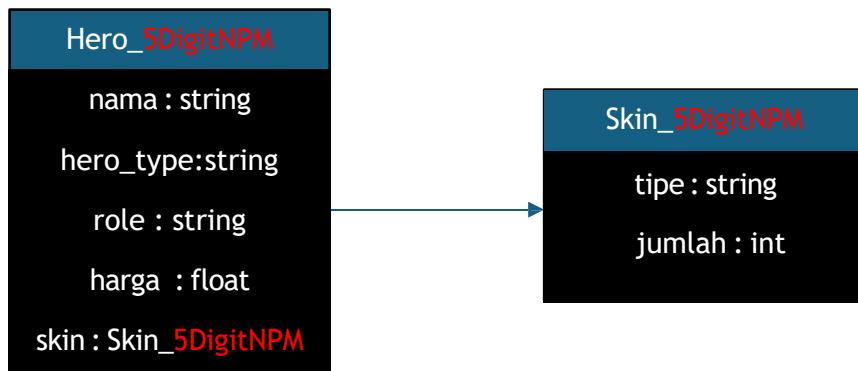


UNGUIDED
MODUL 11 (RECORD)
Tipe D



Perusahaan MOONTON meminta Anda untuk membuat game moba, mereka ingin mendata para hero yang sudah mereka design. Sebagai seorang programer handal anda diminta untuk membuat program yang dapat digunakan oleh Perusahaan MOONTON untuk dapat melihat siapa saja hero tersebut dan skin yang pernah mereka dapatkan, berdasarkan diagram dibawah ini.



Sebelum masuk ke dalam menu utama, pengguna akan diminta untuk login terlebih dahulu. Untuk usernamenya adalah **NamaPanggilan**, dan passwordnya adalah **5DigitNPM**. (Pada demo username : **admin**, password : **admin**)

Ketentuan Login :

- Username dan Password wajib bersifat case sensitive
- Jika username atau password salah maka akan menampilkan error handling

Berikut ini tampilan dari Login Page:

Jika berhasil Login

```
==== LOGIN [MOONTON] ====  
  
Masukkan Username      : NamaPanggilan  
Masukkan Password       : 5DigitNPM  
  
Hallo NamaPanggilan, Anda Berhasil Login!!!|
```

Jika gagal Login

```
==== LOGIN [MOONTON] ====  
  
Masukkan Username      : a  
Masukkan Password       : a  
  
Username atau Password Anda Salah!|
```

1. Menu 1 : Input Data Hero

Menu ini hanya dapat diakses apabila data hero belum terisi semua.

Program akan meminta inputan berupa :

- Nama Hero
- Tipe Hero (Fighter | Marksman | Mage | Support | Tanker | Assasin)
- Role Hero (Roamer | Gold | Exp | Midlane | Jungler)
- Harga Hero (Random Otomatis dari 24000-32000)
- Tipe Skin (Elite | Epic | Legend)
- Jumlah Skin

Ketika pengguna menginputkan selain pilihan diatas maka program akan meminta inputan ulang.

Ketentuan inputan pengguna :

1. Nama hero tidak boleh kosong atau “-“,
2. Tipe hero harus sesuai,
3. Role hero harus sesuai,
4. Tipe skin harus sesuai,
5. Jumlah skin tidak boleh sama dengan “0” dan negatif.

Notes:

Hanya bisa input 2 data saja.

Buatlah error handling jika kedua data hero sudah diisi semua.

2. Menu 2 : Show Data Hero

Menu ini hanya dapat diakses ketika minimal ada 1 data hero yang sudah terisi.

Pada menu ini anda diminta untuk menampilkan semua data yang sudah diinputkan pada menu sebelumnya.

Contoh tampilannya :

```
--- Tampil Data Hero ---  
    [*] Data Hero  
        Nama Hero      : Fanny  
        Type Hero       : Assassin  
        Role Hero       : Jungler  
        Harga Hero     : 28539.00  
  
--- Tampil Skin Hero ---  
        Tipe Skin      : Epic  
        Jumlah Skin     : 10|
```

Note:

Format tampil data harus sama persis seperti pada contoh diatas.

Buatlah error handling jika kedua data hero belum diisi semua.

3. Menu 3 : Reset Data Hero

Menu ini hanya dapat diakses ketika minimal ada 1 data hero yang sudah terisi.

Pada menu ini program akan masuk ke dalam submenu reset data hero, yang didalamnya terdapat 3 buah menu, yaitu:

a. Reset semua data hero

Jika pengguna memilih menu ini maka program akan melakukan reset kepada semua data yang sudah diinputkan.

b. Reset salah satu data hero

Jika pengguna memilih menu ini maka program akan meminta inputan nama hero yang ingin direset. Ketika nama hero yang dicari tidak ditemukan maka akan menampilkan error handling. Namun jika nama yang dicari ada pada data hero, maka program akan langsung mereset data hero tersebut.

Note:

Jika pengguna sudah menginputkan 2 data hero. Ketika pada menu reset data hero, pengguna memilih untuk mereset hero pertama. Maka hero kedua akan otomatis berpindah menjadi hero pertama.

c. Kembali ke main menu

Jika pengguna memilih menu ini maka program akan kembali ke menu awal.

Buatlah error handling jika kedua data hero belum diisi semua.

4. Menu 0 : Exit

Pada menu ini program akan menampilkan nama, npm, dan kelas dari praktikan dengan format berikut ini:

“<Nama Praktikan> | <NPM> | <Kelas>

Contoh : “Alexander Wongso | 230712519 | A”

BONUS 1

Setelah login, menambahkan **loading** sebelum masuk ke menu utama. (untuk lebih jelas, silahkan cek demo)

BONUS 2

1. Menambahkan satu menu lagi yaitu update data.

Menu ini hanya dapat diakses ketika minimal ada 1 data hero yang sudah terisi.

Pada menu ini program akan meminta inputan berupa nama hero yang datanya ingin diupdate. Jika nama hero tidak ditemukan maka akan menampilkan error handling dan akan kembali ke menu utama lagi. Namun jika nama yang dicari ketemu maka akan masuk ke dalam submenu update data hero, yang didalamnya terdapat 4 buah menu, yaitu:

a. Elite

Jika pengguna memilih menu Elite maka program akan meminta inputan berupa jumlah skin yang ingin di update, error handling pada inputan sama seperti pada menu 1.

b. Epic

Jika pengguna memilih menu Epic maka program akan meminta inputan berupa jumlah skin yang ingin di update, error handling pada inputan sama seperti pada menu 1.

c. Legend

Jika pengguna memilih menu Legend maka program akan meminta inputan berupa jumlah skin yang ingin di update, error handling pada inputan sama seperti pada menu 1.

Notes:

Apabila pada menu 1 pengguna telah menginputkan suatu tipe skin, lalu pada menu update data yg dipilih adalah tipe skin yang sama seperti pada saat inputan di menu 1, maka program akan menampilkan error handling bahwa tidak dapat merubah ke tipe skin tersebut. Contoh : pada inputan di menu 1 pengguna menginputkan tipe skin Elite, maka pada menu update data hero pengguna tidak dapat memilih menu Elite.

d. Kembali ke main menu

Jika pengguna memilih menu ini maka program akan kembali ke menu awal.

Buatlah error handling jika kedua data hero belum diisi semua.

BONUS 1:

Menyelesaikan **UGD + BONUS1** dengan benar selama praktikum berlangsung (nilai **UGD 105**)

BONUS 2:

Menyelesaikan **UGD + BONUS2** dengan benar selama praktikum berlangsung (nilai **UGD 115 + nilai TUGAS 100**)

BONUS 3:

Menyelesaikan **UGD + BONUS1 + BONUS2** dengan benar selama praktikum berlangsung (nilai **UGD 120 + nilai TUGAS 100**)

TUGAS



Karena Anda jenuh bekerja, Anda ingin bermain game pokemon. Tetapi untuk memenuhi kebutuhan Anda, Anda membuat program pokemon sendiri. Dimana program tersebut merupakan simulasi pertarungan antara Bulbasaur dan Squirtle, dengan ketentuan:

- Health Bulbasaur : **1000**, Squirtle : **1500**
- Attack Bulbasaur : **75 – 130** , Squirtle : **50 – 80**
- Setiap Attack memungkinkan mendapat critical hit dengan chance **50%**
(critical hit = damage attack * 2)
- Pokemon yang menghabiskan Health lawannya duluan menang.
- Jika satu pokemon menyerah, maka pokemon lainnya menang.



BULBASAUR



SQUIRTLE

Harus menggunakan setidaknya 1 struct dan tidak boleh menggunakan materi Array
Hint :

```
typedef struct
{
    string name;
    int hp;
    int att_min;
    int att_max;
}Pokemon;
```

1. Menu Tugas

```
----- POKEMON BATTLE -----
-----
[Bulbasaur] vs [Squirtle]
-----
HP : 1000          HP : 1500
Att : 75-130       Att : 50-80

Bulbasaur's Turn !
[1] Normal Attack
[2] Surrender
>>> |
```

2. Serangan Pokemon bisa berupa normal attack, bisa berupa critical hit

```
----- POKEMON BATTLE -----
-----
[Bulbasaur] vs [Squirtle]
-----
HP : 1000          HP : 1500
Att : 75-130       Att : 50-80

Bulbasaur's Turn !
[1] Normal Attack
[2] Surrender
>>> 1

[*] 91 Damage has been done!

----- POKEMON BATTLE -----
-----
[Bulbasaur] vs [Squirtle]
-----
HP : 941           HP : 1409
Att : 75-130        Att : 50-80

Bulbasaur's Turn !
[1] Normal Attack
[2] Surrender
>>> 1

[*] CRITICAL HIT !
[*] 176 Damage has been done!
```

3. Ketika salah satu pokemon kehabisan HP

```
----- POKEMON BATTLE -----
-----
[Bulbasaur] vs [Squirtle]
-----
HP : 132           HP : 0
Att : 75-130        Att : 50-80

[!] Squirtle fainted [!]
[*] Bulbasaur won with 132 HP left|
```

4. Ketika salah satu pokemon menyerah

```
----- POKEMON BATTLE -----  
-----|Bulbasaur|     vs     |Squirtle|-----  
HP  : 1000          HP  : 1500  
Att : 75-130        Att : 50-80  
  
Bulbasaur's Turn !  
[1] Normal Attack  
[2] Surrender  
>>> 2  
  
[*] Bulbasaur surrendered, Squirtle won with 1500 HP left|
```

Hint (Tidak Wajib Untuk Digunakan) :

```
● ● ●  
1 //UGD  
2 void showMenu();  
3 void menuReset();  
4 void menuUbah();  
5 void initData(Hero_12519 *H);  
6 bool cekEmpty(Hero_12519 H1, Hero_12519 H2);  
7 bool cekFull(Hero_12519 H1, Hero_12519 H2);  
8 void tampilData(Hero_12519 H);  
9 void hapusData(Hero_12519 *H);  
10 void inputData(Hero_12519 *H, string nama, string hero_type, string role, float harga, string tipe, int jumlah);  
11 //BONUS  
12 void ubahSkin(Hero_12519 *H, string tipe);  
13 //TUGAS  
14 void initPokemon(Pokemon *p, string name, int hp, int min, int max);  
15 void printBattleStatus(Pokemon p1, Pokemon p2);  
16 int generateAttack(int min, int max);  
17 void showAttack(Pokemon p);
```

FORMAT PENAMAAN FILE :

- UGD : **UGD11_X_YYYYYY.zip**
- BONUS 1 : **UGD11_X_YYYYYY_Bonus1.zip**
- BONUS 2 : **UGD11_X_YYYYYY_Bonus2.zip**
- BONUS 3 : **UGD11_X_YYYYYY_Bonus3.zip**
- TUGAS : **TGS11_X_YYYYYY.zip**

Keterangan :

X : Kelas

Y : 5 Digit NPM Terakhir

Peraturan :

- Tambahkan 5 digit NPM terakhir di belakang setiap struct. Contoh : Dosen_12519

```
1 typedef struct{  
2     string nama;  
3     string mataKuliah;  
4     float gaji;  
5 }Dosen_12519;
```

(-10).

- Project harus memiliki struktur record/struct sesuai dengan diagram kelas yang disertakan (di halaman pertama). Nama field di dalam record tersebut tidak harus sama (-20).
- Kesalahan format pengumpulan (-10).
- File code harus ekstensi .c bukan .cpp (-20).
- Bonus hanya bisa diambil jika sudah mengerjakan UGD dengan benar.
- Praktikan hanya memiliki 3x kesempatan untuk pengecekan bonus.
- File code harus ekstensi .c bukan .cpp (-20).
- Perhatikan tempat uploader sebelum mengupload pekerjaan kalian (salah uploader akan dianggap tidak mengumpulkan).
- Tidak diperbolehkan menggunakan materi modul selanjutnya (-20).
- Tindak kecurangan dan plagiasi baik ke sesama praktikan atau pun AI tidak akan diberikan toleransi.
- Peraturan lain pada spreadsheet penilaian juga berlaku.

