Pointer dengan string adalah pemanfaatan penggunaan dari pointer pada string yang merupakan kumpulan dari beberapa karakter dan disimpan di dalam lokasi memori yang berdekatan. Untuk melakukan itu biasanya kita menggunakan tipe data char dengan fungsi array yang membuatnya menjadi sebuah urutan karakter yang berdekatan di dalam lokasi memori.

Tipe data char adalah salah satu tipe data dari bahasa pemrograman yang berfungsi untuk menyimpan data dalam bentuk karakter seperti a-z dan tanda karakter lainya, di dalam memori komputer. Array atau Variabel Larik berfungsi untuk menyimpan serangkaian elemen dari jenis yang sama ditempatkan di lokas memori yang berdekatan dengan elemen lainya. dari kedua fungsi tersebut kita bisa gunakan untuk membuat kumpulan karakter atau string. dan sebenarnya array dan pointer adalah sama. Mngkin penjelasan ini akan terdengar sama seperti Pointer dengan Array tapi penjelasan ini akan dikhususkan pada string.

pada umumnya konsep pointer mirip seperti konsep array, terutama pada penggunaan string. Pointer adalah sebuah variabel yang berisi alamat dari variabel lain, Atau dengan kata lain variabel pointer itu menunjuk ke variabel lainya. bedanya pointer adalah pointer hanya memegang alamat mereka tidak menampung semua karakter dalam array karakter, ini berarti bahwa ketika kita menggunakan char\* untuk menunjuk string, karakter yang termasuk string harus sudah ada dalam memori yang secara static atau dinamis

Bentuk Umum Penulisan

//Pointer dereference

char \*[identifier] = [Variabel String] ;

//Pointer Reference

char &[ptrString] = [Variabel String];

Contoh Penulisan

// Poiter Dereerence

char \*ptrString = arrString ;

// Pointer Reference

char &ptrString = arrString ;

Contoh Program

#include <iostream>

using namespace std;

int main( )

{

char arrString[]="IniBelajarCPP";

char \*ptrString=arrString;

cout<<"Eksekusi Array : "<<endl;

cout<<arrString<<endl;

cout<<arrString[3]<<endl; //Subscript

cout<<arrString+3<<endl; //Aritmatika

cout<<\*(arrString+3)<<endl; //Aritmatika

cout<<"Eksekusi Pointer : "<<endl;

cout<<ptrString<<endl;

cout<<ptrString[3]<<endl; //Subscript

cout<<ptrString+3<<endl; //Aritmatika

cout<<\*(ptrString+3)<<endl; //Aritmatika

return 0;

}

untuk ponter reference penulis rasa itu sudah sangat jelas karena itu hanya sebuah alias/nama lain dari variabel yang ditunjuk, Kali ini kita akan membahas mengenai pointer dereference dengan string. pada penjelasan dan contoh di atas terlihat seperti Type data char di spesialkan oleh pointer dan array.

pada contoh diatas didirikan string pointer dengan nilai / menunjuk ke string yang sudah ada sebelumnya pada string array. Dan di bawah ini adalah gambar illustrasi apa yang sebenarnya terjadi.

---------------------------Gambar 6.2

Ingat, jika pemberian tugas atau inisialisasi pada pointer dengan array kita tidak menggunakan operator Address-of ( & ). Bisa kita buktikan dengan mencetak variabel array selain tipe data char tanpa menggunakan notasi subscript / index untuk penentu elemen, apa yang terjadi adalah program akan mencetak alamat memori dari elemen pertama pada array tersebut begitu juga dalam operasi penugasan. Ini akan berlangsung berbeda jika kita menggunakan tipe data char, apa yang terjadi akan mencetak semua isi elemen dan menjadikanya sebuah string. Walaupun seperti itu, ini akan berlaku sama dalam operasi penugasan / inisialisasi seperti type data lainya. Variabel pointer hanya akan mendapatkan alamat memori dari elemen pertama array.

Dan setelah pemberian penugasan, pengaksesan Pointer “ptrString” akan bersifat sama seperti pengaksesan array “arrString”. Untuk itu bisa kita lihat pada contoh program di atas. Kesimpulanya adalah array dan pointer itu sama.