

PRAKTIKUM ARRAY OF RECORD TIPE D

Negara neverland ingin melaksanakan pemilu, dan anda sebagai programmer handal diminta untuk membuat sistem yang dapat melakukan voting tersebut.

Requirement yang dibutuhkan negara neverland sebagai berikut :

Kandidat	Voter
string nama	string nama
int umur	string NIK
string partai	string asal
Voter voters[MAX_VOTER]	

MAX_VOTER define 10

MAX_KANDIDAT define 10

Fitur pertama (CREATE):

Partai dapat mengajukan kandidat untuk maju pemilu.

```
>>> 1
Masukan Nama Partai Pengusung : PTI
Masukan Nama Kandidat : Andi
Masukan Umur Kandidat : 40

[+] Partai PTI telah mengusung Andi [+]
```

Tetapi ada ketentuan bahwa satu partai hanya boleh mengajukan satu kandidat.

```
>>> 1
Masukan Nama Partai Pengusung : PTI

[!] 1 partai hanya boleh mengusung 1 kandidat [!]
Masukan Nama Partai Pengusung : |
```

Fitur kedua (DELETE):

Sistem dapat melakukan diskualifikasi kandidat dan menghapus data-nya (apabila sudah memiliki suara maka data voters-nya pun dihapus).

```
>>> 2
Masukan Nama Kandidat : Andi

[+] Berhasil Mendiskualifikasi Kandidat [+]
```

```
Masukan Nama Kandidat : Andi

[!] Kandidat Tidak Ditemukan [!]
```

Fitur ketiga :

Menghitung jumlah suara dari kandidat yang dicari. Sistem akan meminta inputan nama kandidat dan akan menampilkan jumlah suara yang diperoleh dari kandidat tersebut apabila kandidat tersebut ada dalam data kandidat.

```
Masukan Nama Kandidat : Andi  
|  
Kandidat Andi sudah memiliki 0 suara
```

Fitur keempat :

Lakukan voting suara. Program akan meminta nama kandidat yang akan di vote. Apabila kandidat tidak ditemukan tampilkan error handling sesuai.

```
Masukan Nama Kandidat : Budi  
[!] Kandidat Tidak Ditemukan [!]
```

Apabila ditemukan maka dapat melakukan vote apabila data voters kandidat tersebut masih ada.

```
Masukan Nama Kandidat : Andi  
Masukan Nama Voter: Budi  
Masukan NIK Kandidat : 123  
Masukan Asal Voter: Jakarta  
|  
[+] Berhasil Vote [!]
```

Untuk inputan NIK harus unik, sehingga tidak boleh ada NIK duplikasi di seluruh data array.

```
Masukan Nama Kandidat : Charlie
Masukan Nama Voter: Tani
Masukan NIK Kandidat    : 123

      [!] NIK ini sudah digunakan [!]
Masukan NIK Kandidat    : |
```

Karena NIK 123 sudah ada pada voter Budi.

Fitur kelima :

Quick count

Pada fitur ini program akan menampilkan kandidat dan jumlah suara diperoleh., lalu tampilkan persentase suara tersebut.

```
          Kandidat Nomor Urut -1

Nama    : Agus
Partai  : PTI
Jumlah Suara : 1
50.00 persen dari populasi suara

          Kandidat Nomor Urut -2

Nama    : Charlie
Partai  : PBB
Jumlah Suara : 1
50.00 persen dari populasi suara
```

Karena sudah ada 2 voter dan masing-masing kandidat memiliki 1 voter maka sama-sama 50 persen.

```
Masukan Nama Kandidat : Agus
Masukan Nama Voter: Joni
Masukan NIK Kandidat    : 789
Masukan Asal Voter: Jogja
```

```
[+] Berhasil Vote [!]
```

|
Apabila melakukan vote lagi untuk kandidat Agus

```
Kandidat Nomor Urut -1
Nama    : Agus
Partai  : PTI
Jumlah Suara : 2
66.67 persen dari populasi suara

Kandidat Nomor Urut -2
Nama    : Charlie
Partai  : PBB
Jumlah Suara : 1
33.33 persen dari populasi suara
```

BONUS (UGD 120)

Tampilkan pemenang.

Pada fitur ini program akan menampilkan siapa yang memenangkan pemilu. Pemilu didapat dengan cara mencari kandidat dengan perolehan suara terbanyak.

```
>>> 6
Pemenangnya adalah: Agus (2 suara)
|
```

Contoh kasus tadi.

Apabila terjadi seri (dimana dua atau lebih kandidat memiliki suara yang sama, maka pemenang adalah kandidat dengan asal voter paling beragam).

Contoh :

Agus memiliki 2 Voter dan keduanya berasal dari kota X.

Budi juga memiliki 2 Voter yang satu berasal dari Kota X dan satunya lagi dari Kota Y.

Karena voter Budi berasal dari 2 kota berbeda maka Budi-lah yang memenangkan pemilihan.

```
        Kandidat Nomor Urut -1
Nama      : Agus
Partai    : PPP
Jumlah Suara : 2
50.00 persen dari populasi suara

        Kandidat Nomor Urut -2
Nama      : Budi
Partai    : PBB
Jumlah Suara : 2
50.00 persen dari populasi suara
|
```

```
>>> 6
Pemenangnya adalah: Budi (2 suara, 2 asal unik)
|
```

TUGAS (SELESAIKAN UGD + Sort Kandidat)

```
>>> 7

        Kandidat Nomor Urut [1]
Nama    : Charlie
Partai  : p3k

        Kandidat Nomor Urut [2]
Nama    : Budi
Partai  : PBB

        Kandidat Nomor Urut [3]
Nama    : Andi
Partai  : PPP
|
```

Lakukan sorting kandidat berdasarkan alfabet dari terbesar ke terkecil dan tampilkan.

KETENTUAN PENGUMPULAN UGD DAN TUGAS

1. File main berekstensi C jangan C++.
2. File di zip dengan format penamaan UGD12_X_YYYYY.zip

X = Kelas

Y = 5 digit terakhir NPM

Untuk bonus UGD12_X_YYYYY_BONUS.zip

3. Untuk tugas format penamaan TGS12_X_YYYYY.zip
4. Jangan sampai salah format. Kesalahan format penamaan file -10.
5. Ini sudah modul terakhir, tidak ada lagi toleransi untuk plagiasi.