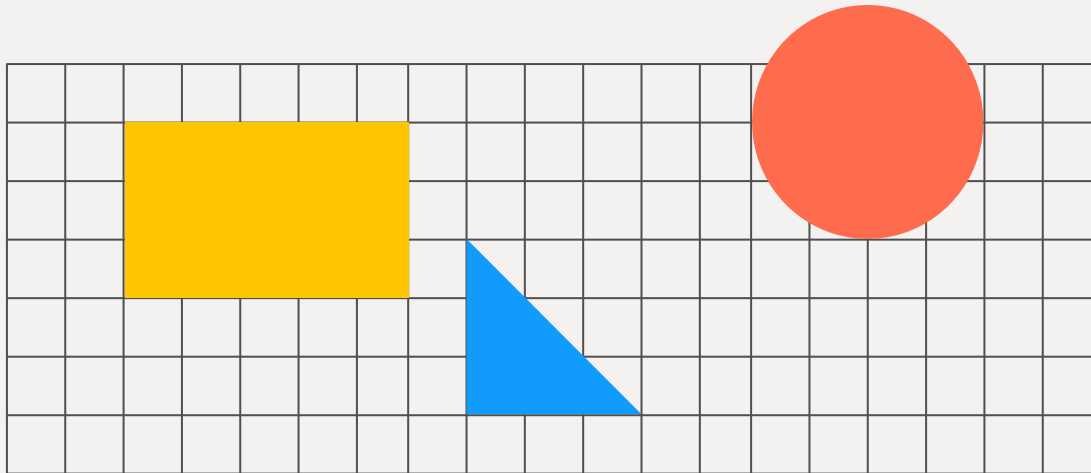
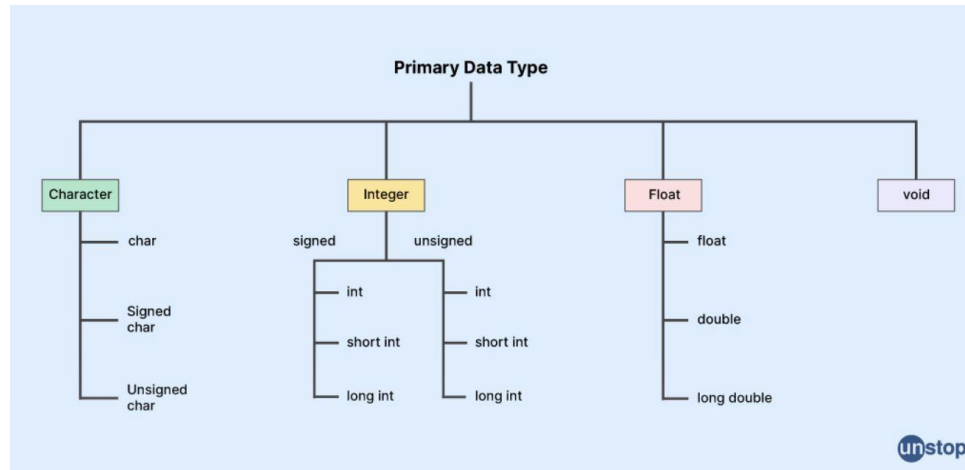


Persiapan Praktikum 4 STI



ADT in C

Data type in C?



How about we need a custom data type like point, time, etc. ?

ADT in C

```
1 typedef struct Time {
2     int jam;
3     int menit;
4     int detik;
5 } time;
```

Struct diatas merupakan struktur data baru untuk merepresentasikan waktu

```
1
2 void createTime(time *t, int jam, int menit, int detik) {
3     t->jam = jam;        // JAM(t) = jam;
4     t->menit = menit;    // MENIT(t) = menit;
5     t->detik = detik;    // DETIK(t) = detik;
6 }
```

Untuk menggunakan struktur data, diperlukan sebuah fungsi/metode untuk bisa menginisiasi sebuah struktur data dengan nilai (biasa disebut **Konstruktor**)

Selektor

```
1  /**
2   * ----- Selektor -----
3   * Getter & Setter
4   */
5  int getJam(time t) {
6      return t.jam;
7  }
8  void setJam(time *t, int jam){
9      t->jam = jam;
10 }
11 ...
```

Getter & Setter pada C

```
1  /**
2   * ----- Selektor -----
3   * Akses data dengan menggunakan macro
4   */
5  # define JAM(x) (x.jam)
6  # define MENIT(x) (x.menit)
7  # define DETIK(x) (x.detik)
```

Getter & Setter pada C
Menggunakan
macro/preprocessor

Selektor

```
1 // Contoh penggunaan
2 int main() {
3     time t;
4     setJam(&t, 10);
5     int jam = getJam(t);
6 }
```

Getter & Setter pada C

```
1 // Contoh penggunaan
2 int main() {
3     time t;
4     JAM(t) = 10;
5     int jam = JAM(t);
6 }
```

Getter & Setter pada C
Menggunakan
macro/preprocessor

Metode Lain

```
// Contoh metode lain-lain
void printTime(time t) {
    printf("%02d:%02d:%02d\n", t.jam, t.menit, t.detik);
    printf("%02d:%02d:%02d\n", getJam(t), getMenit(t), getDetik(t));
    printf("%02d:%02d:%02d\n", JAM(t), MENIT(t), DETIK(t));
}
```

Contoh metode lain yang
diperlukan dalam melakukan
pemrosesan ADT

Notes: 3 line diatas akan mengeluarkan keluaran yang sama

Contoh

Contoh kasus yang memerlukan ADT untuk merepresentasikan data:

- Antrian (List / Stack / Queue)
- Koordinat
- Waktu
- etc.

Setiap ADT harus dibuat menyesuaikan dengan kebutuhan/penggunaan struktur data tersebut. Misal:

ADT Waktu dibuat untuk memudahkan penyimpanan detik, menit, dan jam. Hal tersebut akan memudahkan penyimpanan daripada mengharuskan untuk mengkonversi jam dan menit ke dalam bentuk detik. Metode-metode yang diimplementasikan adalah metode metode yang diperlukan untuk mempermudah penggunaan ADT Waktu

OOT

Penggunaan Header yang baik dan benar:

- Hanya melakukan implementasi fungsi-fungsi pada file header (file yg ext nya .h)
- Include file header nya, jangan include library lain (ex: `#include "Header.h"`)
- Jangan merubah file headernya yaa! Implementasi aja apa yang sudah ada di header

Penggunaan Compiler GCC yang baik dan benar:

```
gcc -o <nama_executable> <nama_file .c yang diperlukan>
```

- Keterangan
`nama_executable` : bebas aja yaa, yang penting kalian inget
`Nama_file .c yang diperlukan` : semua file berekstensi .c yang akan kalian gunakan
- Misal:
kalian punya file **driver.c** dan **NimonsGanteng.c** maka command yang diperlukan untuk meng-compile code menjadi executable bernama **main** adalah

```
gcc -o main driver.c NimonsGanteng.c
```


OOT



Cara Run:

- Setelah kalian meng-compile code kalian, kalian dapat melihat ada file executable baru (dalam kasus contoh dislide sebelumnya, kalian akan melihat ada file baru bernama **main**)
- Jalankan executable tersebut dengan cara
`./main`

Notes: Semua langkah compile & Run dilakukan di terminal yaa!

Thank You

HABIS MUDIK DARI KLATEN
NIH OLEH OLEH DARI ASISTEN



Semoga bermanfaat ya!

Kalau semisal ada pertanyaan tambahan boleh banget tanya ke asisten yaaa!

Valen - Maulvi - Roihan - Jendra

