

## Tutorial 11 DDP 1 - Kelas F For Respect

### GUI + Class

**Deadline :** Rabu, 28 November 2018 pukul 17:40 Waktu Scele

### Berlayar Menuju Asa, Terbang Menuju Angkasa

Familiar dengan kalimat di atas, ya, selamat datang Quanta menjadi bagian dari keluarga baru CSUI, semua angkatan sangat bahagia menyambut kalian, salah satunya Rey, bagian dari Tarung 2017. Pada suatu hari yang cerah, Rey mendapatkan suatu pesan rahasia yang dikirimkan dari petinggi yang lain yang bernama Dipsi, pesan tersebut tertulis dalam bahasa Jepang:

レイさん

今日は 忙しい だから。私は今 ロンドン へ 行って、文化祭 がある から。私が あそこ いる間に 銀行の を セイシステム を つくりませんか。

ディプシ

Rey diharuskan membuat program berbasis GUI yang dapat mensimulasikan bagaimana bank bekerja, kebetulan Quanta memiliki sebuah project akhir tahun untuk membuat sebuah sistem bank online, tetapi tidak menggunakan terminal, karena terminal tidak begitu friendly untuk kebanyakan orang, melainkan harus berbasis GUI

Waktu maksimal produk tersebut harus jadi tertulis dalam bahasa pemrograman C++ . **Jika penasaran**, silakan jalankan pesan tersebut di <https://www.jdoodle.com/online-compiler-c++> dengan memilih Version GCC 8.1.0 langsung tekan "Execute".

```
#include <bits/stdc++.h>
using namespace std;
int main() {
    int sum = 0;
    string str = "dipsi-lala-po";
    for (unsigned int i = 0; i < str.size(); i++) {
        sum += str[i];
    }
    cout << "Kerjakan dalam waktu maksimal: " << sum - 1160 << " menit!" << endl;
    return 0;
}
```

**Program ini akan berbasis GUI**, dan data para nasabah akan diberikan dalam bentuk file **data\_nasabah.in** yang kemudian akan diproses oleh program sebelum program dapat digunakan oleh pengguna. Hasil transaksi, cukup ditampilkan saja. Secara umum, akan ada 4 operasi yang memungkinkan. Setiap operasi merupakan representasi dari 1 tombol pada program berbasis GUI yang akan dibuat

### 1. Deposit (Memasukkan dana)

Saat perintah ini dijalankan, jika total uang setelah di setor melebihi limit, SETOR tetap berhasil, tetapi saldo yang akan masuk ke pengguna hanya sebagian (hingga limit).

**Contoh:** saldo sekarang 175.000, limit saldo 200.000, deposit 100.000, maka yang masuk hanya akan 25.000, **sedangkan 75.000 sisanya menjadi keuntungan bank.** YA, Bank juga menginginkan keuntungan :)

Jika berhasil, cetak dengan:

**Transaksi Berhasil! Uang Masuk [ Uang Deposit ]**

**Dalam kasus ini, berarti akan menampilkan**

**Transaksi Berhasil! Uang Masuk 25000**

### 2. Tarik (Menarik uang dari saldo yang ada)

Saat dijalankan, jika jumlah uang yang di tarik > saldo, cetak:

**Transaksi gagal! Saldo tidak cukup!**

dan tidak ada perubahan pada saldo, jika berhasil, cetak:

**Berhasil menarik sebesar [Jumlah Uang] !**

### 3. Transfer (Melakukan transfer dari satu pengguna ke pengguna lain)

Saat dijalankan, jika uang yang ditransfer < saldo pengirim, cetak:

**Transaksi gagal! Saldo tidak cukup!**

Apabila memungkinkan untuk ditransfer,

**TETAPI** saldo penerima + uang ditransfer melebihi limit akun penerima, maka transfer TETAP BERHASIL, uang pengirim berkurang, uang penerima bertambah hingga mencapai limit, dan **sisanya yang tidak masuk akan menjadi keuntungan bank.**

Contoh:

Saldo pengirim : 100.000

Saldo penerima : 150.000 , dan limit saldo penerima : 200.000

**Kasus: Pengirim hendak transfer 75.000, transaksi pun sukses.**

Saldo pengirim berkurang 75.000

Saldo penerima hanya bertambah 50.000

Sisa 25.000 menjadi keuntungan bank

Hal ini terjadi karena penerima menerima saldo yang melebihi limit yang diizinkan.

Cetak:

**Berhasil transfer sebesar [Jumlah Uang Bersih yang Diterima Penerima] !**

Jika uang yang di transfer  $\leq$  saldo pengirim DAN saldo penerima + uang diterima  $\leq$  limit penerima, cetak:

**Berhasil transfer sebesar [Nominal yang ditransfer] !**

#### 4. Hutang (Melakukan peminjaman kepada pengguna lain)

Saat dijalankan, maka akan menampilkan :

**Uang yang harus dikembalikan [ Jumlah uang + bunga \* jumlah uang ] !**

Bunga yang harus dikembalikan berbeda bergantung pada jenis paket.

- a. Weaboo (Pecinta Budaya) : 20%
- b. Riajuu (Remaja Gaul) : 15%
- c. Normies (Orang Terkenal) : 10%

Contoh Kasus:

Gagah dengan akun Normies meminjam uang kepada Faiz sejumlah 100000, dengan begitu Gagah harus mengembalikan sebesar **(100000 + 10% \* 100000)**, dan saldo Faiz berkurang sebesar 100000. Saldo Gagah tidak bertambah, karena harus diverifikasi bank

#### 5. Revenue (Cetak keuntungan bank)

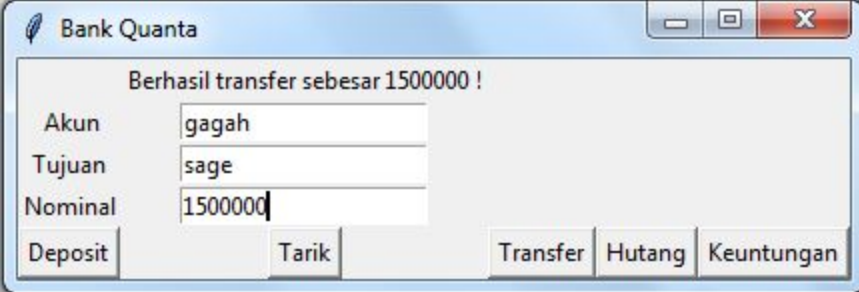
Cukup tampilkan **Keuntungan Bank adalah sebesar [ Jumlah uang ]**

#### Validasi:

1. Saldo hanya boleh **0  $\leq$  saldo  $\leq$  limit**. Jika saldo negatif, ubahlah menjadi nol.
2. Setiap transaksi dijamin selalu bilangan positif dan **TIDAK ADA biaya transaksi**.
3. Setiap nasabah dijamin memiliki nama yang unik. Operasi hanya dapat dilakukan apabila nasabah terdaftar dalam sistem. Jika tidak terdaftar, cukup tampilan **"Nasabah tidak terdaftar"**
  - Weaboo (Pecinta Budaya) : 50000000
  - Riajuu (Remaja Gaul) : 30000000
  - Normies (Orang Terkenal) : 10000000
4. Akan disediakan template untuk GUI + Class nya tersendiri, tetapi masih belum optimal, silakan optimalkan dengan cara melengkapi program.



4. Saat operasi transfer dijalankan



Bank Quanta

Berhasil transfer sebesar 1500000 !

Akun: gagah

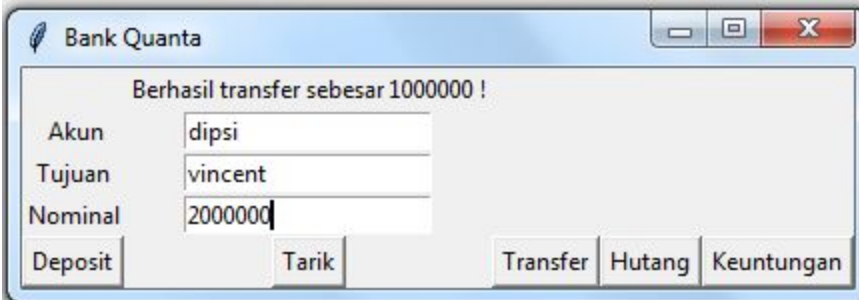
Tujuan: sage

Nominal: 1500000

Deposit Tarik Transfer Hutang Keuntungan

Operasi

transfer untuk kasus khusus, saldo yang masuk hanya 1000000 dikarenakan Vincent memiliki tabungan normies dengan limit 10 juta, dan saldo awal 9,9 juta. Sehingga dia hanya dapat menerima 1 juta agar mencapai limit



Bank Quanta

Berhasil transfer sebesar 1000000 !

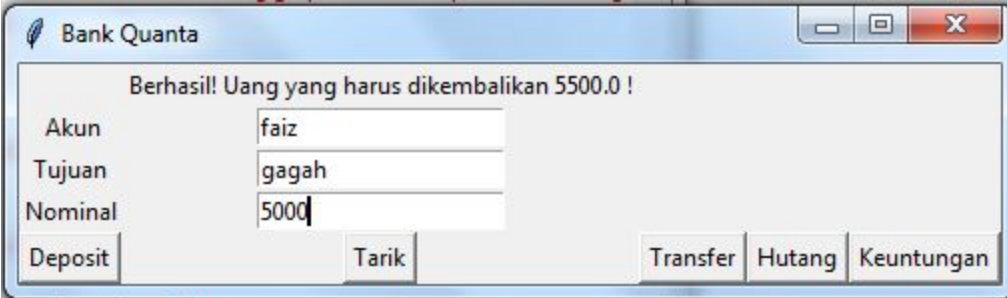
Akun: dipsi

Tujuan: vincent

Nominal: 2000000

Deposit Tarik Transfer Hutang Keuntungan

5. Saat operasi Hutang dilakukan



Bank Quanta

Berhasil! Uang yang harus dikembalikan 5500.0 !

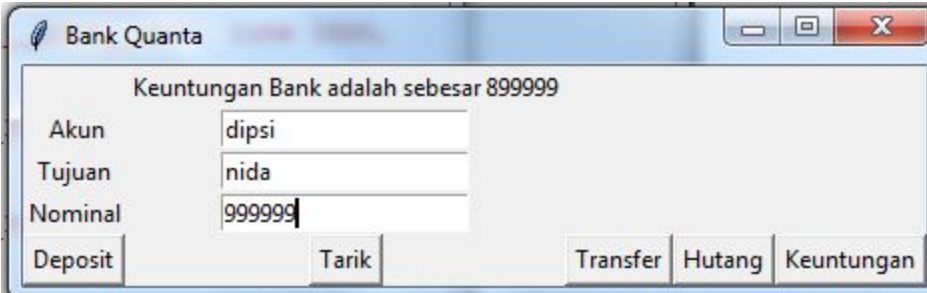
Akun: faiz

Tujuan: gagah

Nominal: 5000

Deposit Tarik Transfer Hutang Keuntungan

6. Saat operasi keuntungan dijalankan



Bank Quanta

Keuntungan Bank adalah sebesar 899999

Akun: dipsi

Tujuan: nida

Nominal: 999999

Deposit Tarik Transfer Hutang Keuntungan

### **Komponen Penilaian**

- 40% : *Output* sesuai yang diharapkan
- 30% : Mengimplementasikan seluruh hal yang dibutuhkan program dengan baik
- 20% : Penggunaan tipe data yang tepat
- 10% : Kerapihan kode, dokumentasi & penamaan file

### **Format Pengumpulan**

- [Kode Asdos]\_[Nama Lengkap]\_[NPM]\_Lab11.py  
Tidak perlu menggunakan tanda [ ]

#### **Contoh:**

GPR\_Rd Pradipta Gitaya S\_1706039566\_Lab11.py

**Selamat Mengoding ^^**