

VERSION 2.0

JULY, 2022



PEMROGRAMAN DASAR

MODUL 2 – STRING, BOOLEAN, OPERATORS

DISUSUN OLEH:

- Alif Fatwa Ramadhani
- Azka Faza Dzulqarnain

DIAUDIT OLEH:

- Hardianto Wibowo, S.Kom, M.T

PRESENTED BY: TIM LAB-IT UNIVERSITAS

MUHAMMADIYAH MALANG

PEMROGRAMAN DASAR

TARGET PRAKTIKUM

1. Mampu menguasai konsep pemrograman
 2. Mengimplementasikan operator, string, dan boolean
 3. Memahami definisi dan kegunaan operator, string, dan boolean
 4. Memahami penggunaan operator, string dan boolean
-

PERSIAPAN SOFTWARE/APLIKASI

- Komputer/Laptop
 - Software (Falcon/Dev C++)
 - Bahasa Pemrograman C/C++
-

MATERI PRAKTIKUM

2.1 STRING LITERALS

String sebenarnya adalah larik karakter satu dimensi yang diakhiri dengan karakter nol '\0'. Jadi string yang diakhiri null berisi karakter yang terdiri dari string yang diikuti oleh null.

Deklarasi dan inisialisasi berikut membuat string yang terdiri dari kata "Hello". Untuk menampung karakter nol di akhir larik, ukuran larik karakter yang berisi string adalah satu lebih banyak dari jumlah karakter pada kata "Halo."

```
char greeting[6] = {'H', 'e', 'l', 'l', 'o', '\0'};
```

Jika Anda mengikuti aturan inisialisasi array maka Anda dapat menulis pernyataan di atas sebagai berikut -

```
char greeting[] = "Hello";
```

```

#include <iostream>
#include <string>

using namespace std;

int main()
{
    string greeting = "Hello world!"
    cout << greeting;
    return 0;
}

```

Sr.No.	Fungsi & Tujuan
1	strcpy(s1, s2); Salin string s2 menjadi string s1.
2	strcat(s1, s2); Menggabungkan string s2 ke ujung string s1.
3	strlen(s1); Mengembalikan panjang string s1.
4	strcmp(s1, s2); Mengembalikan 0 jika s1 dan s2 sama; kurang dari 0 jika s1 < s2; lebih besar dari 0 jika s1 > s2.
5	strchr(s1, ch); Mengembalikan pointer ke kemunculan pertama karakter ch dalam string s1.
6	strstr(s1, s2); Mengembalikan pointer ke kemunculan pertama string s2 dalam string s1.

C mendukung berbagai fungsi yang memanipulasi string yang diakhiri null.

➤ **Penggabungan String**

Operator + dapat digunakan diantara string untuk menambahkannya bersama-sama untuk membuat string baru

```
#include <iostream>
#include <string>

using namespace std;

int main ()
{
    string firstName = "Azka";
    string lastName = "Faza";
    string fullName = firstName + " " + lastName;

    cout << fullName;
    return 0;
}
```

➤ **PANJANG STRING**

Untuk mendapatkan panjang string, gunakan fungsi length().

```
#include <iostream>
#include <string>

using namespace std;

int main ()
{
    string txt = "INFORMATIKA";
    cout << "Panjang string txt = " << txt.length();

    //atau

    cout << "Panjang string txt = " << txt.size();
    return 0;
}
```

➤ **AKSES STRING**

Akses String digunakan untuk mengakses karakter string dengan mengacu pada nomor indeks yang berada di dalam tanda kurung siku [].

```
#include <iostream>
#include <string>

using namespace std;

int main ()
{
    string nama = "ALPHA";
    cout << nama[1]
    return 0;
}
```

➤ **USER INPUT STRING**

Operator ekstraksi (>>) dapat digunakan cin pada untuk menampilkan string yang dimasukkan oleh pengguna.

```
#include <iostream>
#include <string>

using namespace std;

int main ()
{
    string namaPertama;

    cout << "Masukkan nama pertama anda: "
    cin >> namaPertama;
    cout >> "nama anda adalah " << namaPertama;

    return 0;
}
```

```

#include <iostream>
#include <string>

using namespace std;

int main ()
{
    string namaLengkap;

    cout << "Masukkan nama pertama anda: "
    getline (cin , namaLengkap);
    cout >> "Nama lengkap anda adalah " << namaLengkap;

    return 0;
}

```

➤ MENGHILANGKAN NAMESPACE

Baris using namespace std dapat dihilangkan dan diganti dengan kata kunci std, dan diikuti oleh :: sebagai operator string dan cout objek.

```

#include <iostream>
#include <string>

using namespace std;

int main ()
{
    std::string greeting = "Hello World!";
    std::cout << greeting;
    return 0;
}

```

2.2 BOOLEANS

Booleans adalah suatu tipe data yang hanya mempunyai dua nilai yaitu true atau false. tipe data bool dapat mengambil nilai true (1) atau false (0). Ekspresi Boolean adalah ekspresi yang mengembalikan nilai boolean = 1 (benar) atau 0 (salah).

```

#include <iostream>
#include <string>

using namespace std;

int main ()
{
    int x = 1;
    int y = 2;

    cout << (x < y); // memnentukan benar atau salah pada perbandingan x & y
    return 0;
}

```

2.3 OPERATORS

Operator digunakan untuk melakukan operasi pada variabel dan nilai. Berikut jenis-jenis operator dan fungsinya.

- Operator Aritmatika
Operator aritmatika merupakan merupakan operator yang digunakan untuk melakukan operasi aritmatika

Operator Aritmatika	Nama
+	Penambahan
-	Pengurangan
*	Perkalian
/	pembagian
%	Modulus
++	Increment
--	Decrement

- Operator Relasional (Operator perbandingan)

Operator Relasional digunakan untuk menghubungkan atau membandingkan operan yang diberikan. Operasi relasional seperti memeriksa apakah memeriksa apakah dua operan sama atau tidak sama, lebih besar atau lebih kecil, dll.

Operator Relasional	Nama
<code>==</code>	Samadengan
<code>!=</code>	Tidak samadengan
<code>></code>	Lebih dari
<code><</code>	Kurang dari
<code>>=</code>	Lebih dari sama dengan
<code><=</code>	Kurang dari sama dengan

- Operator Logika (Operator Boolean)

Operator Logika adalah operator yang digunakan untuk melakukan operasi yang dimana nilai yang dihasilkan dari operasi tersebut berupa nilai true atau false.

Operator Logika	Nama
<code>&&</code>	Dan
<code> </code>	Atau
<code>!</code>	Tidak

- Operator Bitwise

Operasi bitwise adalah operator khusus untuk menangani operasi logika bilangan biner dalam bentuk bit.

Operator Bitwise	Nama
<code>&</code>	Dan
<code> </code>	Atau
<code>^</code>	Atau eksklusif
<code>~</code>	Bukan
<code><<</code>	shift kiri
<code>>></code>	shift kanan

- Operator Penugasan

Operator penugasan adalah operator untuk memasukkan suatu nilai ke dalam variabel.

Operator Penugasan	Contoh	Sama seperti
=	<code>x = 1</code>	<code>x = 1</code>
+=	<code>x += 1</code>	<code>x = x + 1</code>
-=	<code>x -= 1</code>	<code>x = x - 1</code>
*=	<code>x *= 1</code>	<code>x = x * 1</code>
/=	<code>x /= 1</code>	<code>x = x / 1</code>
%=	<code>x %= 1</code>	<code>x = x % 1</code>
&=	<code>x &= 1</code>	<code>x = x & 1</code>
=	<code>x = 1</code>	<code>x = x 1</code>
^=	<code>x ^= 1</code>	<code>x = x ^ 1</code>
>>=	<code>x >>= 1</code>	<code>x = x >> 1</code>
<<=	<code>x <<= 1</code>	<code>x = x << 1</code>

LAB ACTIVITY

- 1) Membuat program dengan menggunakan salah satu operator.

Contoh operator aritmatika:

```

1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main(int argc, char** argv)
7  {
8      int nilaiPertama=1, nilaiKedua=1;
9      cout << "nilai pertama + nilai kedua = " << (nilaiPertama+nilaiKedua);
10     return 0;
11 }
```

Cobalah dengan memperbaiki program berikut ini (pilih salah satu program C/C++).

C++ Program:

```

1  #include <iostream>
2  #include <string>
3
4
5  int main(int argc, char** argv)
6  {
7      int angkakedua, ketiga
8      cout >> "Hasil dari nilai kedua + nilai ketiga = "(nilaiketiga+kedua)
9  }
```

C Program:

```

1  #include <stdio.h>
2  #include <string.h>
3
4  int main(int argc, char** argv)
5  {
6      int angkakedua, ketiga;
7      printf("Hasil dari nilai kedua + nilai ketiga = , hasil);
8  }
```

- 2) Membuat program boolean.

```

1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main(int argc, char** argv)
7  {
8      int nilaiPertama=1, nilaiKedua=1;
9      cout << (nilaiPertama < nilaiKedua);
10     return 0;
11 }
```

Cobalah dengan memperbaiki program berikut ini (pilih salah satu program C/C++).

C++ Program:

```

1  #include <stdio.h>
2  #include <string.h>
3
4  int main(int argc, char** argv)
5  {
6      int angkakedua, ketiga;
7      printf(Hasil perbandingan dari nilai kedua > nilai ketiga)
8
9      return 0;
10 }

```

C Program:

```

1  #include <stdio.h>
2  #include <string.h>
3
4  int main(int argc, char** argv)
5  {
6      int angkakedua, ketiga;
7      printf(Hasil perbandingan dari nilai kedua > nilai ketiga)
8
9      return 0;
10 }

```

3) Membuat program dengan menggabungkan operator dengan inputan user.

```

1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  int main(int argc, char** argv)
7  {
8      int nilaiPertama;
9      int nilaiKedua;
10     cout << "Cek Perbandingan\n\n";
11     cout << "Masukan nilai pertama:";
12     cin >> nilaiPertama;
13     cout << "Masukan nilai pertama:";
14     cin >> nilaiKedua;
15     cout << "Hasil perbandingan dari nilai pertama dan kedua adalah " << (nilaiPertama < nilaiKedua); //operator kurang dari (<)
16     return 0;
17 }
18
19 /*
20 note: apabila bernilai 0 maka false, dan jika bernilai 1 true
21 */

```

4) Penggunaan operator bitwise.

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int a = 181;
8      int b = 108;
9      int hasil;
10
11     hasil = a & b;
12     cout << "Hasil dari a & b : " << hasil << endl;
13
14     hasil = a | b;
15     cout << "Hasil dari a | b : " << hasil << endl;
16
17     hasil = a ^ b;
18     cout << "Hasil dari a ^ b : " << hasil << endl;
19
20     hasil = ~a;
21     cout << "Hasil dari ~a : " << hasil << endl;
22
23     hasil = a >> 1;
24     cout << "Hasil dari a >> 1 : " << hasil << endl;
25
26     hasil = b << 2;
27     cout << "Hasil dari b << 2 : " << hasil << endl;
28
29     return 0;
30 }
31
32 /*
33
34 note:
35 > endl merupakan suatu fungsi manipulator yang digunakan
36 untuk menyisipkan karakter NewLine atau mengatur pindah baris.
37
38 */

```

Cobalah dengan memperbaiki program berikut ini (pilih salah satu program C/C++).

C++ Program:

```

1  #include <iostream>
2
3  using namespace std;
4
5  int main()
6  {
7      int a = 0;
8      int b = 1;
9      int hasil;
10
11     hasil = a & c;
12     cout << "Hasil dari a & b : " << hasil << endl;
13
14     hasil = a | b;
15     cout << "Hasil dari a | b : " << hasil << endl;
16
17     hasil = b ^ b;
18     scanf << "Hasil dari a ^ b : " << hasil << endl;
19
20     hasil = ~b;
21     cout << "Hasil dari ~a : " << hasil << endl;
22
23     hasil = a >> 1;
24     printf << "Hasil dari a >> 1 : " << hasil << endl;
25
26     hasil = a << 2;
27     cout << "Hasil dari b << 2 : " << hasil << endl;
28
29     return 1;
30 }

```

C Program:

```

1  #include <stdio.h>
2
3
4  int main()
5  {
6      int a = 1;
7      int b = 0;
8      int hasil;
9
10     hasil = a & b;
11     printf ("Hasil dari a & b : %i", hasil);
12
13     hasil = a | b;
14     cout << "Hasil dari a | b : ", hasil;
15
16     hasil = b ^ a;
17     scanf ("Hasil dari a ^ b : "hasil) ;
18
19     hasil = ~b;
20     cout ("Hasil dari ~a : " << hasil);
21
22     hasil = a >> 1;
23     printf ("Hasil dari a >> 1 : " hasil ;
24
25     hasil = a << 2;
26     cin "Hasil dari b << 2 : "
27
28     return 0;
29 }

```

5) Membuat program manipulasi string

```

1  #include <iostream>
2  #include <string.h>
3  #include <ctype.h>
4
5  using namespace std;
6
7  int main()
8  {
9      char a[20], b[20];
10
11     cout << endl;
12     cout << "Masukkan nama pertama: ";
13     cin >> a;
14     cout << "Masukkan nama kedua: ";
15     cin >> b;
16     strcat(a, b);
17     cout << "Nama Lengkap anda " << a;
18     cout << endl;
19
20     return 0;
21 }

```

Cobalah dengan memperbaiki program berikut ini (pilih salah satu program C/C++).

C++ Program:

```

1  #include <iostream>
2  #include <string.h>
3  #include <ctype.h>
4
5  using namespace std;
6
7  int main()
8  {
9      char a, b;
10
11      cout << endl;
12      cout << "Masukkan nama kedua ";
13      printf >> b;
14      cout << "Masukkan nama pertama: ";
15      cin >> a;
16      strcpy(b, a);
17      scanf << "Nama Lengkap anda " << a;
18      cout << endl;
19
20      return 0;
21  }

```

C Program:

```

1  #include <stdio.h>
2  #include <string.h>
3  #include <ctype.h>
4
5  using namespace std;
6
7  int main()
8  {
9      char a, b;
10
11      printf("/n");
12      printf("Masukkan nama kedua ");
13      printf >> b;
14      cout << "Masukkan nama pertama: ";
15      cin >> a;
16      strlen(b, c);
17      scanf ("Nama Lengkap anda ", a);
18      return 0;
19  }

```

TUGAS PRAKTIKUM

Berdasarkan tema tugas besar yang kalian pilih, silakan mengembangkan program kalian dengan kriteria sebagai berikut:

- Telah mengimplementasikan operator, boolean, dan string
- Telah menggunakan syntax programming dengan benar
- Telah mengimplementasikan comments dengan benar
- Telah mendeklarasikan variabel dengan aturan yang benar
- Telah mengimplementasikan sebuah konstanta dalam program
- Telah mengimplementasikan sebuah tipe data beserta value ke dalam program

DETAIL PENILAIAN PRAKTIKUM

Ketentuan	Bobot Penilaian
Dapat mengimplementasikan materi dari modul praktikum	20%
Dapat menjelaskan program dan materi dari modul praktikum	40%
Program berhasil berjalan	10%
Menjawab pertanyaan asisten	10%