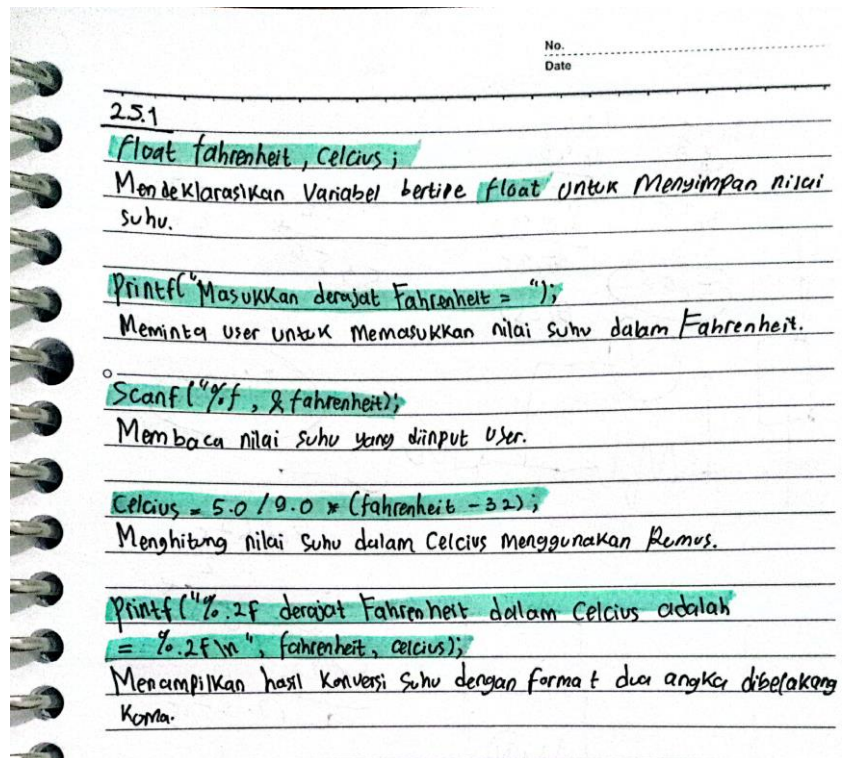
    celsius = 5.0 / 9.0 * (fahrenheit - 32);

    printf("%.2f derajat Fahrenheit dalam Celcius adalah = %.2f\n", fahrenheit, celsius);

    return 0;
}
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

Masukkan derajat Fahrenheit = 212
212.00 derajat Fahrenheit dalam Celcius adalah = 100.00
PS C:\Users\Lenovo\Documents\c projects\github_tugas\pertemuan_2>
```



2.5.2 Buatlah program untuk mengevaluasi formula dibawah ini:

$$y = 3x^2 + 6x + 9$$

$$z = \frac{2y^2 + 5x^2}{9y}$$

Contoh tampilan:

Formula: $y = 3x^2 + 6x + 9$

Masukkan nilai $x = 2$ ↴

Didapatkan nilai $y = 33$ dan nilai $z = 7.4006734$

```
#include <stdio.h>
#include <math.h>

int main() {
    double x, y, z;

    printf("Formula 1 & 2\n");
    printf("Masukan nilai x = ");
    scanf("%lf", &x);

    // cari y pakai formula 1
    y = 3 * pow(x, 2) + 6 * x + 9;

    // hitung z pakai formula 2
```

```

z = (2 * pow(y, 2) + 5 * pow(x, 2)) / (9 * y);

// output
printf("Didapatkan nilai y = %.2f dan nilai z = %.7f\n", y, z);

return 0;
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

Formula 1 & 2
 Masukan nilai x = 2
 Didapatkan nilai y = 33.00 dan nilai z = 7.4006734
 PS C:\Users\Lenovo\Documents\c projects\github_tugas\pertemuan_2> |

No. _____
 Date _____

Formula 1 : $y = 3x^2 + 6x + 9$

- $\text{pow}(x, 2)$ digunakan untuk menghitung x^2
- Hasil perhitungan y disimpan dalam Variabel y

Formula 2 : $z = \frac{2y^2 + 5x^2}{9y}$

- $\text{pow}(y, 2)$ digunakan untuk menghitung y^2
- $\text{pow}(x, 2)$ digunakan untuk menghitung x^2
- Hasil perhitungan z disimpan dalam Variabel z

$\%.2f$ dalam `printf` digunakan untuk menampilkan angka desimal dengan dua angka dibelakang koma.

2.5.3 Buatlah program untuk menghitung keliling dan luas suatu lingkaran, dengan formula :

$$\text{Keliling} = 2\pi r$$

$$\text{Luas} = \pi r^2$$

Contoh tampilan:

Masukkan jari-jari lingkaran = 12 ↵

Keliling lingkaran dengan jari-jari 12 adalah = 75.36

Luas Lingkaran dengan jari-jari 12 adalah = 452.16

```

#include <stdio.h>
#include <math.h>
#define M_PI 3.14159

int main() {
    float jarijari;

    printf("Masukan jari-jari lingkaran = ");
    scanf("%f", &jarijari);

    float keliling = 2 * M_PI * jarijari;
    float luas = M_PI * pow(jarijari, 2);

    // output
    printf("Keliling lingkaran dengan jari-jari %.2f adalah = %.2f\n",
jarijari, keliling);
    printf("Luas lingkaran dengan jari-jari %.2f adalah = %.2f\n", jarijari,
luas);

    return 0;
}

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```

Masukan jari-jari lingkaran = 12
Keliling lingkaran dengan jari-jari 12.00 adalah = 75.40
Luas lingkaran dengan jari-jari 12.00 adalah = 452.39
PS C:\Users\Lenovo\Documents\c projects\github_tugas\pertemuan_2>

```

No. _____
Date _____

$\text{float Keliling} = 2 * M_PI * \text{jarijari};$
Menghitung Keliling lingkaran menggunakan rumus $2\pi r$.

$\text{float luas} = M_PI * \text{pow}(\text{jarijari}, 2);$
Menghitung luas lingkaran menggunakan rumus πr^2

$\%.2f$ dalam `printf` digunakan untuk menampilkan angka desimal dengan dua angka dibelakang koma.

2.5.4 Buatlah program untuk mengkonversi dari jam ke menit

Contoh tampilan:

Program konversi jam ke menit

Masukkan jam dan menit (jj:mm) = 11:07 ↵

Jam 11:07 adalah setara dengan 667 menit

Petunjuk: gunakan fungsi scanf() sebagai berikut:

```
scanf("%d:%d",&jam,&menit);  
#include <stdio.h>  
  
int main() {  
    int jam, menit;  
    printf("program konversi jam ke menit\n");  
    printf("masukan jam dan menit (jj:mm) = ");  
    scanf("%d:%d", &jam, &menit);  
    printf("jam %d:%d adalah setara dengan %d menit\n", jam, menit, jam * 60 +  
menit);  
    return 0;  
}
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS COMMENTS

```
program konversi jam ke menit  
masukan jam dan menit (jj:mm) = 11:07  
jam 11:7 adalah setara dengan 667 menit  
PS C:\Users\Lenovo\Documents\c projects\github_tugas\pertemuan_2> |
```

Fungsi **printf** dan **scanf**:

printf digunakan untuk Menampilkan output ke layar.

scanf digunakan untuk Membaca input dari user dalam format **jj:mm**.

Total Menit dihitung dengan rumus **jam * 60 + menit**, dimana **jam** dikonversi ke Menit dengan Mengalikannya dengan 60, Kemudian ditambahkan dengan **menit**.

2.5.5 Bagaimana tampilan dari program dibawah ini? Mengapa demikian?

```
/* Nama File: EXPRESS8.C */

void main()
{
    int a, b, c, d;
    double e, f, g, h;

    a=75; b=259; c=3315; d=12345;
    e=75.07; f=259.055; g=3315.3310; h=12345.67890;

    printf("a=%7d\nb=%7d\nc=%7d\nd=%7d\n\n", a, b, c, d);
    printf("e=%12.5f\nf=%12.5f\ng=%12.5f\nh=%12.5f\n", e, f, g, h);

    getch();
}
```

```
#include <stdio.h>

void main()
{
    int a, b, c, d;
    double e, f, g, h;

    a=75, b=259, c=3315, d=12345;
    e=75.07, f=259.055, g=3315.3310, h=12345.67890;

    printf("a=%7d\nb=%7d\nc=%7d\nd=%7d\n\n", a, b, c, d);
    printf("e=%12.5f\nf=%12.5f\ng=%12.5f\nh=%12.5f\n", e, f, g, h);

    getch();
}
```

PROBLEMS	OUTPUT	DEBUG CONSOLE	TERMINAL	PORTS	COMMENTS
	a= 75 b= 259 c= 3315 d= 12345 e= 75.07000 f= 259.05500 g= 3315.33100 h= 12345.67890 █				

Fungsi printf :

```
printf("a=%7d\nb=%7d\nc=%7d\n\n", a, b, c, d);
```

Menampilkan nilai Integer dengan lebar 7 Karakter, sehingga nilai akan diratakan ke kanan.

```
printf("e=%12.5f\nf=%12.5f\ng=%12.5f\nh=%12.5%f\n",  
e, f, g, h);
```

- Menampilkan nilai desimal dengan lebar 12 karakter dan 5 angka di belakang koma, sehingga nilai akan diratakan ke kanan dan ditampilkan dengan presisi 5 angka desimal.

Penggunaan void main():

void main() tidak sesuai dengan standar C. Sebaiknya menggunakan int main() dan mengembalikan nilai 0 dengan return 0;

Fungsi getch() digunakan untuk menunggu input karakter dari user sebelum mengakhiri program. Fungsi ini memerlukan #include <conio.h>