Shandy Ilham Alamsyah\_015

Tugas Pemrograman Dasar (06 Sep 2021)

* Macam – macam Tipe Data di C++:

|  |  |  |
| --- | --- | --- |
| Nama Tipe Data | Ukuran / range | Deskripsi |
| **void** (kosong) | 0 byte /  - | Tipe data yang tidak bertipe biasanya digunakan untuk tipe data kosong seperti membuat fungsi. |
| **Int** (interger) | 4 byte /  -2147483648 sampai 2147483647 | Untuk diisi angka (bukan desimal), misal 1, 30, 500, dll. |
| **Float** | 4 byte /  -3,4 x 1038 hingga 3,4 x 1038 | Untuk diisi angka desimal dan hanya cukup untuk menyimpan sampai 7 digit. |
| **Double** | 8 byte /  -1,8 x 10308 hingga 1,8 x 10308 | Untuk diisi angka desimal dan hanya cukup untuk menyimpan sampai 15 digit. |
| **Boolean** | 1 byte /  True or false (1 or 0) | Untuk diisi salah satu dari nilai true atau false. |
| **char** (character) | 1 byte /  -127 sampai 127 atau 0 sampai 255 | Tipe data yang digunakan untuk menampung 1 digit karakter, misal A, b, dll. |
| **Wchar\_t** | 2 or 4 byte /  1 wide character | Tipe data yang digunakan untuk menyimpan karakter yang tidak ada di abjad biasa, misal huruf jepang, china, korea, dll. |
| **string** | 4 byte /  - | Tipe data yang merupakan kumpulan dari karakter, bisa berupa kata atau kalimat, misal “hello world”. |

* Macam – macam modifikasi tipe data :

|  |  |  |
| --- | --- | --- |
| Nama modify | Ukuran / Range | Deskripsi |
| **Byte** | 1 byte /  -128 hingga 127 | Tipe ini diperlukan saat kita melakukan proses seperti membaca dan menulis. Selain itu, byte juga digunakan saat kita menggunakan data biner yang tidak kompatibel dengan tipe lainnya pada program bahasa Java. |
| **Short** | 2 byte /  -32768 hingga 32767 | sangat umum digunakan pada komputer yang berkapasitas 16-bit, sehingga sangat jarang sekali kita temui dengan kapasitas spesifikasi komputer saat ini. |
| **Long** | 8 byte /  -9223372036854775808 hingga 9223372036854775807 | Digunakan saat rentang int diluar kapasitas, karena tipe long punya range sangat tinggi dibanding tipe data lainnya. Dengan kata lain, tipe long dibutuhkan saat data memiliki range di luar jangkauan tipe int, short, maupun byte. |

* PreProcessor

Preprocessor dalam C++ mempengaruhi proses yang dilakukan compiler pada kode program dengan menggantikan atau mengisi suatu bagian kode program sesuai perintah yang diberikan. Preprocessor diawali dengan "#". Ada beberapa preprocessor dalam C++, yaitu :

1. #include : Menyertakan isi file header ("\*.h") dalam kode program.
2. #define : Menggantikan identifier dalam program dengan makro atau suatu nilai.
3. #undef : Menghapus identifier yang sudah dibuat dengan #define.
4. #Pragma : Perintah khusus compiler yang bisa berbeda untuk setiap compiler.
5. Conditional Preprocessor ( #if, #elseif, #else) : Menentukan bagian program yang akan di-compile berdasarkan suatu persyaratan seperti OS atau compilernya.

Sebelumnya kita pernah menggunakan #include untuk menyertakan header standar. Ada dua cara menggunakan #include.

1. Jika #include menggunakan header standar C++ atau header default yang sudah ditentukan lokasinya, nama file diapit dengan "<" dan ">".
2. Jika #include digunakan untuk menambahkan header yang berada di dalam folder yang sama dengan kode program C++, nama file diapit diantara dua tanda kutip.

* Compiler

Compiler adalah program yang berfungsi untuk menerjemahkan bahasa C menjadi bahasa mesin, shingga dapat dimengerti oleh komputer.

Ada beberapa compiler yang bisa digunakan untuk compile program C++.

Ada GCC (Gnu Compiler Collection), Clang, Visual C++ 2017, Embarcadero, Oracle Solaris Studio C++ Compiler, dan IBM XL C/C++ Compiler.

* Assembler

Assembler merupakan sebuah program yang mengkonvrensi kode program kedalam mesin. Sedangkan bahasa Assembler merupakan bahasa yang merespon perintah dengan bahasa mesin untuk memahami intraksi antara perangkat keras dan perangkat lunak di komputer.

* Teks editor

Teks editor adalah program yang digunakan untuk menulis kode program C++. Contohnya Ada Notepad, Notepad++, Gedit, Geany, Mousepad, Kate, Atom, VS Code, Sublime Text, dsb.

* Header file

Header file merupakan file file yang berisikan tentang deklarasikan definisi berbagai file makro, fungsi, variable, dan knstanta pada program C++ dan berfungsi memanggil dan menjalankan fungsi yang terdapat dalam library file header tersebut. Berikut merupakan beberapa header file di C++:

1. <iostream.h> berisi perintah:

**Cin**  
Merupakan fungsi masukan (digunakan untuk menyimpan data dalam suatu variabel).  
Bentuk umum: cin>>var x;  
 **Cout**  
Merupakan fungsi keluaran (digunakan untuk menampilkan data ataupun tulisan). Bentuk umum: cout<<”tulisan”; atau cout<<var x;  
  
**Endl**  
Digunakan untuk pindah baris/ enter.  
Bentuk umum: cout<<”tulisan”<<endl;  
  
**Ends**  
Merupakan suatu fungsi manipulator yang digunakan untuk menambah karakter null (nilai ASCII NOL) ke deretan suatu karakter. Fungsi ini akan berguna untuk mengirim sejumlah karakter ke file di disk atau modem dan mangakhirinya dengan karakter NULL.

1. <conio.h> berisi perintah:

**Getch**  
Berfungsi untuk menahan tampilan.  
Bentuk umum: getch();  
  
**Clrscr**  
Berfungsi untuk membersihkan layar  
Bentuk umum: clrscr;  
  
**Getche**  
Fungsi dipakai untuk membaca sebuah karakter dengan sifat karakter yang dimasukkan tidak perlu diakhiri dengan menekan tombol ENTER, dan karakter yang dimasukan ditampilkan di layar.  
  
**Putch**  
Akan menampilkan karakter ASCII dari nilai x ke layer monitor tanpa memindahkan letak kursor ke baris berikutnya.  
  
**Clreol**  
Fungsi ini digunakan untuk membersihkan layar mulai dari posisi kursor hingga kolom terakhir, posisi kursor tidak berubah.  
  
**Gotoxy**  
Fungsi gotoxy digunakan untuk memindahkan kursor ke kolom x, baris y.  
  
**Wherex**  
Fungsi wherex digunakan untuk mengembalikan posisi kolom kursor.  
  
**Wherey**  
Fungsi wherey digunakan untuk mengembalikan posisi baris kursor.  
  
**Window**  
Fungsi window digunakan untuk mendefinisikan sebuah window berdasarkan koordinat kiri atas dan kanan bawah.

1. <stdio.h> berisi perintah:

**Printf**  
Merupakan fungsi keluaran.  
Bentuk umum: printf (“tulisan”);  
  
**Scanf**  
Merupakan fungsi masukan.  
Bentuk umum: scanf (“%c”, &karakter);  
 **Gets**  
Fungsi inputan yang bisa membaca spasi.  
Bentuk umum: gets(var x);

1. <string.h> berisi perintah:

**Strcpy**  
Digunakan untuk menyalin nilai string  
  
**Strlen**  
Untuk mengetahui panjang string  
  
**Strupr**  
Digunakan untuk membuat string menjadi capital.  
 **Strcmp**  
Digunakan untuk membandingkan dua buah string.  
Hasil dari fungsi ini bertipe integer dengan nilai: Negative, jika string pertama kurang dari string kedua. Nol, jika string pertama sama dengan string kedua Positif, jika string pertama lebih besar dari string kedua  
Bentuk umum : strcmp(string1, string2);  
  
**Strlwr**  
Digunakan untuk mengubah huruf menjadi kecil semua.  
Bentuk umum : strlwr(string);  
 **Strcat**  
Digunakan untuk menggabungkan string.  
Bentuk umum : strcat(string1,string2);

1. <math.h> berisi perintah:

**Sqrt**  
Fungsi akar  
  
**Pow**  
Fungsi pangkat  
  
**Sin (), Cos(), Tan()**  
Masing-masing digunakan untuk menghitung nilai sinus, cosinus dan tangens dari suatu sudut. Bentuk umum: Sin(sudut); Cos(sudut); Tan(sudut);  
 **Max**  
Digunakan untuk menghitung hasil pembagian dan sisa pembagian.  
Bentuk umum: max(bilangan1, bilangan2);  
 **Min**  
Digunakan untuk menentukan bilangan terkecil dari dua buah bilangan.  
Bentuk umum : min(bilangan1, bilangan2).

1. <windows.h> berisi perintah:

**System**  
Digunakan untuk memberi warna.

1. <iomanip.h> berisi perintah:

**Setiosflags()**  
Digunakan untuk mengatur jumlah digit decimal dibelakang koma.

1. <stdlib.h> berisi perintah:

**Atof()**  
Digunakan untuk mengonfersi nilai string menjadi bilangan bertipe double.  
Bentuk umum : atof(char x);  
  
**Atoi()**  
Digunakan untuk merubah tipe data string menjadi integer.  
 **Pow**  
Digunakan untuk pemangkatan suatu bilangan.Bentuk umum : pow(bilangan, pangkat).

1. <assert.h> Berisi menegaskan makro, digunakan untuk membantu mendeteksi kesalahan logis dan jenis lain dari bug dalam debugging versi dari sebuah program.
2. <complex.h> Sebuah set fungsi untuk memanipulasi bilangan kompleks.
3. <ctype.h> Mendefinisikan set fungsi yang digunakan untuk mengklasifikasikan karakter dengan jenis mereka atau untuk mengkonversi antara atas dan huruf kecil dengan cara yang independen dari yang digunakan set karakter (biasanya ASCII atau salah satu ekstensi, meskipun implementasi menggunakan EBCDIC juga dikenal).
4. <errno.h> Untuk menguji kode kesalahan dilaporkan oleh fungsi perpustakaan.
5. <fenv.h> Mendefinisikan sebuah set fungsi untuk mengendalikan floating-point lingkungan.
6. <float.h> Mendefinisikan konstanta makro menentukan implementasi khusus properti dari floating-point library.
7. <inttypes.h> Mendefinisikan tipe bilangan bulat lebar yang tepat.
8. <iso646.h> Mendefinisikan macro beberapa yang menerapkan cara-cara alternatif untuk mengekspresikan token beberapa standar. Untuk pemrograman di ISO 646 set varian karakter.
9. <limits.h> Mendefinisikan konstanta makro menentukan implementasi khusus properti dari tipe integer.
10. <locale.h> Mendefinisikan fungsi lokalisasi.
11. <setjmp.h> Demikianlah macro setjmp dan longjmp , yang digunakan untuk non-lokal keluar.
12. <signal.h> Mendefinisikan fungsi sinyal penanganan.
13. <stdalign.h> Untuk query dan menentukan keselarasan benda.
14. <stdarg.h> Untuk mengakses berbagai jumlah argumen dilewatkan ke fungsi.
15. <stdatomic.h> Untuk operasi atom pada data dibagi antara benang.
16. <stdbool.h> Mendefinisikan tipe data Boolean.
17. <stddef.h> Mendefinisikan jenis beberapa berguna dan macro.
18. <stdint.h> Mendefinisikan tipe bilangan bulat lebar yang tepat.
19. <stdnoreturn.h> Untuk menentukan non-kembali fungsi.
20. <tgmath.h> Mendefinisikan jenis generik-fungsi matematika.
21. <threads.h> Mendefinisikan fungsi untuk mengelola beberapa Threads serta mutexes dan variable kondisi.
22. <time.h> Mendefinisikan fungsi tanggal dan waktu penanganan.
23. <uchar.h> Jenis dan fungsi untuk memanipulasi Unicode karakter.
24. <wchar.h> Mendefinisikan fungsi penanganan string lebar.
25. <wctype.h> Mendefinisikan set fungsi yang digunakan untuk mengklasifikasikan karakter lebar jenis mereka atau untuk mengkonversi antara huruf besar dan kecil.