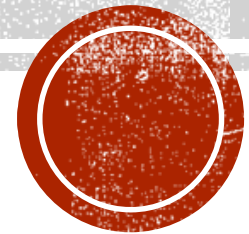


# DASAR STRING

PEMROGRAMAN LANJUT  
TEKNIK MEKATRONIKA



# PENGANTAR STRING

- String sangat memudahkan tugas pemrogram
- String dapat digunakan untuk menampilkan pesan kesalahan, menampilkan prompt masukan dari keyboard, ataupun memberikan informasi pada layar dengan mudah
- String dapat berupa konstanta atau variable
- Konstanta string telah sering digunakan, contoh :

```
cout << "C++" << endl ;
```



# KONSTANTA STRING

- Suatu konstanta string ditulis dengan awalan dan akhiran tanda petik ganda (“”).
- Konstanta string seperti di atas disimpan dalam memori secara berurutan



- Setiap karakter menempati memori sebesar 1 byte.
- Setelah karakter yang terakhir disebut karakter NULL (karakter dengan nilai ASCII sama dengan nol atau disimbolkan dengan ‘\0’).
- Bila suatu string hanya berisi karakter NULL, string disebut sebagai string kosong atau “”
- Karakter NULL merupakan ciri khas pengakhir string pada C++



# VARIABEL STRING

- Variabel string adalah variable yang dipakai untuk menyimpan string,
- `char teks[10]`
- Merupakan pernyataan untuk mendefinisikan variable string dengan panjang maksimal 10 karakter, sudah termasuk NULL
- Pernyataan di atas tidak lain adalah untuk mendefinisikan array bertipe karakter
- Variabel string tidak lain adalah array bertipe char yang mengandung elemen berupa karakter NULL, yang menyatakan akhir string



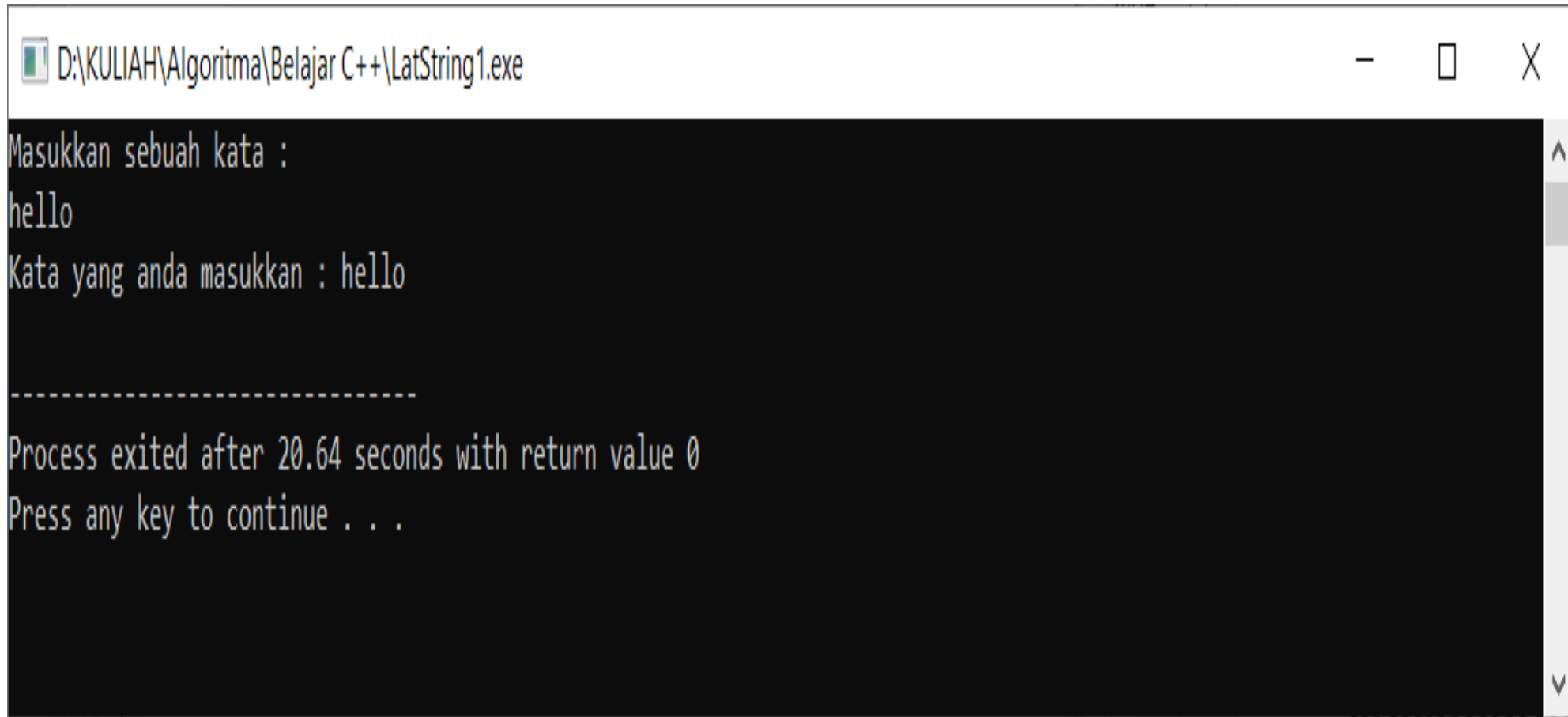
# MEMASUKKAN DATA STRING DARI KEYBOARD

- Contoh program :

```
char teks[13];  
    cout << "Masukkan sebuah kata : " << endl;  
    cin >> teks;  
    cout << "Kata yang anda masukkan : " << teks << endl;
```



# DIMASUKKAN 1 KATA



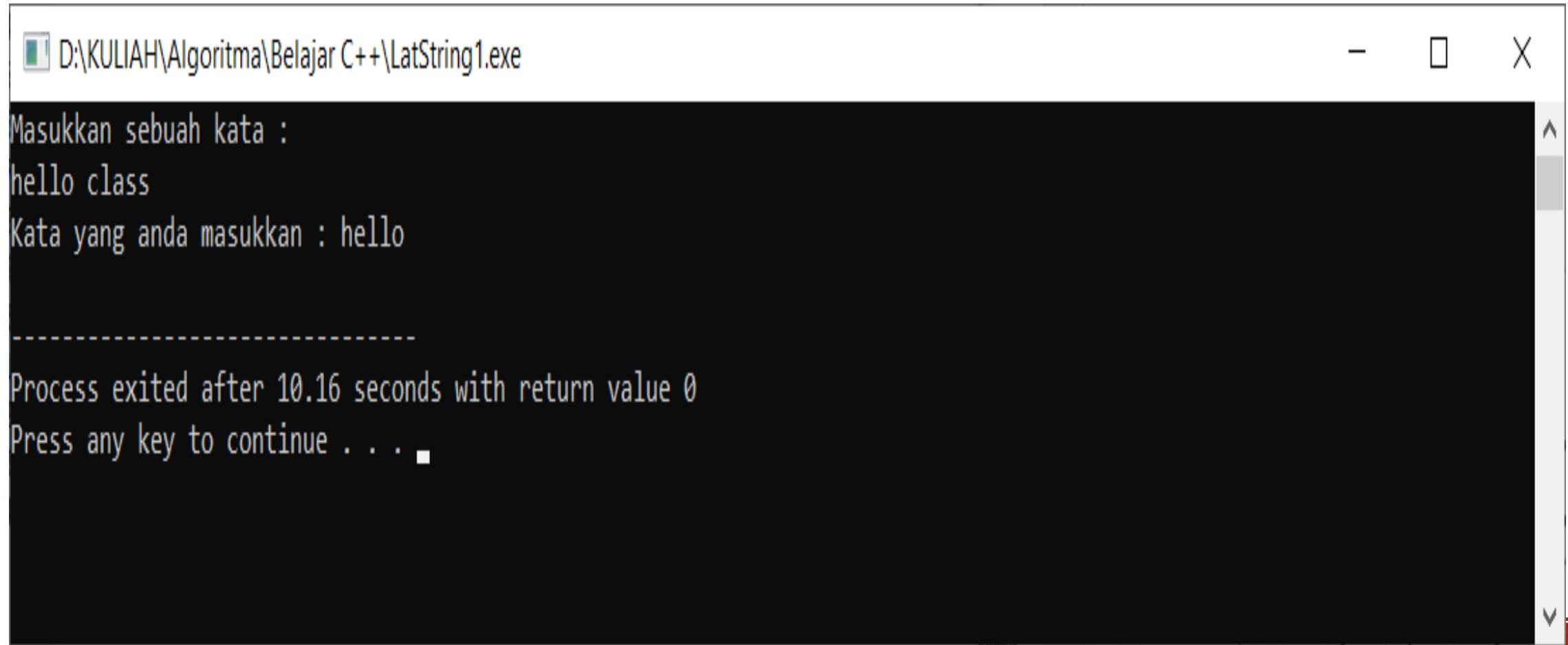
A screenshot of a Windows command prompt window. The title bar at the top shows the file path "D:\KULIAH\Algoritma\Belajar C++\LatString1.exe" and standard window controls (minimize, maximize, close). The command prompt has a black background with white text. The text inside the window shows the program's execution: it prompts "Masukkan sebuah kata :", the user enters "hello", and the program outputs "Kata yang anda masukkan : hello". Below this, there is a dashed line, followed by the message "Process exited after 20.64 seconds with return value 0" and "Press any key to continue . . .". A vertical scrollbar is visible on the right side of the command prompt window.

```
D:\KULIAH\Algoritma\Belajar C++\LatString1.exe
Masukkan sebuah kata :
hello
Kata yang anda masukkan : hello

-----
Process exited after 20.64 seconds with return value 0
Press any key to continue . . .
```



# DIMASUKKAN 2 KATA



```
D:\KULIAH\Algoritma\Belajar C++\LatString1.exe
Masukkan sebuah kata :
hello class
Kata yang anda masukkan : hello
-----
Process exited after 10.16 seconds with return value 0
Press any key to continue . . .
```

# MENGUNAKAN GET()

```
char teks[13];
```

```
cout << "Masukkan sebuah kata : " << endl;
```

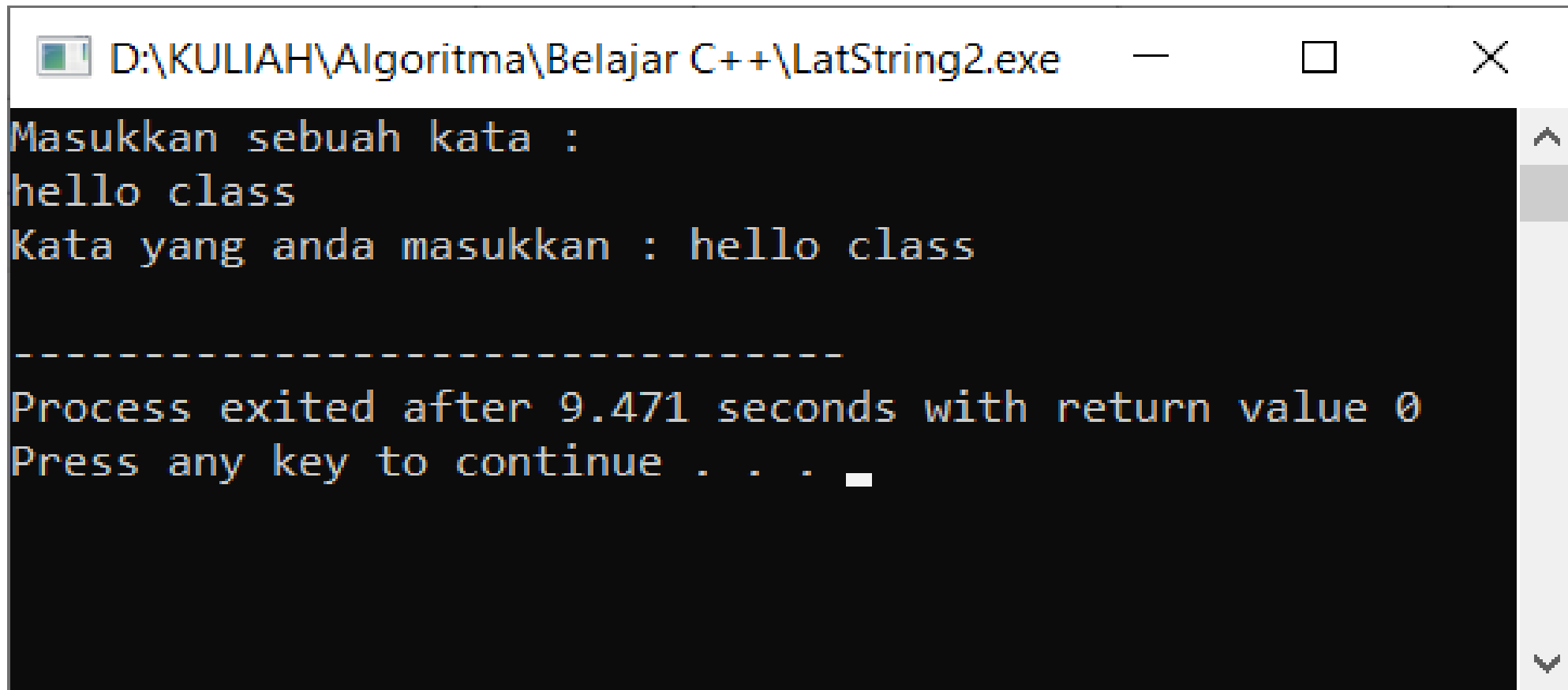
```
cin.get(teks, 13);
```

```
cout << "Kata yang anda masukkan : " << teks << endl;
```





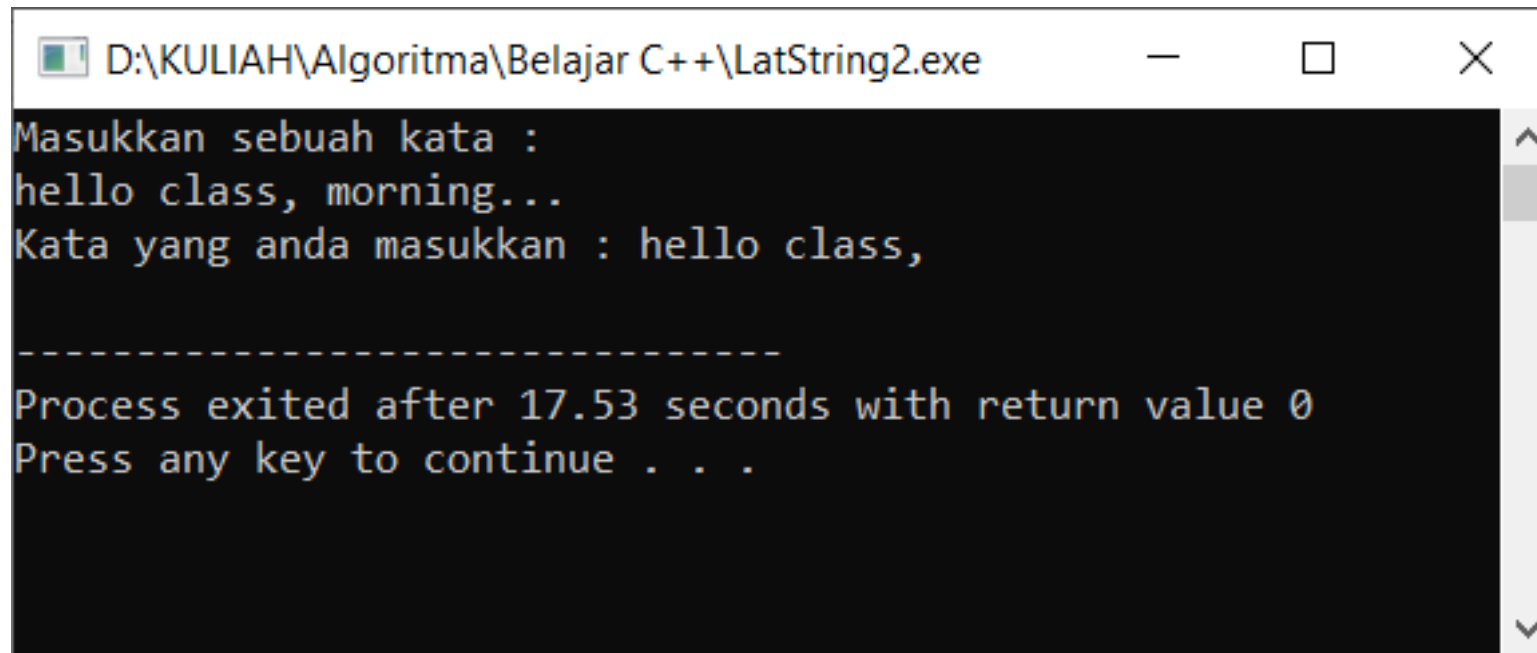
# HASIL RUNNING DENGAN INPUT 2 KATA



```
D:\KULIAH\Algoritma\Belajar C++\LatString2.exe
Masukkan sebuah kata :
hello class
Kata yang anda masukkan : hello class
-----
Process exited after 9.471 seconds with return value 0
Press any key to continue . . .
```



- Perhatikan! Ada bagian kata yang terpotong dari sejumlah kata input dari user.
- Maka untuk menampilkan kata secara utuh, simak slide berikutnya



```
D:\KULIAH\Algoritma\Belajar C++\LatString2.exe
Masukkan sebuah kata :
hello class, morning...
Kata yang anda masukkan : hello class,
-----
Process exited after 17.53 seconds with return value 0
Press any key to continue . . .
```



# PENGUNAAN SIZEOF

```
char teks[30];
```

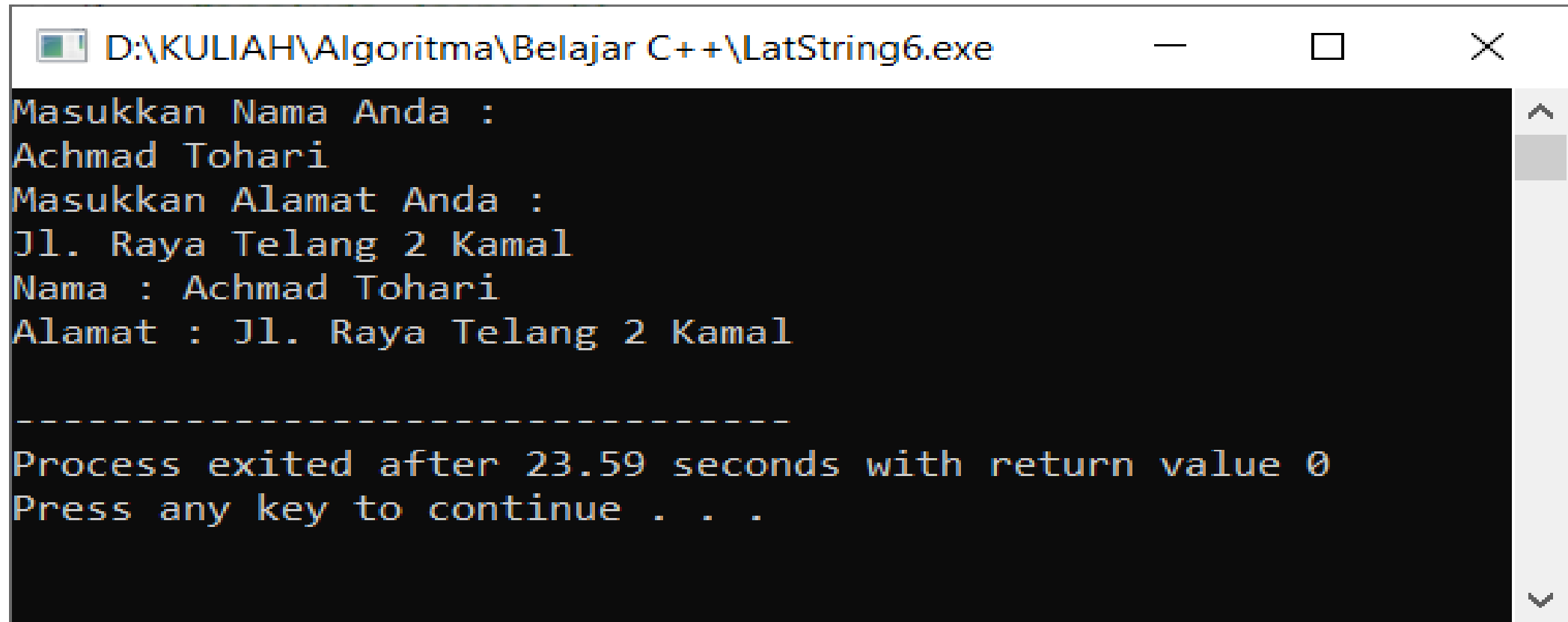
```
cout << "Masukkan sebuah kata : " << endl;
```

```
cin.get(teks, sizeof(teks));
```

```
cout << "Kata yang anda masukkan : " << teks << endl;
```



# BUAT PROGRAM DENGAN OUTPUT :



```
D:\KULIAH\Algoritma\Belajar C++\LatString6.exe
Masukkan Nama Anda :
Achmad Tohari
Masukkan Alamat Anda :
Jl. Raya Telang 2 Kamal
Nama : Achmad Tohari
Alamat : Jl. Raya Telang 2 Kamal

-----
Process exited after 23.59 seconds with return value 0
Press any key to continue . . .
```



# YOUR PROGRAM? HASILNYA?

```
char nama[30];  
char alamat[45];  
  
cout << "Masukkan Nama Anda : " << endl;  
cin.get(nama, sizeof(nama));  
cout << "Masukkan Alamat Anda : " << endl;  
cin.get(alamat, sizeof(alamat));  
  
cout << "Nama : " << nama << endl;  
cout << "Alamat : " << alamat << endl;
```



# REVISI DENGAN FUNGSI ANGGOTA GETLINE

- Your Revisi Program?



```
char nama[30];  
char alamat[45];
```

```
cout << "Masukkan Nama Anda : " << endl;  
cin.getline(nama, sizeof(nama));  
cout << "Masukkan Alamat Anda : " << endl;  
cin.getline(alamat, sizeof(alamat));
```

```
cout << "Nama : " << nama << endl;  
cout << "Alamat : " << alamat << endl;
```



# MENGHITUNG JUMLAH KARAKTER

```
jumlah = 0  
for (int i = 0; teks[i]; i++)  
    jumlah ++;
```







D:\KULIAH\Algoritma\Belajar C++\LatString7.exe



Masukkan String apa saja :

Algoritma Pemrograman

Jumlah karakter = 21

-----

Process exited after 7.736 seconds with return value 0

Press any key to continue . . .



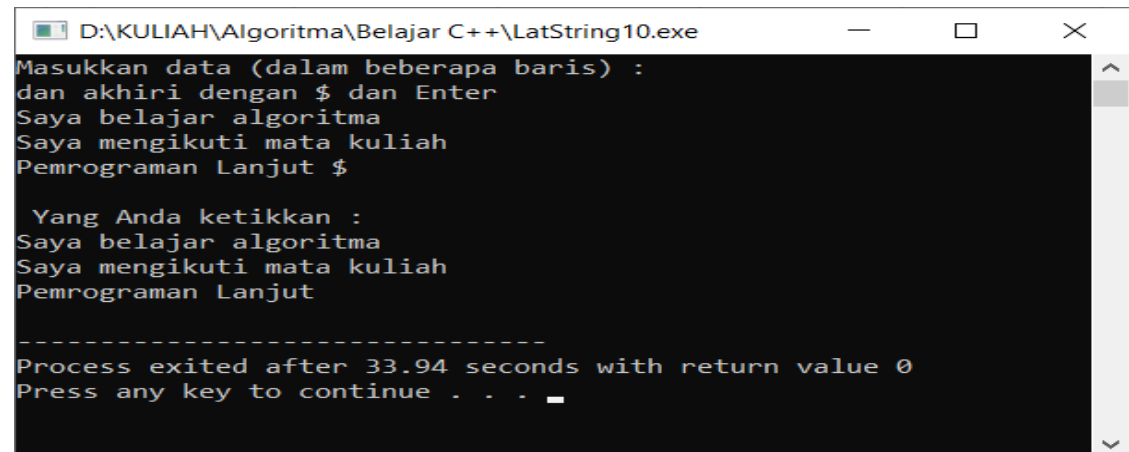
```
#include<iostream>
#include <conio.h>
using namespace std;
int main(){
    char teks[128];
    int jumlah;
    cout << "Masukkan String apa saja : " << endl;
    cin.getline(teks, sizeof(teks));
    cout << "Jumlah karakter = " << cin.gcount() -1 << endl;
}
```

```
include<iostream>
#include <conio.h>
#include<string.h>
using namespace std;
int main(){
    char teks[128];
    int jumlah;
    cout << "Masukkan String apa saja : " << endl;
    cin.getline(teks, sizeof(teks));
    cout << "Jumlah karakter = " << strlen(teks)<< endl;
}
```



# MEMBACA SEJUMLAH BARIS

- `getline()` juga bisa dipakai untuk membaca sejumlah baris
- Harus menyertakan argument ke-3 pada fungsi tersebut, berupa karakter pengakhir, misal : `@`, `$`
- `cin.getline(teks, sizeof(teks), '$');` akan menyebabkan seluruh karakter yang terletak sebelum tanda `$` diletakkan ke teks
- Jika jumlah karakter melebihi ukuran array teks, sisanya akan diabaikan



```
D:\KULIAH\Algoritma\Belajar C++\LatString10.exe
Masukkan data (dalam beberapa baris) :
dan akhiri dengan $ dan Enter
Saya belajar algoritma
Saya mengikuti mata kuliah
Pemrograman Lanjut $

Yang Anda ketikkan :
Saya belajar algoritma
Saya mengikuti mata kuliah
Pemrograman Lanjut

-----
Process exited after 33.94 seconds with return value 0
Press any key to continue . . .
```



```
#include<iostream>
#include <conio.h>
using namespace std;

int main()
{
    char teks[128];

    cout << "Masukkan data (dalam beberapa baris) : " << endl;
    cout << "dan akhiri dengan $ dan Enter" << endl;

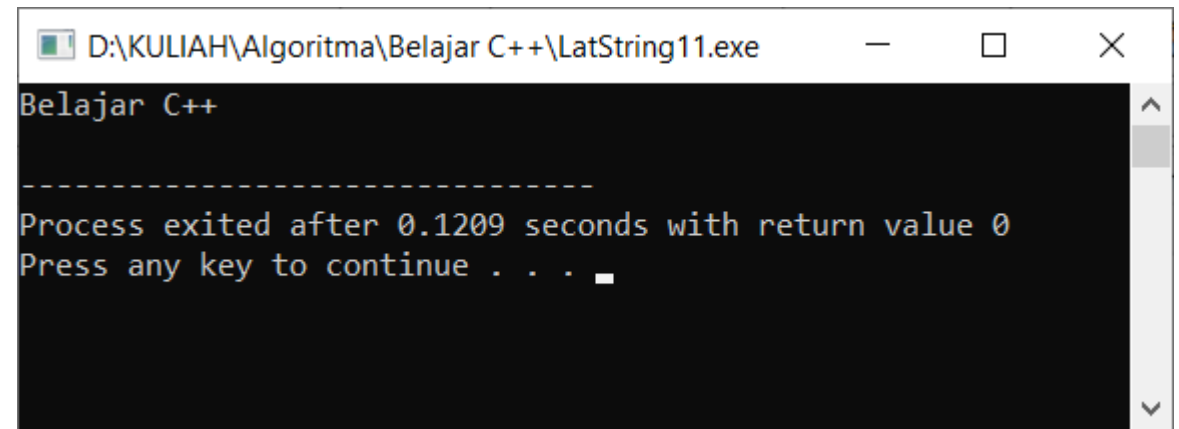
    cin.getline(teks, sizeof(teks), '$');

    cout << "\n Yang Anda ketikkan : " << endl;
    cout << teks << endl;

}
```



# INISIALISASI STRING



```
D:\KULIAH\Algoritma\Belajar C++\LatString11.exe
Belajar C++
-----
Process exited after 0.1209 seconds with return value 0
Press any key to continue . . . _
```

- Suatu string dapat diberi nilai awal seperti halnya variable lain
- Oleh karena variable string sebenarnya adalah jenis khusus dari array, maka inisialisasi dapat berupa seperti berikut :

```
char Bahasa[] = 'B', 'e', 'l', 'a', 'j', 'a', 'r', 'C', '+', '+', '\0');
```

- Tampak bahwa elemen terakhir pada pendefinisian di atas berupa karakter NULL.
- Bentuk pendefinisian dan inisialisasi seperti di atas dapat disederhanakan menjadi :

```
char bahasa[] = "Belajar C++";
```



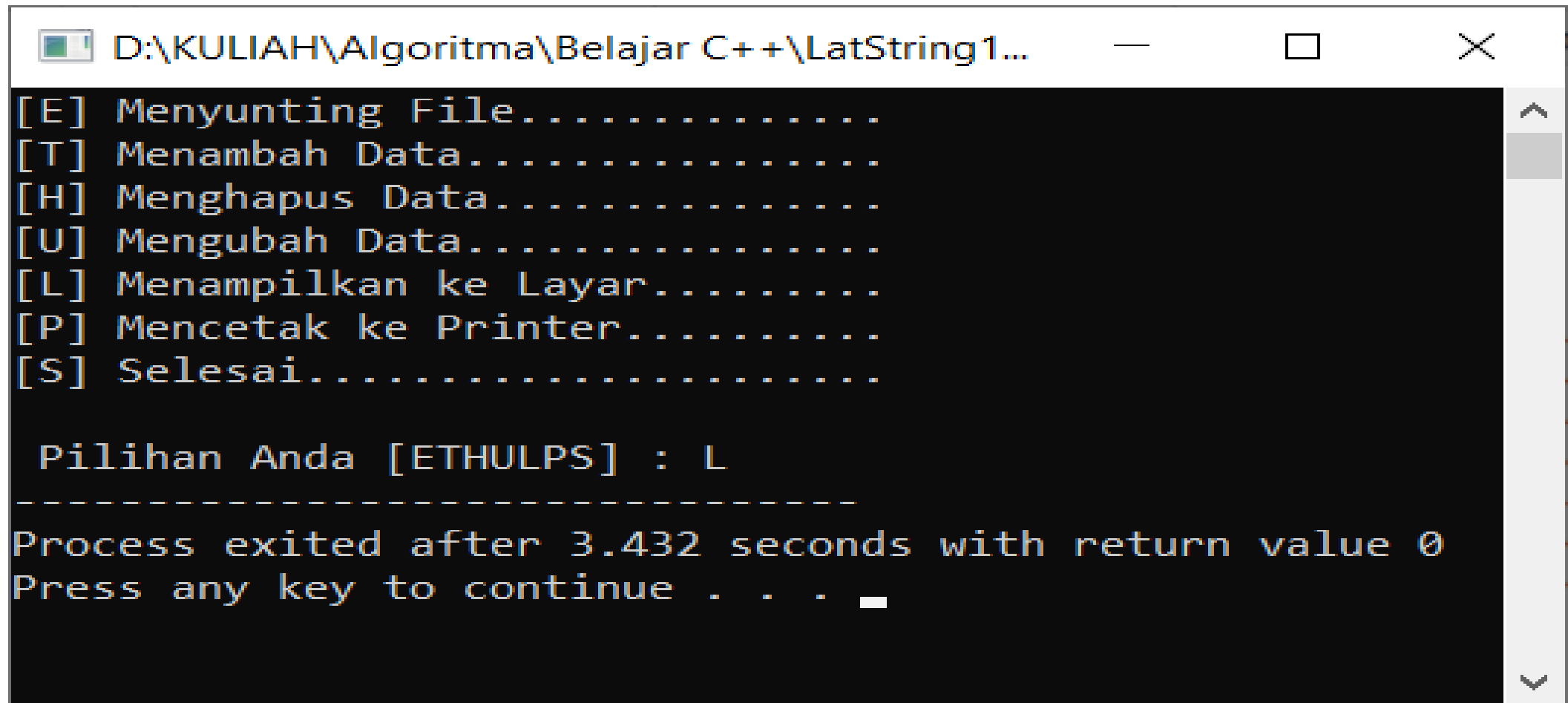
```
#include<iostream>
#include <conio.h>
using namespace std;

int main()
{
    char bahasa[] = "Belajar C++";

    cout << bahasa << endl;
}
```



# MENCARI SUATU KARAKTER PADA STRING



```
D:\KULIAH\Algoritma\Belajar C++\LatString1...
[E] Menyunting File.....
[T] Menambah Data.....
[H] Menghapus Data.....
[U] Mengubah Data.....
[L] Menampilkan ke Layar.....
[P] Mencetak ke Printer.....
[S] Selesai.....

Pilihan Anda [ETHULPS] : L
-----
Process exited after 3.432 seconds with return value 0
Press any key to continue . . .
```



```
#include<iostream>
#include <conio.h>
#include<string.h>
#include<ctype.h>

using namespace std;

int main()
{
    char pilihan;

    cout << "[E] Menyunting File....." << endl;
    cout << "[T] Menambah Data....." << endl;
    cout << "[H] Menghapus Data....." << endl;
    cout << "[U] Mengubah Data....." << endl;
    cout << "[L] Menampilkan ke Layar....." << endl;
    cout << "[P] Mencetak ke Printer....." << endl;
    cout << "[S] Selesai....." << endl;
    cout << "\n Pilihan Anda [ETHULPS] : ";

    do
    {
        pilihan = toupper(getch());
    }
    while (!strchr("ETHULPS", pilihan));
    cout << pilihan;

}
```





- Fungsi `strchr()` untuk mencari suatu karakter pada suatu string
- Pada program sebelumnya `strchr()` digunakan untuk memeriksa isi pilihan terhadap string “ETHULPS”.
- Pointer menunjuk ke NULL kalau isi pilihan tidak terdapat pada string “ETHULPS”
- Pointer yang menunjuk ke karakter pertama pada “ETHULPS” yang cocok dengan isi pilihan.
- Nilai balik dari fungsi `strchr()` berupa pointer yang menunjuk ke string
- Pointer dibahas pada bab berikutnya
- Misalkan didefinisikan :

`char *p;` //didepan p ada \*

- Pada pendefinisian ini p berkedudukan sebagai pointer, bila kemudian terdapat perintah :

`p = strchr( “ETHULPS” , pilihan);`

- Dan **pilihan** sendiri berisi **U**, p akan menunjuk ke karakter U pada string “ETHULPS”.



# MENCARI SUBSTRING

```
D:\KULIAH\Algoritma\Belajar C++\LatString13.exe
Substring BANGKA terdapat pada BANGKALAN

-----
Process exited after 0.04516 seconds with return value 0
Press any key to continue . . .
```

```
#include <iostream>
#include <conio.h>
#include <string.h>

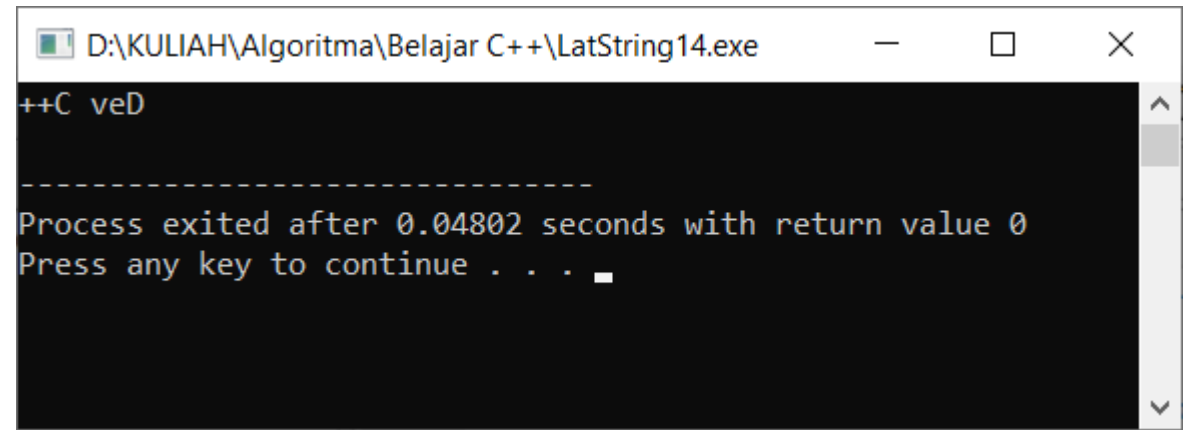
using namespace std;
int main()
{
    if(strstr("BANGKALAN", "BANGKA"))
        cout << "Substring BANGKA terdapat pada BANGKALAN" << endl;
    if(strstr("BANGKALAN", "bangka"))
        cout << "Substring bangka terdapat pada BANGKALAN" << endl;
}
```



- Selain untuk keperluan mencari suatu karakter dalam suatu string, juga terdapat fungsi yang berguna untuk mencari suatu substring.
- Fungsi tersebut adalah **strstr()**
- Tampak bahwa string : **Substring bangka terdapat pada BANGKALAN** tidak ditampilkan
- Ini disebabkan memang pada string **“BANGKALAN”** tidak terdapat substring **“Bangka”** tetapi substring **“BANGKA”** ada.
- Hasil exe menunjukkan bahwa fungsi **strstr()** memberikan nilai balik berupa benar kalau substring argument kedua terdapat pada string argument pertama.



# MEMBALIK STRING



```
D:\KULIAH\Algoritma\Belajar C++\LatString14.exe
++C veD
-----
Process exited after 0.04802 seconds with return value 0
Press any key to continue . . . _
```

- Isi dari suatu string dapat dibalik dengan mudah melalui perantaraan fungsi **strrev()**
- Misalnya : string “Dev C++” akan dibalik menjadi “++C veD”
- **Buat program untuk menghasilkan output sebagaimana ditampilkan pada exe.**



```
#include <iostream>
#include <conio.h>
#include <string.h>

using namespace std;
int main()
{
    char kompiler[] = "Dev C++";
    strrev(kompiler);
    cout << kompiler << endl;
}
```



# KONVERSI STRING KE ANGKA DAN SEBALIKNYA

- Untuk melakukan konversi dari suatu string ke bilangan, pemrogram dapat menggunakan sejumlah fungsi bawaan
- Fungsi-fungsi tersebut :

Fungsi	Prototipe	Keterangan
atoi()	stdlib.h	Mengonversi string argument menjadi nilai bertipe int
atof()	stdlib.h	Mengonversi string argument menjadi nilai bertipe float
atol()	stdlib.h	Mengonversi string argument menjadi nilai bertipe long int
_atold()	stdlib.h	Mengonversi string argument menjadi nilai bertipe long double  (Hanya terdapat pada Borland C++)



- Kegunaan fungsi-fungsi di atas diantaranya untuk mengonversi string argument baris perintah menjadi suatu nilai bilangan (argument baris perintah)
- Sedangkan fungsi untuk mengubah nilai bilangan menjadi suatu string :

Fungsi	Prototipe	Keterangan
itoa()	stdlib.h	Mengonversi int menjadi string
ltoa()	stdlib.h	Mengonversi long int menjadi string
ultoa()	stdlib.h	Mengonversi unsigned long menjadi string



## TABEL SEBELUMNYA MEMPUNYAI 3 ARGUMEN :

- Argumen pertama berupa nilai yang akan dikonversi ke string
- Argumen kedua berupa variable penerima string hasil konversi
- Argumen ketiga berupa basis bilangan yang digunakan, bisa diisi dengan nilai antara 2 sampai dengan 36.
- Misal, apabila hasil didefinisikan sebagai berikut :

char hasil[24];

int nilai = 2345;

pernyataan :

itoa(nilai, hasil, 10)

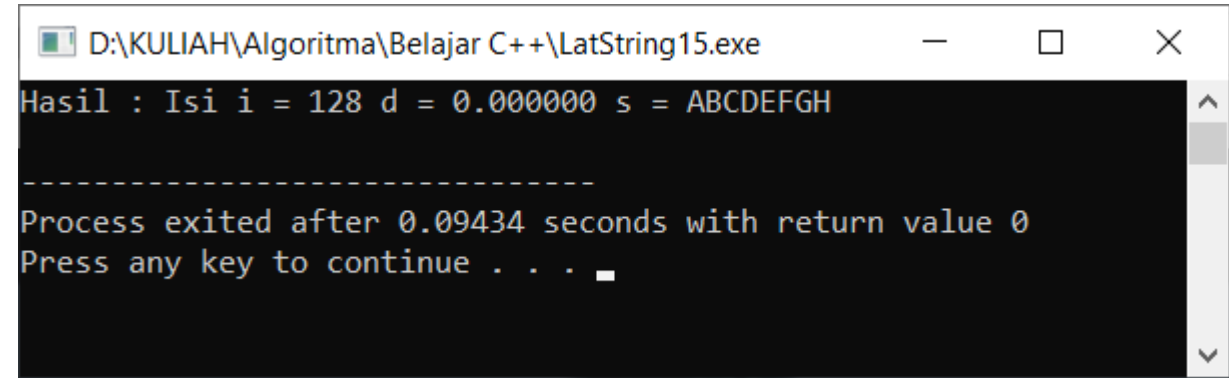
Membuat hasil berisi "2345"

- Pada C++ terdapat fungsi bernama sprintf()
- Berguna untuk memformat satu atau beberapa data menjadi sebuah string
- Format yang digunakan serupa fungsi printf() (yang biasa dipakai untuk menampilkan data pada Bahasa C)
- Prototipe ada pada file stdio.h





# CONTOH



```
D:\KULIAH\Algoritma\Belajar C++\LatString15.exe
Hasil : Isi i = 128 d = 0.000000 s = ABCDEFGH

-----
Process exited after 0.09434 seconds with return value 0
Press any key to continue . . .
```

- Pada contoh, pernyataan :
- `printf(hasil, "Isi I = %d d = %lf s = %s", i, d, s);`
- Akan menyebabkan hasil berisi string :
- "Hasil : Isi I = 128 d = 128.997766 s = ABCDEFGH"
- Simbol `%d`, `%lf` dan `%s` pada argumen kedua `printf()` digunakan untuk menyatakan tipe data yang akan diberikan ke string hasil
- Beberapa simbol yang digunakan untuk memformat data pada fungsi `printf()` :
- `%d` data bertipe **int**
- `%f` data bertipe **double**
- `%c` data bertipe **char**
- `%ld` data bertipe **long int**
- `%lf` data bertipe **long double**
- `%s` data bertipe **string**



```
int main()
{
    char hasil[128];
    char s[] = "ABCDEFGH";
    int i = 128;
    long double d = 128.997766;

    sprintf(hasil, "Isi i = %d d = %lf s = %s", i, d, s);
    cout << "Hasil : " << hasil << endl;
}
```

