

# Struktur Data

Dosen : Drs.Wahyu Pujiyono, M. Kom.



Disusun Oleh :

- 1.Refaldi Ergian (1700018013), Status : Aktif**
- 2.Aji Gelar Prayogo (1700018016), Status : Aktif**
- 3.Bayu Krisna Murti (1700018015), Status : Aktif**

Universitas Ahmad Dahlan  
Fakultas TI Program Studi Teknik Informatika  
Kelas : A  
2018

# KATA PENGANTAR

Puji syukur kehadirat Allah Yang Maha Esa atas limpahan rahmat dan hidayah-Nya kepada penulis, sehingga penulis dapat menyelesaikan laporan ini guna memenuhi Tugas Proyek Struktur Data.

Laporan ini memiliki pembahasan tentang “Sistem Penjualan/Penyetoran Beras”.

Sholawat serta salam tetap tercurahkan kepada junjungan besar kita, Nabi Muhammad S.A.W, Keluarga, dan Sahabat. Semoga kita sebagai umatnya, bisa mendapatkan syafaat beliau di akherat kelak.

Dalam penyusunan laporan ini, tentunya penulis mendapatkan pengetahuan dan pengalaman dari beberapa pihak. Oleh karena itu, penulis mengucapkan terimakasih kepada :

1. Drs. Wahyu Pujiono, M.Pd, selaku dosen pengampu mata kuliah Struktur Data, Prodi Teknik Informatika, Fakultas Teknologi Industri, Universitas Ahmad Dahlan Yogyakarta
2. Teman-teman seperjuangan dalam mata kuliah Struktur data, Prodi Teknik Informatika, Fakultas Teknologi Industri, Universitas Ahmad Dahlan Yogyakarta.
3. Klien yang telah bersedia mendengar, mengamati, mencoba, dan meminati mengenai presentasi program yang telah kami buat.
4. Serta seluruh pihak yang tidak mungkin penulis sebutkan satu persatu yang telah terlibat banyak membantu, sehingga tugas proyek dan laporan ini dapat diselesaikan.

Penulis menyadari bahwa laporan dan tugas proyek ini masih jauh dari sempurna. Oleh karena itu, kritik dan saran yang membangun selalu penulis harapkan demi penyusunan laporan yang lebih baik kedepannya. Penulis berharap, semoga laporan ini dapat bermanfaat untuk penulis khususnya, dan pembaca pada umumnya.

## DAFTAR ISI

- **BAB 1 (Hasil Observasi).....Halaman 4**
  - A) Gambaran Umum.....Halaman 4
    - Permasalahan.....Halaman 4
    - Data Observasi.....Halaman 4
    - Foto Pendukung.....Halaman 5
  - B) Rumusan Masalah.....Halaman 6
    - Analisis Data.....Halaman 7
    - Target Proyek.....Halaman 7
- **BAB 2 (Pembahasan).....Halaman 8**
  - A) Simulasi Proses Bisnis.....Halaman 8
  - B) Program dan penjelasannya.....Halaman 8-54
  - C) Uji Coba Dan Dokumentasi Program.....Halaman 55-60
- **BAB 3 (Kesimpulan).....Halaman 60**
  - 1. Kesimpulan Hasil.....Halaman 60
  - 2. Kendala-Kendala Yang Ada.....Halaman 60-61
- **BAB 4 (Penutup).....Halaman 61**

# BAB 1

## Hasil Observaasi

### A.Gambaran Umum

Disini program ditunjukkan kepada pemilik beras. Tema yang digunakan adalah “Sistem Penjualan/Penyetoran Beras” , Artinya dimana jalan/prosesnya penjualan atau penyetoran beras. Proses bisnisnya singkat, yaitu Distributor → Toko → Masyarakat. Didalam Toko terdapa tumpukan lagi dan lagi yang disalurkan kepada distributor tadi.

Di toko terdiri dari banyak stack dan beras, bagi pemilik beras, perhitungan masih menggunakan manual dan kalkulator begitu juga dengan nota kontannya.

- Permasalahan :

Di toko, sering kali banyak terjadi seperti :

Banyaknya pelanggan yang datang sehingga pegawai/pemilik toko kewalahan sedangkan untuk note/pencatatan barang masih dilakukan manual menggunakan kalkulator dan note catat, Begitu juga dengan pencatatan distribusi yang masih manual sehingga masih berbekalkan catatan dan pengingat semata sehingga waktu kurang efektif.

- Data Observasi :

Beberapa hasil observasi yang kami dapatkan, diantaranya:

- Beberapa Jenis Beras dan merek :
  - -Beras C4 (harga : 120000 ; dengan kualitas terbaik)
  - -Beras Delanggu (harga 115000 : dengan kualitas terbaik)
  - -Beras Merah (harga 160000 : yang merupakan beras spesial)
- Banyak barang dan tumpukan yang ada (Beberapa dicampur berasnya dan beberapa tidak).
- Adanya gudang penyimpanan didalam ruang toko.
- Ada catatan distributor serta log waktu apa saja pengiriman,penyetoran maupun pembeliannya.

- Foto Foto Pendukung :



## B.Rumusan Masalah

- Analisis Data :

---

### Sistem Penjualan/Penyetoran Beras

Sasaran Kegiatan :

Mengetahui bagaimana cara / proses bisnis dari sistem penjualan beras, misalnya menyetor dengan memasukkannya ke toko dalam stack 1 (tumpukan 1) dengan memilih jenis setiap beras dalam grade atau merek/brand dan penambahan lainnya yang memungkinkan, misalkan update atau mengambil beras, mengganti karungnya, dll.

Gambaran Proses Bisnis :

- Beras pada awalnya ada pada distributor beras , didalamnya ada berbagai macam jenis beras serta grade dari beberapa perusahaan.
- Untuk mengambil beras yang ada pada distributor, sebagai syarat pengambilan maka kita tambahkan saldo didalamnya , dan harga beras juga di tingkatkan berdasarkan brand/grade.
- Kita sebagai pemilik toko membeli dan menyetor beras ke dalam toko dengan jumlah stack yang di tentukan, misalnya 10 tingkat ke atas per stacknya untuk di toko. Misalnya di toko hanya ada / terdapat 10 stack dengan 1 stack sebanyak 10 tingkatan ke atas. Apabila berlebih maka stack dari beras yg kelebihan, otomatis akan dipindahkan ke gudang sebagai penyimpanan stock (stock tambahan).
- Sekedar informasi, beras yang baru di beli(restock), maka posisi berada pada paling atas dari sebuah tumpukan(stack) sesuai jenis beras masing masing serta beras juga bisa dihitung jumlah tumpukannya sebagai catatan informasi.
- Jika kantung beras yang ditemukan di antara tumpukan beras sudah bocor/rusak, maka beras akan di ambil dari yang paling atas sampai ke yang bocor untuk di ganti, sedangkan kantung beras yang lain akan di pindahkan ke stack sementara di toko.
- Beras yang sudah di tambahkan di toko tadi, diberi harga yg sesuai untuk diperjualkan ke masyarakat.

TIDAK SELURUH DATA DIBAWAH INI AKAN DI IMPLEMENTASIKAN KESELURUHANNYA

DATA DISTRIBUTOR :

<b><u>BERAS DISTRIBUTOR</u></b>			
<b>No</b>	<b>Jenis Beras</b>	<b>Kode Beras</b>	<b>Harga Satuan (Karung/10kg)</b>
1	C4 (Super)	C1234E	120000
2	Delanggu (Biasa)	D23HJA	115000
3	BM (Beras Merah)	A23EED	160000



---

**CONTOH DATA TOKO YG SUDAH DIISI :**  
**HANYA SIMULASI SAJA**

<b><u>STACK 1</u></b>						
NO	Jenis Beras	Kode Beras	Kode Unik	Harga Beras	Dari	Tanggal Sampai
1	BM	A23EED	KDR15A	160000	Distributor	2017-01-06
2	C4	C1234E	URMP26	120000	Distributor	2017-01-10
3	Delanggu	D23HJA	2QYTR7	115000	Distributor	2017-01-15

<b><u>STACK 2</u></b>						
NO	Jenis Beras	Kode Beras	Kode Unik	Harga Beras	Dari	Tanggal Sampai
1	Delanggu	D23HJA	URTQ12	115000	Distributor	2017-02-01
2	C4	C1234E	2TY3DA	120000	Distributor	2017-02-04
3	BM	A23EED	UIRT07	160000	Distributor	2017-02-10

<b><u>STACK 3</u></b>						
NO	Jenis Beras	Kode Beras	Kode Unik	Harga Beras	Dari	Tanggal Sampai
1	C4	C1234E	HJL23K	120000	Distributor	2017-03-01
2	C4	C1234E	90OKAW	120000	Distributor	2017-03-12
3	BM	A23EED	LADJV2	160000	Distributor	2017-03-25

---

**Target Proyek :**

Target proyek secara umumnya adalah membantu kesusahan yang terjadi dengan membuat program proyek ini , yang awalnya ditulis secara manual atau kalkulator menjadi auto aplikasi/program. Dan target proyek dalam program adalah membantu memudahkan dalam menambah data kedalam tumpukan toko serta manajemen penjualan yang fleksibel dari stack toko ke masyarakat.

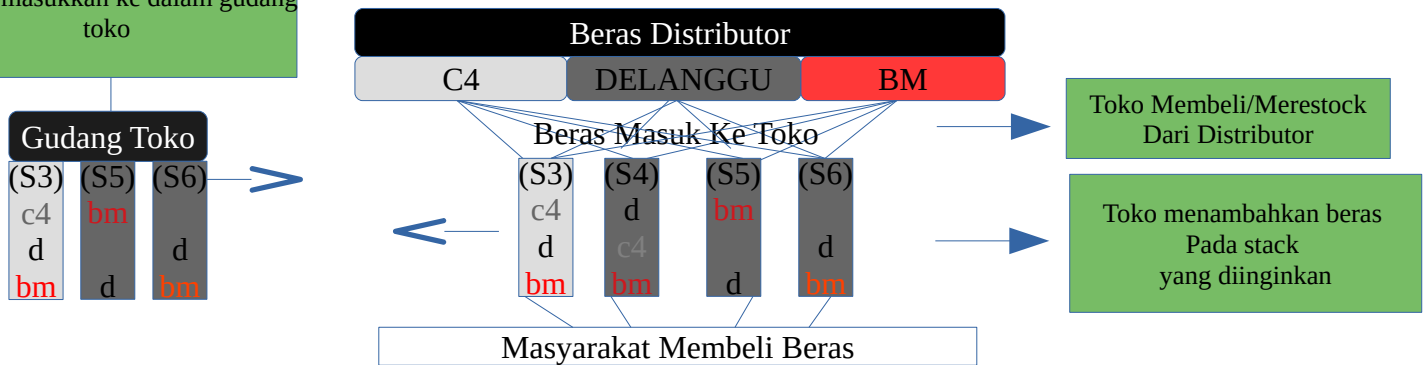
## BAB 2

### Pembahasan

#### A.Simulasi Proses Bisnis

Jika ada tumpukan kelebihan di toko  
Maka tumpukan yang kelebihan  
Akan di masukkan ke dalam gudang  
toko

CONTOH PROSES BISNISNYA :  
Untuk penjelasan lebih, ada pada deksripsi sebelumnya.



Jadi, singkatnya : Distributor → Toko → Masyarakat

Detail :

- Pada distributor terdapat 3 jenis beras yaitu (C4,Delanggu, dan BM).
- Toko membeli pada distributor beras yang ingin dibeli.
- Distributor menyalurkannya ke toko dan toko berhak memilih stack/tumpukan yang ingin diisi.
- Apabila stack di toko sudah penuh maka, akan di alihkan ke stack/tumpukan yang digudang.
- Jika masyarakat ingin membeli maka beras yang di ambil adalah yang paling atas.
- Stack/tumpukan akan di proses ke jual beli.

#### B.Program Dan Penjelasan

##### \*data.h (Header)

```
#include <string> // agar string bisa di deklarasikan di file header
using namespace std; //pemendekan std::
#ifdef data //apabila mendefinisikan "data"
#define data //definisikan data

class datas{
private:
    //(ADT) yang nantinya akan digunakan pada menu
    typedef int pilih; //Merupakan tipe data abstract (ADT) khusus int
    typedef char pilih2; //Merupakan tipe data abstract (ADT) khusus char
public:
    //Penjelasan selanjutnya akan ada di .cpp
    void masuk(int pil);
```



```

void distrib();
void saldo();
void menus();
int getCount(int pil2);
void gstack();
void s1(int pil, int pil2),s2(int pil, int pil2),
s3(int pil, int pil2),s4(int pil, int pil2);
void gs(int pil, int pil2);
////////// Pemisah fungsi
void read();
void input(int nomor, string c4, string gc4, string kc4, int kodeunik, long
long int hc4, string from, string waktu, int input);
void tampilkan();
void ts1(),ts2(),ts3(),ts4(),tgs1(),tgs2();
////////// end
void
stack1beli(),stack2beli(),stack3beli(),stack4beli(),gstack1beli(),gstack2beli();
void beliberas();
void backtomenus();
};

#endif

```

---

END OF data.h (HEADER)

---

### \*TPFinal.cpp (Bodynya)

---

```

#include <iostream> //fungsi i/o
#include <stdlib.h>
#include <cstdlib>
#include <windows.h> //pemanggilan system(cls) atau clearscreen
#include <fstream> //untuk akses file
#include <time.h> //untuk waktu
#include <sstream> //untuk penkonversian suatu int/char ke string
#include <ctime> //untuk waktu dalam time clock
#include <iomanip> //mempermudah dalam pembuatan tabel
#include "data.h" //file header

using namespace std; //penyingkatan std::

time_t waktu; //deklarasi pemanggilan waktu dalam format time

////////// PANGGIL/BUTUH/NEEDED UNIVERSAL
string tgl(){ //mendeklarasikan tanggal dalam string ::dd::mm::yy
    time_t t = time(0); //panggil waktu sekarang
    struct tm * now = localtime( & t );
    stringstream ss,aa,bb;
    ss<<(now->tm_year + 1900); //tahun
    aa<<(now->tm_mon + 1); //bulan
    bb<<(now->tm_mday); //hari
    string str1 = ss.str();
    string str2 = aa.str();
    string str3 = bb.str();
    string str1 = str3+"-"+str2+"-"+str1; //konversikan ke string
    return str1; //fungsi mengirimkan ke return
}

```

```

}

int uniquecode(){ //Menggenerate kode unik dari 10000 sampai 99999 (5digit)
    srand(time(NULL));
    int range = 99999 - 10000 + 1;
    int num = rand() % 99999 + 10000;
    return num;
}
//////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////END OF
UNIVERSAL

struct itemlist{ //penulisan item yang static dipisah agar mudah
    char c4[15] = "C4";char d[15] = "Delanggu";char bm[15] = "BerasMerah";
    char kc4[15] = "C1234E";char kd[15] = "D23HJA";char kbm[15] = "A23EED";
    char gc4[15] = "Super";char gd[15] = "Standart";char gbm[15] = "Spesial"; //grade
    char from[15] = "Distributor";
}it;

//Deklarasi banyak stack ( stack yang tersedia : 1,2,3,4,g1,g2 )
//setiap stack punya deklarasi sendiri dengan fungsi yang sama.
struct stack1{
    string c4= it.c4;string d = it.d;string bm = it.bm; //panggil item statis tadi
    string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;
    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode(); //ambil fungsi kodeunik tadi dan masukkan ke kodeunik
(stack)
    string from = it.from;
    string waktu = tgl(); //ambil fungsi tanggal tadi dan masukkan ke waktu
    int nomor;
    struct stack1* next;
    struct stack1* prev;
}; struct stack1 *head1 = NULL, *last1 = NULL, *curr1 = NULL; //buat head,akhir dan
current (sekarang)

//SETERUSNYA SAMA HINGGA KEBAWAH

struct stack2{
    string c4= it.c4;string d = it.d;string bm = it.bm;
    string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;
    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode();
    string from = it.from;
    string waktu = tgl();
    int nomor;
    struct stack2* next;
    struct stack2* prev;
}; struct stack2 *head2 = NULL, *last2 = NULL, *curr2 = NULL;

struct stack3{
    string c4= it.c4;string d = it.d;string bm = it.bm;
    string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;

```

```

    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode();
    string from = it.from;
    string waktu = tgl();
    int nomor;
        struct stack3* next;
        struct stack3* prev;
}; struct stack3 *head3 = NULL, *last3 = NULL, *curr3 = NULL;

struct stack4{
    string c4= it.c4;string d = it.d;string bm = it.bm;
        string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;
    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode();
    string from = it.from;
    string waktu = tgl();
    int nomor;
        struct stack4* next;
        struct stack4* prev;
}; struct stack4 *head4 = NULL, *last4 = NULL, *curr4 = NULL;

struct gstack1{
    string c4= it.c4;string d = it.d;string bm = it.bm;
        string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;
    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode();
    string from = it.from;
    string waktu = tgl();
    int nomor;
        struct gstack1* next;
        struct gstack1* prev;
}; struct gstack1 *ghead1 = NULL, *glast1 = NULL, *gcurr1 = NULL;

struct gstack2{
    string c4= it.c4;string d = it.d;string bm = it.bm;
        string kc4 = it.kc4;string kd = it.kd;string kbm = it.kbm;
    long long int hc4 = 0;long long int hd = 0;long long int hbm = 0;
    string gc4 = it.gc4;string gd = it.gd;string gbm = it.gbm;
    int kodeunik = uniquecode();
    string from = it.from;
    string waktu = tgl();
    int nomor;
        struct gstack2* next;
        struct gstack2* prev;
}; struct gstack2 *ghead2 = NULL, *glast2 = NULL, *gcurr2 = NULL;

```

```

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////

```

```

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
void datas::backtomenus(){ //prosedur untuk kembali ke menu (guna mempersingkat kode)
    pilih2 input;
    cout<<"Apa anda ingin kembali ke menu ? (Y/T) : ";cin>>input;
    if(input == 'y' || input == 'Y'){
        menus();
    }
    else{
        exit(0);
    }
}

void datas::beliberas(){ //menu untuk membeli beras
    pilih pil;
    system("cls");
    cout<<"|=====PILIH BERAS DI TUMPUKAN BERAPA=====|"<<endl;
    cout<<"1.Tumpukan 1 (Toko)"<<endl;
    cout<<"2.Tumpukan 2 (Toko)"<<endl;
    cout<<"3.Tumpukan 3 (Toko)"<<endl;
    cout<<"4.Tumpukan 4 (Toko)"<<endl;
    cout<<"5.Tumpukan 1 (Gudang)"<<endl;
    cout<<"6.Tumpukan 2 (Gudang)"<<endl;
    cout<<"Masukkan pilihan : ";cin>>pil;
    if(pil == 1){
        stack1beli();
    }
    else if(pil == 2){
        stack2beli();
    }
    else if(pil == 3){
        stack3beli();
    }
    else if(pil == 4){
        stack4beli();
    }
    else if(pil == 5){
        gstack1beli();
    }
    else if(pil == 6){
        gstack2beli();
    }
}

//////////Stack (nomor stack) beli : prosedur (void) untuk membeli stack
void datas::stack1beli(){
    struct stack1 *ptr = head1; //buat singkatan deklarasi baru untuk head1 (stack1
head)
    struct stack1 *hapus = NULL;
    pilih2 input; //panggil ADT di header file tadi (tipe data abstact)
    long long int hberas; //menggunakan long long int agar menjadi panjang intnya (lebih
dari 11 baris)

```

```

        ifstream ifsaldo("saldo.txt",ios::in); //buka saldo dalam fungsi nor(Dan Atau)
ifstream
        ifsaldo>>hberas;
        system("cls");
        if(ptr != NULL){
            cout<<"||===Beras yang tersedia sekarang===||"<<endl;
            //apabila di deteksi beras C4 maka hapus(pop) <- berupa tampilan saja
            if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
                //pop " nomor,c4,gc4,kc4,kodeunik,hc4
                cout<<"Tumpukan No : "<<ptr->nomor<<endl; //output ambil di ptr (head)
untuk nomor
                cout<<"Nama          : "<<ptr->c4<<endl; // -- untuk beras
                cout<<"Grade          : "<<ptr->gc4<<endl; // untuk grade
                cout<<"Kode Beras       : "<<ptr->kc4<<endl; // untuk kode beras
                cout<<"Kode Unik        : "<<ptr->kodeunik<<endl; // untuk kode unik
                cout<<"Harga Beras      : "<<ptr->hc4<<endl; // untuk harga beras
                hberas = hberas + (ptr->hc4); //ambil harga hc4
                ofstream ofsaldo("Saldo.txt", ios::out); //update harga -> masukkan ke
saldo
                ofsaldo<<hberas;
                ofsaldo.close(); //close agar memori aman
                cout<<"Dari          : "<<ptr->from<<endl; // ambil form di stack struct
                cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
                cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl; // ambil
waktu sekarang langsung dari fungsi
                cout<<endl;
            }
            //DIBAWAH DAN SETERUSNYA SAMA, HANYA SAJA
            //apabila di deteksi beras Delanggu maka hapus(pop) <- berupa tampilan saja
            else if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
                cout<<"Tumpukan No : "<<ptr->nomor<<endl;
                cout<<"Nama          : "<<ptr->d<<endl;
                cout<<"Grade          : "<<ptr->gd<<endl;
                cout<<"Kode Beras       : "<<ptr->kd<<endl;
                cout<<"Kode Unik        : "<<ptr->kodeunik<<endl;
                cout<<"Harga Beras      : "<<ptr->hd<<endl;
                hberas = hberas + (ptr->hd);
                ofstream ofsaldo("Saldo.txt", ios::out);
                ofsaldo<<hberas;
                ofsaldo.close();
                cout<<"Dari          : "<<ptr->from<<endl;
                cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
                cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
                cout<<endl;
            }
            //apabila di deteksi beras Merah maka hapus(pop) <- berupa tampilan saja
            else if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
                cout<<"Tumpukan No : "<<ptr->nomor<<endl;
                cout<<"Nama          : "<<ptr->bm<<endl;
                cout<<"Grade          : "<<ptr->gbm<<endl;
                cout<<"Kode Beras       : "<<ptr->kbm<<endl;
                cout<<"Kode Unik        : "<<ptr->kodeunik<<endl;
                cout<<"Harga Beras      : "<<ptr->hbm<<endl;
                hberas = hberas + (ptr->hbm);
                ofstream ofsaldo("Saldo.txt", ios::out);

```

```

        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

//konfirmasi beli

//JALANKAN FUNGSI HAPUS (POP) FROM TOP
if(input == 'Y' || input == 'y'){
    if(head1==NULL){
        cout<<"Kosong";
    }
    else if(head1->next==NULL){ //apabila head yang selanjutnya ==
NULL (0)
        hapus=head1;
        head1=NULL;
        last1=NULL;
        delete hapus; //hapus
    }
    else{
        hapus=ptr;
        head1=hapus->next;
        head1->prev=NULL;
        delete hapus; //hapus
    }
    cout<<"====Beras sudah dibeli===="<<endl;
    cout<<endl;
    ts1(); //panggil fungsi agar file langsung terupdate di .txt
}
else{
    cout<<"||===Anda tidak jadi membeli===||"<<endl;
    backtomenus();
}
}
else{
    cout<<"||===Maaf stock pada tumpukan 1 tidak ada===||"<<endl;
    backtomenus();
}
}

//SAMA SEPERTI STACK1 TADI HANYA SAJA INI STACK2//////////
void datas::stack2beli(){
    struct stack2 *ptr = head2;
    struct stack2 *hapus = NULL;
    pilih2 input;
    long long int hberas;
    ifstream ifsaldo("saldo.txt",ios::in);
    ifsaldo>>hberas;
    system("cls");
    if(ptr != NULL){
        cout<<"||===Beras yang tersedia sekarang===||"<<endl;
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;

```



```

        cout<<"Nama          : "<<ptr->c4<<endl;
        cout<<"Grade          : "<<ptr->gc4<<endl;
        cout<<"Kode Beras      : "<<ptr->kc4<<endl;
        cout<<"Kode Unik       : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras    : "<<ptr->hc4<<endl;
        hberas = hberas + (ptr->hc4);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama          : "<<ptr->d<<endl;
        cout<<"Grade          : "<<ptr->gd<<endl;
        cout<<"Kode Beras      : "<<ptr->kd<<endl;
        cout<<"Kode Unik       : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras    : "<<ptr->hd<<endl;
        hberas = hberas + (ptr->hd);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama          : "<<ptr->bm<<endl;
        cout<<"Grade          : "<<ptr->gbm<<endl;
        cout<<"Kode Beras      : "<<ptr->kbm<<endl;
        cout<<"Kode Unik       : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras    : "<<ptr->hbm<<endl;
        hberas = hberas + (ptr->hbm);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }

    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

    if(input == 'Y' || input == 'y'){
        if(head2==NULL){
            cout<<"Kosong";
        }
        else if(head2->next==NULL){
            hapus=ptr;
            head2=NULL;
        }
    }

```

```

        last2=NULL;
        delete hapus;
    }
    else{
        hapus=head2;
        head2=hapus->next;
        head2->prev=NULL;
        delete hapus;
    }
    cout<<"====Beras sudah dibeli===="<<endl;
    cout<<endl;
    ts2();
}
else{
    cout<<"||===Anda tidak jadi membeli===||"<<endl;
    backtomenus();
}
}
else{
    cout<<"||===Maaf stock pada tumpukan 1 tidak ada===||"<<endl;
    backtomenus();
}
}
}

```

```

void datas::stack3beli(){
    struct stack3 *ptr = head3;
    struct stack3 *hapus = NULL;
    pilih2 input;
    long long int hberas;
    ifstream ifsaldo("saldo.txt",ios::in);
    ifsaldo>>hberas;
    system("cls");
    if(ptr != NULL){
        cout<<"||===Beras yang tersedia sekarang===||"<<endl;
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->c4<<endl;
            cout<<"Grade : "<<ptr->gc4<<endl;
            cout<<"Kode Beras : "<<ptr->kc4<<endl;
            cout<<"Kode Unik : "<<ptr->kodeunik<<endl;
            cout<<"Harga Beras : "<<ptr->hc4<<endl;
            hberas = hberas + (ptr->hc4);
            ofstream ofsaldo("Saldo.txt", ios::out);
            ofsaldo<<hberas;
            ofsaldo.close();
            cout<<"Dari : "<<ptr->from<<endl;
            cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
            cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
            cout<<endl;
        }
        if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->d<<endl;
            cout<<"Grade : "<<ptr->gd<<endl;
            cout<<"Kode Beras : "<<ptr->kd<<endl;
        }
    }
}

```

```

        cout<<"Kode Unik    : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras : "<<ptr->hd<<endl;
        hberas = hberas + (ptr->hd);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari        : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama        : "<<ptr->bm<<endl;
        cout<<"Grade        : "<<ptr->gbm<<endl;
        cout<<"Kode Beras   : "<<ptr->kbm<<endl;
        cout<<"Kode Unik    : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras : "<<ptr->hbm<<endl;
        hberas = hberas + (ptr->hbm);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari        : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }

    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

    if(input == 'Y' || input == 'y'){
        if(head3==NULL){
            cout<<"Kosong";
        }
        else if(head3->next==NULL){
            hapus=head3;
            head3=NULL;
            last3=NULL;
            delete hapus;
        }
        else{
            hapus=head3;
            head3=hapus->next;
            head3->prev=NULL;
            delete hapus;
        }
        cout<<"====Beras sudah dibeli===="<<endl;
        cout<<endl;
        ts3();
    }
    else{
        cout<<"||====Anda tidak jadi membeli====||"<<endl;
        backtomenus();
    }
}
else{

```

```

        cout<<"||===Maaf stock pada tumpukan 1 tidak ada===||"<<endl;
        backtomenus();
    }
}

void datas::stack4beli(){
    struct stack4 *ptr = head4;
    struct stack4 *hapus = NULL;
    pilih2 input;
    long long int hberas;
    ifstream ifsaldo("saldo.txt",ios::in);
    ifsaldo>>hberas;
    system("cls");
    if(ptr != NULL){
        cout<<"||===Beras yang tersedia sekarang===||"<<endl;
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->c4<<endl;
            cout<<"Grade : "<<ptr->gc4<<endl;
            cout<<"Kode Beras : "<<ptr->kc4<<endl;
            cout<<"Kode Unik : "<<ptr->kodeunik<<endl;
            cout<<"Harga Beras : "<<ptr->hc4<<endl;
            hberas = hberas + (ptr->hc4);
            ofstream ofsaldo("Saldo.txt", ios::out);
            ofsaldo<<hberas;
            ofsaldo.close();
            cout<<"Dari : "<<ptr->from<<endl;
            cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
            cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
            cout<<endl;
        }
        if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->d<<endl;
            cout<<"Grade : "<<ptr->gd<<endl;
            cout<<"Kode Beras : "<<ptr->kd<<endl;
            cout<<"Kode Unik : "<<ptr->kodeunik<<endl;
            cout<<"Harga Beras : "<<ptr->hd<<endl;
            hberas = hberas + (ptr->hd);
            ofstream ofsaldo("Saldo.txt", ios::out);
            ofsaldo<<hberas;
            ofsaldo.close();
            cout<<"Dari : "<<ptr->from<<endl;
            cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
            cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
            cout<<endl;
        }
        if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->bm<<endl;
            cout<<"Grade : "<<ptr->gbm<<endl;
            cout<<"Kode Beras : "<<ptr->kbm<<endl;
            cout<<"Kode Unik : "<<ptr->kodeunik<<endl;
            cout<<"Harga Beras : "<<ptr->hbm<<endl;
            hberas = hberas + (ptr->hbm);
        }
    }
}

```

```

        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }

    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

    if(input == 'Y' || input == 'y'){
        if(head4==NULL){
            cout<<"Kosong";
        }
        else if(head4->next==NULL){
            hapus=head4;
            ptr=NULL;
            last4=NULL;
            delete hapus;
        }
        else{
            hapus=head4;
            head4=hapus->next;
            head4->prev=NULL;
            delete hapus;
        }
        cout<<"====Beras sudah dibeli===="<<endl;
        cout<<endl;
        ts4();
    }
    else{
        cout<<"||===Anda tidak jadi membeli===||"<<endl;
        backtomenus();
    }
}
else{
    cout<<"||===Maaf stock pada tumpukan 1 tidak ada===||"<<endl;
    backtomenus();
}
}
}

```

//stack gudang kebawah //////////

```

void datas::gstack1beli(){
    struct gstack1 *ptr = ghead1;
    struct gstack1 *hapus = NULL;
    pilih2 input;
    long long int hberas;
    ifstream ifsaldo("saldo.txt",ios::in);
    ifsaldo>>hberas;
    system("cls");
    if(ptr != NULL){
        cout<<"||===Beras yang tersedia sekarang===||"<<endl;
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        }
    }
}

```

```

        cout<<"Nama          : "<<ptr->c4<<endl;
        cout<<"Grade          : "<<ptr->gc4<<endl;
        cout<<"Kode Beras    : "<<ptr->kc4<<endl;
        cout<<"Kode Unik     : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras   : "<<ptr->hc4<<endl;
        hberas = hberas + (ptr->hc4);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama          : "<<ptr->d<<endl;
        cout<<"Grade          : "<<ptr->gd<<endl;
        cout<<"Kode Beras    : "<<ptr->kd<<endl;
        cout<<"Kode Unik     : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras   : "<<ptr->hd<<endl;
        hberas = hberas + (ptr->hd);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama          : "<<ptr->bm<<endl;
        cout<<"Grade          : "<<ptr->gbm<<endl;
        cout<<"Kode Beras    : "<<ptr->kbm<<endl;
        cout<<"Kode Unik     : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras   : "<<ptr->hbm<<endl;
        hberas = hberas + (ptr->hbm);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari          : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }

    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

    if(input == 'Y' || input == 'y'){
        if(ghead1==NULL){
            cout<<"Kosong";
        }
        else if(ghead1->next==NULL){
            hapus=ghead1;
            ghead1=NULL;
        }
    }

```



```

        glast1=NULL;
        delete hapus;
    }
    else{
        hapus=ghead1;
        ghead1=hapus->next;
        ghead1->prev=NULL;
        delete hapus;
    }
    cout<<"====Beras sudah dibeli===="<<endl;
    cout<<endl;
    tgs1();
}
else{
    cout<<"||===Anda tidak jadi membeli===||"<<endl;
    backtomenus();
}
}
else{
    cout<<"||===Maaf stock pada tumpukan 1 tidak ada===||"<<endl;
    backtomenus();
}
}
}

```

```

void datas::gstack2beli(){
    struct gstack2 *ptr = ghead2;
    struct gstack2 *hapus = NULL;
    pilih2 input;
    long long int hberas;
    ifstream ifsaldo("saldo.txt",ios::in);
    ifsaldo>>hberas;
    system("cls");
    if(ptr != NULL){
        cout<<"||===Beras yang tersedia sekarang===||"<<endl;
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->c4<<endl;
            cout<<"Grade : "<<ptr->gc4<<endl;
            cout<<"Kode Beras : "<<ptr->kc4<<endl;
            cout<<"Kode Unik : "<<ptr->kodeunik<<endl;
            cout<<"Harga Beras : "<<ptr->hc4<<endl;
            hberas = hberas + (ptr->hc4);
            ofstream ofsaldo("Saldo.txt", ios::out);
            ofsaldo<<hberas;
            ofsaldo.close();
            cout<<"Dari : "<<ptr->from<<endl;
            cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
            cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
            cout<<endl;
        }
        if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
            cout<<"Tumpukan No : "<<ptr->nomor<<endl;
            cout<<"Nama : "<<ptr->d<<endl;
            cout<<"Grade : "<<ptr->gd<<endl;
            cout<<"Kode Beras : "<<ptr->kd<<endl;
        }
    }
}

```

```

        cout<<"Kode Unik    : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras : "<<ptr->hd<<endl;
        hberas = hberas + (ptr->hd);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari        : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"Tumpukan No : "<<ptr->nomor<<endl;
        cout<<"Nama          : "<<ptr->bm<<endl;
        cout<<"Grade          : "<<ptr->gbm<<endl;
        cout<<"Kode Beras    : "<<ptr->kbm<<endl;
        cout<<"Kode Unik     : "<<ptr->kodeunik<<endl;
        cout<<"Harga Beras    : "<<ptr->hbm<<endl;
        hberas = hberas + (ptr->hbm);
        ofstream ofsaldo("Saldo.txt", ios::out);
        ofsaldo<<hberas;
        ofsaldo.close();
        cout<<"Dari        : "<<ptr->from<<endl;
        cout<<"Tersedia dari tanggal ("<<ptr->waktu<<")"<<endl;
        cout<<"Akan dibeli sekarang pada tanggal ("<<tgl()<<")"<<endl;
        cout<<endl;
    }

    cout<<"Apa anda yakin ingin membeli ? (Y/T) : ";cin>>input;

    if(input == 'Y' || input == 'y'){
        if(ghead2==NULL){
            cout<<"Kosong";
        }
        else if(ghead2->next==NULL){
            hapus=ghead2;
            ghead2=NULL;
            glast2=NULL;
            delete hapus;
        }
        else{
            hapus=ghead2;
            ghead2=hapus->next;
            ghead2->prev=NULL;
            delete hapus;
        }
        cout<<"====Beras sudah dibeli===="<<endl;
        cout<<endl;
        tgs2();
    }
    else{
        cout<<"||===Anda tidak jadi membeli===||"<<endl;
        backtomenus();
    }
}
else{

```

```

        cout<<"||==Maaf stock pada tumpukan 1 tidak ada==||"<<endl;
        backtomenus();
    }
}

////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////
////////////////////////////////////

void datas::ts1(){ //Tampil stack (no) : digunakan untuk menampilkan stack dan
memasukkannya ke file
    struct stack1 *ptr = head1; // ptr sebagai head
    ofstream cekin("stack1.txt",ios::out); //panggil file stack(no) .txt

    cout<<"=====
===== "<<endl;
        cout<<" | No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari | DD-MM-YYYY | "<<endl;

    cout<<"-----
----- "<<endl;
        while(ptr != NULL){ // apabila ptr(head) tidak == Null (Kosong) maka jalankan
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){ //Untuk
beras C4 apabila terdeteksi
            cout<<"| "<<setw(5)<<ptr->nomor<<"|"; // gunakan setw(no) untuk spacing
khusus tabel dan panggil head-> data yang diinginkan
            cout<<setw(13)<<ptr->c4<<"|";
            cout<<setw(9)<<ptr->gc4<<"|";
            cout<<setw(12)<<ptr->kc4<<"|";
            cout<<setw(11)<<ptr->kodeunik<<"|";
            cout<<setw(9)<<ptr->hc4<<"|";
            cout<<setw(12)<<ptr->from<<"|";
            cout<<setw(12)<<ptr->waktu<<"|";
            cout<<endl;
            //////////////////////////////////
            cekin<<ptr->nomor<<" "; //jangan lupa masukkkan kefile (input selalu
file agar update datanya)
            cekin<<ptr->c4<<" ";
            cekin<<ptr->gc4<<" ";
            cekin<<ptr->kc4<<" ";
            cekin<<ptr->kodeunik<<" ";
            cekin<<ptr->hc4<<" ";
            cekin<<ptr->from<<" ";
            cekin<<ptr->waktu<<" ";

```

```

        cekin<<endl;
    }
    //SETERUSNYA SAMA HINGGA BAWAH
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){ //Untuk
beras Delanggu apabila terdeteksi
        cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
        cout<<setw(13)<<ptr->d<<"| ";
        cout<<setw(9)<<ptr->gd<<"| ";
        cout<<setw(12)<<ptr->kd<<"| ";
        cout<<setw(11)<<ptr->kodeunik<<"| ";
        cout<<setw(9)<<ptr->hd<<"| ";
        cout<<setw(12)<<ptr->from<<"| ";
        cout<<setw(12)<<ptr->waktu<<"| ";
        cout<<endl;
        ///////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->d<<" ";
        cekin<<ptr->gd<<" ";
        cekin<<ptr->kd<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hd<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
//Untuk Beras Merah apabila terdeteksi
        cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
        cout<<setw(13)<<ptr->bm<<"| ";
        cout<<setw(9)<<ptr->gbm<<"| ";
        cout<<setw(12)<<ptr->kbm<<"| ";
        cout<<setw(11)<<ptr->kodeunik<<"| ";
        cout<<setw(9)<<ptr->hbm<<"| ";
        cout<<setw(12)<<ptr->from<<"| ";
        cout<<setw(12)<<ptr->waktu<<"| ";
        cout<<endl;
        ///////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    ptr = ptr->next;
}

cout<<"=====
===== "<<endl;
    cout<<"Jumlah Semua : "<<(getCount(1)-1)<<endl;
    backtomenus();

```

```

}

//////////SETERUSNYA SAMA HINGGA BAWAH
void datas::ts2(){
    struct stack2 *ptr = head2;
    ofstream cekin("stack2.txt",ios::out);

    cout<<"=====
===== "<<endl;
        cout<<" | No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari | DD-MM-YYYY | "<<endl;

    cout<<"-----
----- "<<endl;
        while(ptr != NULL){
            if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
                cout<<" | "<<setw(5)<<ptr->nomor<<" | ";
                cout<<setw(13)<<ptr->c4<<" | ";
                cout<<setw(9)<<ptr->gc4<<" | ";
                cout<<setw(12)<<ptr->kc4<<" | ";
                cout<<setw(11)<<ptr->kodeunik<<" | ";
                cout<<setw(9)<<ptr->hc4<<" | ";
                cout<<setw(12)<<ptr->from<<" | ";
                cout<<setw(12)<<ptr->waktu<<" | ";
                cout<<endl;
                //////////
                cekin<<ptr->nomor<<" ";
                cekin<<ptr->c4<<" ";
                cekin<<ptr->gc4<<" ";
                cekin<<ptr->kc4<<" ";
                cekin<<ptr->kodeunik<<" ";
                cekin<<ptr->hc4<<" ";
                cekin<<ptr->from<<" ";
                cekin<<ptr->waktu<<" ";
                cekin<<endl;
            }
            if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
                cout<<" | "<<setw(5)<<ptr->nomor<<" | ";
                cout<<setw(13)<<ptr->d<<" | ";
                cout<<setw(9)<<ptr->gd<<" | ";
                cout<<setw(12)<<ptr->kd<<" | ";
                cout<<setw(11)<<ptr->kodeunik<<" | ";
                cout<<setw(9)<<ptr->hd<<" | ";
                cout<<setw(12)<<ptr->from<<" | ";
                cout<<setw(12)<<ptr->waktu<<" | ";
                cout<<endl;
                //////////
                cekin<<ptr->nomor<<" ";
                cekin<<ptr->d<<" ";
                cekin<<ptr->gd<<" ";
                cekin<<ptr->kd<<" ";
                cekin<<ptr->kodeunik<<" ";
                cekin<<ptr->hd<<" ";
            }
        }
    }
}

```

```

        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
        cout<<setw(13)<<ptr->bm<<"| ";
        cout<<setw(9)<<ptr->gbm<<"| ";
        cout<<setw(12)<<ptr->kbm<<"| ";
        cout<<setw(11)<<ptr->kodeunik<<"| ";
        cout<<setw(9)<<ptr->hbm<<"| ";
        cout<<setw(12)<<ptr->from<<"| ";
        cout<<setw(12)<<ptr->waktu<<"| ";
        cout<<endl;
        ///////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    ptr = ptr->next;
}

cout<<"=====
===== "<<endl;
    cout<<"Jumlah Semua : "<<(getCount(2)-1)<<endl;
    backtomenus();
}

void datas::ts3(){
    struct stack3 *ptr = head3;
    ofstream cekin("stack3.txt",ios::out);

    cout<<"=====
===== "<<endl;
    cout<<"| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari | DD-MM-YYYY | "<<endl;

    cout<<"-----
----- "<<endl;
    while(ptr != NULL){
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
            cout<<setw(13)<<ptr->c4<<"| ";
            cout<<setw(9)<<ptr->gc4<<"| ";
            cout<<setw(12)<<ptr->kc4<<"| ";
            cout<<setw(11)<<ptr->kodeunik<<"| ";

```



```

        cout<<setw(9)<<ptr->hc4<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->c4<<" ";
        cekin<<ptr->gc4<<" ";
        cekin<<ptr->kc4<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hc4<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->d<<"|";
        cout<<setw(9)<<ptr->gd<<"|";
        cout<<setw(12)<<ptr->kd<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hd<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->d<<" ";
        cekin<<ptr->gd<<" ";
        cekin<<ptr->kd<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hd<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->bm<<"|";
        cout<<setw(9)<<ptr->gbm<<"|";
        cout<<setw(12)<<ptr->kbm<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hbm<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
    }

```

```

        cekin<<endl;
    }
    ptr = ptr->next;
}

cout<<"=====
===== "<<endl;
    cout<<"Jumlah Semua : "<<(getCount(3)-1)<<endl;
    backtomenus();
}

void datas::ts4(){
    struct stack4 *ptr = head4;
    ofstream cekin("stack4.txt",ios::out);

    cout<<"=====
===== "<<endl;
    cout<<"| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari    | DD-MM-YYYY | "<<endl;

    cout<<"-----
----- "<<endl;
    while(ptr != NULL){
        if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
            cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
            cout<<setw(13)<<ptr->c4<<"| ";
            cout<<setw(9)<<ptr->gc4<<"| ";
            cout<<setw(12)<<ptr->kc4<<"| ";
            cout<<setw(11)<<ptr->kodeunik<<"| ";
            cout<<setw(9)<<ptr->hc4<<"| ";
            cout<<setw(12)<<ptr->from<<"| ";
            cout<<setw(12)<<ptr->waktu<<"| ";
            cout<<endl;
            ///////////////////////////////////
            cekin<<ptr->nomor<<" ";
            cekin<<ptr->c4<<" ";
            cekin<<ptr->gc4<<" ";
            cekin<<ptr->kc4<<" ";
            cekin<<ptr->kodeunik<<" ";
            cekin<<ptr->hc4<<" ";
            cekin<<ptr->from<<" ";
            cekin<<ptr->waktu<<" ";
            cekin<<endl;
        }
        if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
            cout<<setw(5)<<ptr->nomor<<"| ";
            cout<<"| "<<setw(13)<<ptr->d<<"| ";
            cout<<setw(9)<<ptr->gd<<"| ";
            cout<<setw(12)<<ptr->kd<<"| ";
            cout<<setw(11)<<ptr->kodeunik<<"| ";
            cout<<setw(9)<<ptr->hd<<"| ";
            cout<<setw(12)<<ptr->from<<"| ";

```

```

        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->d<<" ";
        cekin<<ptr->gd<<" ";
        cekin<<ptr->kd<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hd<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->bm<<"|";
        cout<<setw(9)<<ptr->gbm<<"|";
        cout<<setw(12)<<ptr->kbm<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hbm<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    ptr = ptr->next;
}

cout<<"=====
===== "<<endl;
        cout<<"Jumlah Semua : "<<(getCount(4)-1)<<endl;
        backtomenus();
}

//////////STACK GUDANG ( PENJELASAN SAMA SEPERTI DI ATAS )
void datas::tgs1(){
    struct gstack1 *ptr = ghead1;
    ofstream cekin("gstack1.txt",ios::out);

    cout<<"=====
===== "<<endl;
        cout<<"| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari | DD-MM-YYYY | "<<endl;

```

```

cout<<"-----"
-----"<<endl;
while(ptr != NULL){
    if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
        cout<<"|"<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->c4<<"|";
        cout<<setw(9)<<ptr->gc4<<"|";
        cout<<setw(12)<<ptr->kc4<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hc4<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->c4<<" ";
        cekin<<ptr->gc4<<" ";
        cekin<<ptr->kc4<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hc4<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
        cout<<"|"<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->d<<"|";
        cout<<setw(9)<<ptr->gd<<"|";
        cout<<setw(12)<<ptr->kd<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hd<<"|";
        cout<<setw(12)<<ptr->from<<"|";
        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->d<<" ";
        cekin<<ptr->gd<<" ";
        cekin<<ptr->kd<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hd<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"|"<<setw(5)<<ptr->nomor<<"|";
        cout<<setw(13)<<ptr->bm<<"|";
        cout<<setw(9)<<ptr->gbm<<"|";
        cout<<setw(12)<<ptr->kbm<<"|";
        cout<<setw(11)<<ptr->kodeunik<<"|";
        cout<<setw(9)<<ptr->hbm<<"|";
        cout<<setw(12)<<ptr->from<<"|";
    }
}

```

```

        cout<<setw(12)<<ptr->waktu<<"|";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    ptr = ptr->next;
}

cout<<"=====
===== "<<endl;
        cout<<"Jumlah Semua : "<<(getCount(5)-1)<<endl;
        backtomenus();
}

void datas::tgs2(){
    struct gstack2 *ptr = ghead2;
    ofstream cekin("gstack2.txt",ios::out);

    cout<<"=====
===== "<<endl;
        cout<<"| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga |
Dari    | DD-MM-YYYY | "<<endl;

    cout<<"-----
----- "<<endl;
        while(ptr != NULL){
            if(ptr->gc4 == "Super" && ptr->kc4 == "C1234E" && ptr->hc4 != NULL){
                cout<<"| "<<setw(5)<<ptr->nomor<<"|";
                cout<<setw(13)<<ptr->c4<<"|";
                cout<<setw(9)<<ptr->gc4<<"|";
                cout<<setw(12)<<ptr->kc4<<"|";
                cout<<setw(11)<<ptr->kodeunik<<"|";
                cout<<setw(9)<<ptr->hc4<<"|";
                cout<<setw(12)<<ptr->from<<"|";
                cout<<setw(12)<<ptr->waktu<<"|";
                cout<<endl;
                //////////////////////////////////
                cekin<<ptr->nomor<<" ";
                cekin<<ptr->c4<<" ";
                cekin<<ptr->gc4<<" ";
                cekin<<ptr->kc4<<" ";
                cekin<<ptr->kodeunik<<" ";
                cekin<<ptr->hc4<<" ";
                cekin<<ptr->from<<" ";
            }
        }
    }
}

```

```

        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gd == "Standart" && ptr->kd == "D23HJA" && ptr->hd != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
        cout<<setw(13)<<ptr->d<<"| ";
        cout<<setw(9)<<ptr->gd<<"| ";
        cout<<setw(12)<<ptr->kd<<"| ";
        cout<<setw(11)<<ptr->kodeunik<<"| ";
        cout<<setw(9)<<ptr->hd<<"| ";
        cout<<setw(12)<<ptr->from<<"| ";
        cout<<setw(12)<<ptr->waktu<<"| ";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->d<<" ";
        cekin<<ptr->gd<<" ";
        cekin<<ptr->kd<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hd<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    if(ptr->gbm == "Spesial" && ptr->kbm == "A23EED" && ptr->hbm != NULL){
        cout<<"| "<<setw(5)<<ptr->nomor<<"| ";
        cout<<setw(13)<<ptr->bm<<"| ";
        cout<<setw(9)<<ptr->gbm<<"| ";
        cout<<setw(12)<<ptr->kbm<<"| ";
        cout<<setw(11)<<ptr->kodeunik<<"| ";
        cout<<setw(9)<<ptr->hbm<<"| ";
        cout<<setw(12)<<ptr->from<<"| ";
        cout<<setw(12)<<ptr->waktu<<"| ";
        cout<<endl;
        //////////////////////////////////
        cekin<<ptr->nomor<<" ";
        cekin<<ptr->bm<<" ";
        cekin<<ptr->gbm<<" ";
        cekin<<ptr->kbm<<" ";
        cekin<<ptr->kodeunik<<" ";
        cekin<<ptr->hbm<<" ";
        cekin<<ptr->from<<" ";
        cekin<<ptr->waktu<<" ";
        cekin<<endl;
    }
    ptr = ptr->next;
}

```

```

cout<<"=====
===== "<<endl;
        cout<<"Jumlah Semua : "<<(getCount(6)-1)<<endl;
        backtomenus();
    }

```



```

void datas::tampilkan(){ //Menu untuk menampilkan data pada setiap stack/tumpukan
    system("cls");
    int pil;char pil2; //deklarasi baru , digunakan untuk txt
    char penyangga[100];
    int MAKS = 100;
    cout<<"|=====TAMPILKAN=====|"<<endl;
    cout<<"1.Data Tumpukan 1 (Toko)"<<endl;
    cout<<"2.Data Tumpukan 2 (Toko)"<<endl;
    cout<<"3.Data Tumpukan 3 (Toko)"<<endl;
    cout<<"4.Data Tumpukan 4 (Toko)"<<endl;
    cout<<"5.Data Tumpukan 1 (Gudang) "<<endl;
    cout<<"6.Data Tumpukan 2 (Gudang)"<<endl<<endl;
    cout<<"Masukkan Pilihan : ";cin>>pil;
    if(pil == 1){
        ts1(); //Ambil fungsi yang di atas tadi
    }
    else if(pil == 2){
        ts2();
    }
    else if(pil == 3){
        ts3();
    }
    else if(pil == 4){
        ts4();
    }
    else if(pil == 5){
        tgs1();
    }
    else if(pil == 6){
        tgs2();
    }
    else{
        exit(0);
    }
}

//FUNGSI KHUSUS UNTUK READ AWAL C++
//ambil seluruh data dari .txt dan salurkan ke input VVVVVVVVVVVV
void datas::input(int nomor, string c4, string gc4, string kc4, int kodeunik, long long
int hc4, string from, string waktu, int input){
    if(input == 1){ //apabila yang dipilih == 1 (Stack 1) maka
        struct stack1*new_node = (struct stack1*)malloc(sizeof(struct stack1));
//deklarasi stack (ini program lama atau dalam bahasa "C")
        new_node = new(stack1); //Bahasa "C++" (Terbaru)
        cout<<endl;
        new_node->nomor = nomor; //MASUKKAN KE SETIAP NODE DATA DATA yang di .txt
tadi
        new_node->c4 = c4;
        new_node->gc4 = gc4;
        new_node->kc4 = kc4;
        new_node->kodeunik = kodeunik;
        new_node->from = from;
        new_node->>waktu = waktu;
        new_node->hc4 = hc4;
        new_node->next = head1;

```

```

new_node->prev = NULL;

//////////MERUPAKAN RUMUS DOUBLY LINKED LIST
if(head1 != NULL){ //apabila head tidak == NULL (Kosong) maka
head1->prev = new_node; //yang sebelumnya == new_node
}

head1 = new_node; //masukkan new_node ke head
}

//////////PENJELASANNYA SAMA HINGGA KEBAWAH
else if(input == 2){ // stack 2
    struct stack2*new_node = (struct stack2*)malloc(sizeof(struct stack2));
    new_node = new(stack2);
    cout<<endl;
    new_node->nomor = nomor;
    new_node->c4 = c4;
    new_node->gc4 = gc4;
    new_node->kc4 = kc4;
    new_node->kodeunik = kodeunik;
    new_node->from = from;
    new_node->>waktu = waktu;
    new_node->hc4 = hc4;
    new_node->next = head2;
    new_node->prev = NULL;

    if(head2 != NULL){
        head2->prev = new_node;
    }

    head2 = new_node;
}
else if(input == 3){ //stack 5
    struct stack3*new_node = (struct stack3*)malloc(sizeof(struct stack3));
    new_node = new(stack3);
    cout<<endl;
    new_node->nomor = nomor;
    new_node->c4 = c4;
    new_node->gc4 = gc4;
    new_node->kc4 = kc4;
    new_node->kodeunik = kodeunik;
    new_node->from = from;
    new_node->>waktu = waktu;
    new_node->hc4 = hc4;
    new_node->next = head3;
    new_node->prev = NULL;

    if(head3 != NULL){
        head3->prev = new_node;
    }

    head3 = new_node;
}
else if(input == 4){ //stack 4
    struct stack4*new_node = (struct stack4*)malloc(sizeof(struct stack4));

```

```

        new_node = new(stack4);
        cout<<endl;
        new_node->nomor = nomor;
        new_node->c4 = c4;
        new_node->gc4 = gc4;
        new_node->kc4 = kc4;
        new_node->kodeunik = kodeunik;
        new_node->from = from;
        new_node->>waktu = waktu;
        new_node->hc4 = hc4;
        new_node->next = head4;
        new_node->prev = NULL;

        if(head4 != NULL){
            head4->prev = new_node;
        }

        head4 = new_node;
    }
    else if(input == 5){ //stack gudang 1
        struct gstack1*new_node = (struct gstack1*)malloc(sizeof(struct gstack1));
        new_node = new(gstack1);
        cout<<endl;
        new_node->nomor = nomor;
        new_node->c4 = c4;
        new_node->gc4 = gc4;
        new_node->kc4 = kc4;
        new_node->kodeunik = kodeunik;
        new_node->from = from;
        new_node->>waktu = waktu;
        new_node->hc4 = hc4;
        new_node->next = ghead1;
        new_node->prev = NULL;

        if(ghead1 != NULL){
            ghead1->prev = new_node;
        }

        ghead1 = new_node;
    }
    else if(input == 6){ //stack gudang 2
        struct gstack2*new_node = (struct gstack2*)malloc(sizeof(struct gstack2));
        new_node = new(gstack2);
        cout<<endl;
        new_node->nomor = nomor;
        new_node->c4 = c4;
        new_node->gc4 = gc4;
        new_node->kc4 = kc4;
        new_node->kodeunik = kodeunik;
        new_node->from = from;
        new_node->>waktu = waktu;
        new_node->hc4 = hc4;
        new_node->next = ghead2;
        new_node->prev = NULL;
    }
}

```



```

int count = 1; // masukkan countnya
if(pil2 == 1){
    struct stack1* curr1 = head1; // deklarasikan curr(no) tadi == head(no)
    while (curr1 != NULL) //Selama curr(head) tadi tidak == NULL (tidak kosong) maka
count akan terus di tambah hingga curr(head) kosong
    {
        count++;
        curr1 = curr1->next;
    }
    return count; // return count karena ini fungsi
}

//PENJELASANNYA SAMA
else if(pil2 == 2){
    struct stack2* curr2 = head2;
    while (curr2 != NULL)
    {
        count++;
        curr2 = curr2->next;
    }
    return count;
}

else if(pil2 == 3){
    struct stack3* curr3 = head3; // Initialize current
    while (curr3 != NULL)
    {
        count++;
        curr3 = curr3->next;
    }
    return count;
}

else if(pil2 == 4){
    struct stack4* curr4 = head4; // Initialize current
    while (curr4 != NULL)
    {
        count++;
        curr4 = curr4->next;
    }
    return count;
}

else if(pil2 == 5){
    struct gstack1* gcurr1 = ghead1; // Initialize current
    while (gcurr1 != NULL)
    {
        count++;
        gcurr1 = gcurr1->next;
    }
    return count;
}

else if(pil2 == 6){
    struct gstack2* gcurr2 = ghead2; // Initialize current

```

```

        while (gcurr2 != NULL)
        {
            count++;
            gcurr2 = gcurr2->next;
        }
        return count;
    }
}

void datas::gs(int pil, int pil2){ //Gudang stack (stack gudang)
    system("cls");
    char pil3;
    int pil4;
    cout<<"||===Data di tumpukan anda pilih sudah penuh===||"<<endl;
    cout<<"Apa anda ingin memasukkan data yang berlebih ke gudang ? (Y/T) : ";cin>>pil3;
    if(pil3 == 'y' || pil3 == 'Y'){
        cout<<"Di gudang ada 2 tumpukan tersisa silahkan masukkan :
(1/2) :";cin>>pil4;
        if(pil4 == 1){ //stack gudang 1
            struct gstack1*new_node = (struct gstack1*)malloc(sizeof(struct
gstack1)); //"C"
            new_node = new(gstack1); // deklarasi new_node untuk stack(no) "C++"
            ofstream infile;
            infile.open("gstack1.txt",ios::out); //Panggil stack1 untuk ifstream
            long long int uang;
            ifstream in("saldo.txt",ios::in); //panggil saldo untuk arithmetic
(perhitungan nantinya)
            in>>uang; //masukkan data di txt yg berupa saldo (abgka) ke uang
            in.close(); // jangan lupa close
            cout<<endl;
            if(getCount(5) < 11){ // apabila stack untuk gudang(no) dibawah max
atau penuh maka jalankan
                if(pil == 1 && uang >= 120000){ // cek requirements
uang ,apabila terpenuhi dan pilihan == 1 maka jalankan
                    long long int harga = 120000;
                    int hasil = uang-harga; // uang tadi dikurangi harga beras
                    ofstream outsaldo;
                    outsaldo.open("Saldo.txt", ios::out);
                    outsaldo<<hasil; //masukkan hasil tadi ke saldo kembali
( dengan ios::out ) agar data terupdate (bukan tereplace)
                    outsaldo.close(); //close
                    cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl; // masukkan data ke new_node
                    cout<<"Jenis Beras                : "<<new_node->c4<<endl;
                    cout<<"Kualitas                    : "<<new_node->gc4<<endl;
                    cout<<"Kode Beras                  : "<<new_node->kc4<<endl;
                    cout<<"Kode Unik                    : "<<new_node->
>kodeunik<<endl;

                    cout<<"Harga Distributor : "<<harga<<endl;
                    cout<<"Dari                    : "<<new_node->from<<endl;
                    cout<<"Waktu                    : "<<new_node->
>waktu<<endl;

                    cout<<"Harga Beras                :
";cin>>new_node->hc4;cout<<endl; // input dan masukkan ke new_node
/////////////////OPERASI FILE/////////////////

```

```

        infile<<(new_node->nomor = getCount(pil2))<<" "; //jangan
lupa masukkan juga ke file ,agar nanti bisa operasi file
        infile<<new_node->c4<<" ";
        infile<<new_node->gc4<<" ";
        infile<<new_node->kc4<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hc4<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->>waktu<<endl;
    }
    //////////PENJELASANNYA SAMA HINGGA BAWAH
else if(pil == 2 && uang >= 115000){
    long long int harga = 115000;
    int hasil = uang-harga;
    ofstream outsaldo;
    outsaldo.open("Saldo.txt", ios::out);
    outsaldo<<hasil;
    outsaldo.close();
    cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;

    cout<<"Jenis Beras                : "<<new_node->d<<endl;
    cout<<"Kualitas                    : "<<new_node->gd<<endl;
    cout<<"Kode Beras                  : "<<new_node->kd<<endl;
    cout<<"Kode Unik                    : "<<new_node-
>kodeunik<<endl;

    cout<<"Harga Distributor  : "<<harga<<endl;
    cout<<"Dari                    : "<<new_node->from<<endl;
    cout<<"Waktu                    : "<<new_node-
>waktu<<endl;

    cout<<"Harga Beras                : ";cin>>new_node-

    //////////OPERASI FILE//////////
    infile<<(new_node->nomor = getCount(pil2))<<" ";
    infile<<new_node->d<<" ";
    infile<<new_node->gd<<" ";
    infile<<new_node->kd<<" ";
    infile<<new_node->kodeunik<<" ";
    infile<<new_node->hd<<" ";
    infile<<new_node->from<<" ";
    infile<<new_node->>waktu<<endl;
}
else if(pil == 3 && uang >= 160000){
    long long int harga = 160000;
    int hasil = uang-harga;
    ofstream outsaldo;
    outsaldo.open("Saldo.txt", ios::out);
    outsaldo<<hasil;
    outsaldo.close();
    cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;

    cout<<"Jenis Beras                : "<<new_node->bm<<endl;
    cout<<"Kualitas                    : "<<new_node->gbm<<endl;
    cout<<"Kode Beras                  : "<<new_node->kbm<<endl;
    cout<<"Kode Unik                    : "<<new_node-
>kodeunik<<endl;

```

```

        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                : "<<new_node->from<<endl;
        cout<<"Waktu                : "<<new_node->waktu<<endl;

        cout<<"Harga Beras          : ";cin>>new_node->hbm;cout<<endl;

        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->bm<<" ";
        infile<<new_node->gbm<<" ";
        infile<<new_node->kbm<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hbm<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else{
        //di jalankan jika uang tidak cukup, maka kembali ke menu
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) : ";cin>>pil3;

        if(pil3 == 'y' || pil3 == 'Y'){
            menus();
        }
        if(pil3 == 'T' || pil3 == 't'){
            exit(0);
        }
    }

    //jalankan rumus DOUBLY LINKED LIST
    new_node->next = ghead1;
    new_node->prev = NULL;

    if (ghead1 != NULL)
        ghead1->prev = new_node;

    ghead1 = new_node;
}
else{ //selain itu semya apa bila uang cukup tapi stack penuh
    pilih2 input;
    cout<<"||==Stack yang anda pilih sudah penuh==||"<<endl;
    cout<<"Apa anda ingin kembali ke menu atau tetap
    menginputkan ?"<<endl;

    cout<<"Menu/Input - (M/I) : ";cin>>input;
    if(input == 'M' || input == 'm'){
        menus();
    }
    else if(input == 'I' || input == 'i'){
        masuk(pil);
    }
}
}
}

```



```

//////////PENJELASANNYA SAMA
else if(pil4 == 2){
    struct gstack2*new_node = (struct gstack2*)malloc(sizeof(struct
gstack2));

    new_node = new(gstack2);
    ofstream infile;
    infile.open("gstack2.txt",ios::app);
    long long int uang;
    ifstream in("saldo.txt",ios::in);
    in>>uang;
    in.close();
    cout<<endl;
    if(getCount(6) < 11){
        if(pil == 1 && uang >= 120000){
            long long int harga = 120000;
            int hasil = uang-harga;
            ofstream outsaldo;
            outsaldo.open("Saldo.txt", ios::out);
            outsaldo<<hasil;
            outsaldo.close();
            cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;

            cout<<"Jenis Beras                : "<<(new_node->c4<<endl;
            cout<<"Kualitas                    : "<<(new_node->gc4<<endl;
            cout<<"Kode Beras                  : "<<(new_node->kc4<<endl;
            cout<<"Kode Unik                    : "<<(new_node->
>kodeunik<<endl;

            cout<<"Harga Distributor : "<<harga<<endl;
            cout<<"Dari                : "<<(new_node->from<<endl;
            cout<<"Waktu                : "<<(new_node->
>waktu<<endl;

            cout<<"Harga Beras            : ";cin>>new_node->
>hc4;cout<<endl;

            //////////OPERASI FILE//////////
            infile<<(new_node->nomor = getCount(pil2))<<" ";
            infile<<(new_node->c4<<" ";
            infile<<(new_node->gc4<<" ";
            infile<<(new_node->kc4<<" ";
            infile<<(new_node->kodeunik<<" ";
            infile<<(new_node->hc4<<" ";
            infile<<(new_node->from<<" ";
            infile<<(new_node->waktu<<endl;
        }
        else if(pil == 2 && uang >= 115000){
            long long int harga = 115000;
            int hasil = uang-harga;
            ofstream outsaldo;
            outsaldo.open("Saldo.txt", ios::out);
            outsaldo<<hasil;
            outsaldo.close();
            cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;

            cout<<"Jenis Beras                : "<<(new_node->d<<endl;
            cout<<"Kualitas                    : "<<(new_node->gd<<endl;
            cout<<"Kode Beras                  : "<<(new_node->kd<<endl;

```

```

        cout<<"Kode Unik                : "<<new_node-
>kodeunik<<endl;

        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                : "<<new_node->from<<endl;
        cout<<"Waktu                : "<<new_node-

>waktu<<endl;

        cout<<"Harga Beras          : ";cin>>new_node-
>hd;cout<<endl;

        ///////////////////////////////////OPERASI FILE/////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->d<<" ";
        infile<<new_node->gd<<" ";
        infile<<new_node->kd<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hd<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else if(pil == 3 && uang >= 160000){
        long long int harga = 160000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                    : "<<(new_node->nomor =
getCount(pil2))<<endl;

        cout<<"Jenis Beras          : "<<new_node->bm<<endl;
        cout<<"Kualitas              : "<<new_node->gbm<<endl;
        cout<<"Kode Beras            : "<<new_node->kbm<<endl;
        cout<<"Kode Unik              : "<<new_node-

>kodeunik<<endl;

        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                : "<<new_node->from<<endl;
        cout<<"Waktu                : "<<new_node-

>waktu<<endl;

        cout<<"Harga Beras          : ";cin>>new_node-
>hbm;cout<<endl;

        ///////////////////////////////////OPERASI FILE/////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->bm<<" ";
        infile<<new_node->gbm<<" ";
        infile<<new_node->kbm<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hbm<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else{
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) :

";cin>>pil3;

        if(pil3 == 'y' || pil3 == 'Y'){

```

```

        menus();
    }
    if(pil3 == 'T' || pil3 == 't'){
        exit(0);
    }
}

new_node->next = ghead2;
new_node->prev = NULL;

if (ghead2 != NULL)
    ghead2->prev = new_node;

ghead2 = new_node;
}
else{
    pilih2 input;
    cout<<"||==Stack yang anda pilih sudah
penuh==||"<<endl;

    menginputkan ?"<<endl;

    cout<<"Apa anda ingin kembali ke menu atau tetap

    cout<<"Menu/Input - (M/I) : ";cin>>input;
    if(input == 'M' || input == 'm'){
        menus();
    }
    else if(input == 'I' || input == 'i'){
        masuk(pil);
    }
}

}

}
if(pil3 == 't' || pil3 == 'T'){
    system("cls");
    menus();
}
}

```

//untuk stack(no) begitu prosedur/fungsi yang dibawahnya

```

void datas::s1(int pil, int pil2){
    struct stack1*new_node = (struct stack1*)malloc(sizeof(struct stack1));
    new_node = new(stack1);
    ofstream infile;
    infile.open("stack1.txt",ios::app);
    long long int uang;
    ifstream in("saldo.txt",ios::in);
    in>>uang;
    in.close();
    cout<<endl;
    if(getCount(pil2) < 11){
        if(pil == 1 && uang >= 120000){
            long long int harga = 120000;
            int hasil = uang-harga;
            ofstream outsaldo;

```

```

        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->c4<<endl;
        cout<<"Kualitas                    : "<<(new_node->gc4<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kc4<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                        : "<<(new_node->waktu<<endl;
        cout<<"Harga Beras                  : ";cin>>new_node->hc4;cout<<endl;
        ///////////////////////////////////OPERASI FILE////////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->c4<<" ";
        infile<<(new_node->gc4<<" ";
        infile<<(new_node->kc4<<" ";
        infile<<(new_node->kodeunik<<" ";
        infile<<(new_node->hc4<<" ";
        infile<<(new_node->from<<" ";
        infile<<(new_node->waktu<<endl;
    }
    else if(pil == 2 && uang >= 115000){
        long long int harga = 115000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->d<<endl;
        cout<<"Kualitas                    : "<<(new_node->gd<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kd<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                        : "<<(new_node->waktu<<endl;
        cout<<"Harga Beras                  : ";cin>>new_node->hd;cout<<endl;
        ///////////////////////////////////OPERASI FILE////////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->d<<" ";
        infile<<(new_node->gd<<" ";
        infile<<(new_node->kd<<" ";
        infile<<(new_node->kodeunik<<" ";
        infile<<(new_node->hd<<" ";
        infile<<(new_node->from<<" ";
        infile<<(new_node->waktu<<endl;
    }
    else if(pil == 3 && uang >= 160000){
        long long int harga = 160000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);

```

```

        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<new_node->bm<<endl;
        cout<<"Kualitas                    : "<<new_node->gbm<<endl;
        cout<<"Kode Beras                  : "<<new_node->kbm<<endl;
        cout<<"Kode Unik                    : "<<new_node->kodeunik<<endl;
        cout<<"Harga Distributor : "<<harga<<endl;
        cout<<"Dari                        : "<<new_node->from<<endl;
        cout<<"Waktu                        : "<<new_node->waktu<<endl;
        cout<<"Harga Beras                  : ";cin>>new_node->hbm;cout<<endl;
        ///////////////////////////////////OPERASI FILE////////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->bm<<" ";
        infile<<new_node->gbm<<" ";
        infile<<new_node->kbm<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hbm<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else{
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) : ";cin>>pil3;
        if(pil3 == 'y' || pil3 == 'Y'){
            menus();
        }
        if(pil3 == 'T' || pil3 == 't'){
            exit(0);
        }
    }
}
else{
    gs(pil, pil2);
}

new_node->next = head1;
new_node->prev = NULL;

if (head1 != NULL)
    head1->prev = new_node;

head1 = new_node;
}

//STACK 2
void datas::s2(int pil, int pil2){
    struct stack2*new_node = (struct stack2*)malloc(sizeof(struct stack2));
    new_node = new(stack2);
    ofstream infile;
    infile.open("stack2.txt",ios::app);
    long long int uang;

```

```

ifstream in("saldo.txt",ios::in);
in>>uang;
in.close();
cout<<endl;
if(getCount(pil2) < 11){
    if(pil == 1 && uang >= 120000){
        long long int harga = 120000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->c4<<endl;
        cout<<"Kualitas                    : "<<(new_node->gc4<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kc4<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                      : "<<(new_node->>waktu<<endl;
        cout<<"Harga Beras                    : ";cin>>new_node->hc4;cout<<endl;
        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->c4<<" ";
        infile<<(new_node->gc4<<" ";
        infile<<(new_node->kc4<<" ";
        infile<<(new_node->kodeunik<<" ";
        infile<<(new_node->hc4<<" ";
        infile<<(new_node->from<<" ";
        infile<<(new_node->>waktu<<endl;
    }
    else if(pil == 2 && uang >= 115000){
        long long int harga = 115000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->d<<endl;
        cout<<"Kualitas                    : "<<(new_node->gd<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kd<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                      : "<<(new_node->>waktu<<endl;
        cout<<"Harga Beras                    : ";cin>>new_node->hd;
        cout<<endl;
        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->d<<" ";
        infile<<(new_node->gd<<" ";
        infile<<(new_node->kd<<" ";
    }
}

```

```

        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hd<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else if(pil == 3 && uang >= 160000){
        long long int harga = 160000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<new_node->bm<<endl;
        cout<<"Kualitas                    : "<<new_node->gbm<<endl;
        cout<<"Kode Beras                  : "<<new_node->kbm<<endl;
        cout<<"Kode Unik                    : "<<new_node->kodeunik<<endl;
        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                        : "<<new_node->from<<endl;
        cout<<"Waktu                        : "<<new_node->waktu<<endl;
        cout<<"Harga Beras                    : ";cin>>new_node->hbm;
        cout<<endl;
        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->bm<<" ";
        infile<<new_node->gbm<<" ";
        infile<<new_node->kbm<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hbm<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else{
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) : ";cin>>pil3;
        if(pil3 == 'y' || pil3 == 'Y'){
            menus();
        }
        if(pil3 == 'T' || pil3 == 't'){
            exit(0);
        }
    }
}
else{
    gs(pil,pil2);
}

new_node->next = head2;
new_node->prev = NULL;

if (head2 != NULL)
    head2->prev = new_node;

```

```

        head2 = new_node;
    }

//STACK 3
void datas::s3(int pil, int pil2){
    struct stack3*new_node = (struct stack3*)malloc(sizeof(struct stack3));
    new_node = new(stack3);
    ofstream infile;
    infile.open("stack3.txt",ios::app);
    long long int uang;
    ifstream in("saldo.txt",ios::in);
    in>>uang;
    in.close();
    cout<<endl;
    if(getCount(pil2) < 11){
        if(pil == 1 && uang >= 120000){
            long long int harga = 120000;
            int hasil = uang-harga;
            ofstream outsaldo;
            outsaldo.open("Saldo.txt", ios::out);
            outsaldo<<hasil;
            outsaldo.close();
            cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
            cout<<"Jenis Beras                : "<<new_node->c4<<endl;
            cout<<"Kualitas                    : "<<new_node->gc4<<endl;
            cout<<"Kode Beras                  : "<<new_node->kc4<<endl;
            cout<<"Kode Unik                    : "<<new_node->kodeunik<<endl;
            cout<<"Harga Distributor : "<<harga<<endl;
            cout<<"Dari                        : "<<new_node->from<<endl;
            cout<<"Waktu                      : "<<new_node->waktu<<endl;
            cout<<"Harga Beras                  : ";cin>>new_node->hc4;
            cout<<endl;
            ////////////OPERASI FILE//////////
            infile<<(new_node->nomor = getCount(pil2))<<" ";
            infile<<new_node->c4<<" ";
            infile<<new_node->gc4<<" ";
            infile<<new_node->kc4<<" ";
            infile<<new_node->kodeunik<<" ";
            infile<<new_node->hc4<<" ";
            infile<<new_node->from<<" ";
            infile<<new_node->waktu<<endl;
        }
        else if(pil == 2 && uang >= 115000){
            long long int harga = 115000;
            int hasil = uang-harga;
            ofstream outsaldo;
            outsaldo.open("Saldo.txt", ios::out);
            outsaldo<<hasil;
            outsaldo.close();
            cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
            cout<<"Jenis Beras                : "<<new_node->d<<endl;
            cout<<"Kualitas                    : "<<new_node->gd<<endl;

```



```

        cout<<"Kode Beras          : "<<new_node->kd<<endl;
        cout<<"Kode Unik          : "<<new_node->kodeunik<<endl;
        cout<<"Harga Distributor : "<<harga<<endl;
        cout<<"Dari              : "<<new_node->from<<endl;
        cout<<"Waktu              : "<<new_node->waktu<<endl;
        cout<<"Harga Beras        : ";cin>>new_node->hd;
        cout<<endl;
        ///////////////////////////////////OPERASI FILE/////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->d<<" ";
        infile<<new_node->gd<<" ";
        infile<<new_node->kd<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hd<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else if(pil == 3 && uang >= 160000){
        long long int harga = 160000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                  : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras          : "<<new_node->bm<<endl;
        cout<<"Kualitas            : "<<new_node->gbm<<endl;
        cout<<"Kode Beras          : "<<new_node->kbm<<endl;
        cout<<"Kode Unik          : "<<new_node->kodeunik<<endl;
        cout<<"Harga Distributor : "<<harga<<endl;
        cout<<"Dari              : "<<new_node->from<<endl;
        cout<<"Waktu              : "<<new_node->waktu<<endl;
        cout<<"Harga Beras        : ";cin>>new_node->hbm;
        cout<<endl;
        ///////////////////////////////////OPERASI FILE/////////////////////////////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<new_node->bm<<" ";
        infile<<new_node->gbm<<" ";
        infile<<new_node->kbm<<" ";
        infile<<new_node->kodeunik<<" ";
        infile<<new_node->hbm<<" ";
        infile<<new_node->from<<" ";
        infile<<new_node->waktu<<endl;
    }
    else{
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) : ";cin>>pil3;
        if(pil3 == 'y' || pil3 == 'Y'){
            menus();
        }
        if(pil3 == 'T' || pil3 == 't'){
            exit(0);
        }
    }
}

```

```

        }
    }
}
else{
    gs(pil,pil2);
}

new_node->next = head3;
new_node->prev = NULL;

if (head3 != NULL)
    head3->prev = new_node;

head3 = new_node;
}

```

//STACK 4

```

void datas::s4(int pil, int pil2){
    struct stack4*new_node = (struct stack4*)malloc(sizeof(struct stack4));
    new_node = new(stack4);
    ofstream infile;
    infile.open("stack4.txt",ios::app);
    long long int uang;
    ifstream in("saldo.txt",ios::in);
    in>>uang;
    in.close();
    cout<<endl;
    if(getCount(pil2) < 11){
        if(pil == 1 && uang >= 120000){
            long long int harga = 120000;
            int hasil = uang-harga;
            ofstream outsaldo;
            outsaldo.open("Saldo.txt", ios::out);
            outsaldo<<hasil;
            outsaldo.close();
            cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
            cout<<"Jenis Beras                : "<<new_node->c4<<endl;
            cout<<"Kualitas                    : "<<new_node->gc4<<endl;
            cout<<"Kode Beras                    : "<<new_node->kc4<<endl;
            cout<<"Kode Unik                    : "<<new_node->kodeunik<<endl;
            cout<<"Harga Distributor : "<<harga<<endl;
            cout<<"Dari                        : "<<new_node->from<<endl;
            cout<<"Waktu                        : "<<new_node->waktu<<endl;
            cout<<"Harga Beras                    : ";cin>>new_node->hc4;
            cout<<endl;
            ////////////OPERASI FILE//////////
            infile<<(new_node->nomor = getCount(pil2))<<" ";
            infile<<new_node->c4<<" ";
            infile<<new_node->gc4<<" ";
            infile<<new_node->kc4<<" ";
            infile<<new_node->kodeunik<<" ";
            infile<<new_node->hc4<<" ";
            infile<<new_node->from<<" ";
            infile<<new_node->waktu<<endl;
        }
    }
}

```

```

    }
    else if(pil == 2 && uang >= 115000){
        long long int harga = 115000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->d<<endl;
        cout<<"Kualitas                    : "<<(new_node->gd<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kd<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                      : "<<(new_node->waktu<<endl;
        cout<<"Harga Beras                  : ";cin>>new_node->hd;
        cout<<endl;
        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->d<<" ";
        infile<<(new_node->gd<<" ";
        infile<<(new_node->kd<<" ";
        infile<<(new_node->kodeunik<<" ";
        infile<<(new_node->hd<<" ";
        infile<<(new_node->from<<" ";
        infile<<(new_node->waktu<<endl;
    }
    else if(pil == 3 && uang >= 160000){
        long long int harga = 160000;
        int hasil = uang-harga;
        ofstream outsaldo;
        outsaldo.open("Saldo.txt", ios::out);
        outsaldo<<hasil;
        outsaldo.close();
        cout<<"No.                : "<<(new_node->nomor =
getCount(pil2))<<endl;
        cout<<"Jenis Beras                : "<<(new_node->bm<<endl;
        cout<<"Kualitas                    : "<<(new_node->gbm<<endl;
        cout<<"Kode Beras                  : "<<(new_node->kbm<<endl;
        cout<<"Kode Unik                    : "<<(new_node->kodeunik<<endl;
        cout<<"Harga Distributor  : "<<harga<<endl;
        cout<<"Dari                        : "<<(new_node->from<<endl;
        cout<<"Waktu                      : "<<(new_node->waktu<<endl;
        cout<<"Harga Beras                  : ";cin>>new_node->hbm;
        cout<<endl;
        ////////////OPERASI FILE//////////
        infile<<(new_node->nomor = getCount(pil2))<<" ";
        infile<<(new_node->bm<<" ";
        infile<<(new_node->gbm<<" ";
        infile<<(new_node->kbm<<" ";
        infile<<(new_node->kodeunik<<" ";
        infile<<(new_node->hbm<<" ";
        infile<<(new_node->from<<" ";

```

```

        infile<<new_node->waktu<<endl;
    }
    else{
        char pil3;
        cout<<endl;
        cout<<"||==MAAF UANG ANDA TIDAK CUKUP==||"<<endl;
        cout<<"apa anda ingin kembali ke menu ? (Y/T) : ";cin>>pil3;
        if(pil3 == 'y' || pil3 == 'Y'){
            menus();
        }
        if(pil3 == 'T' || pil3 == 't'){
            exit(0);
        }
    }
}

else{
    gs(pil,pil2);
}

new_node->next = head4;
new_node->prev = NULL;

if (head4 != NULL)
    head4->prev = new_node;

head4 = new_node;
}

```

```

//Menu masuk (untuk memilih tumpukan yang ada
void datas::masuk(int pil){
    system("cls");
    viewmasuk: //fungsi goto (agar bisa redirect/kembali
    pilih pil2;
    cout<<"||=====Di toko ada 4 tumpukan=====||"<<endl;
    cout<<"Masukkan ke tumpukan (1-4) : ";cin>>pil2;
    if(pil2 == 1){
        s1(pil, pil2);
    }

    else if(pil2 == 2){
        s2(pil, pil2);
    }

    else if(pil2 == 3){
        s3(pil, pil2);
    }

    else if(pil2 == 4){
        s4(pil,pil2);
    }

    else{
        //jika pilihan tidak tepat
        system("cls");
    }
}

```

```

        cout<<"||===Maaf pilihan anda tidak tepat===||"<<endl;
        cout<<"||=====Silahkan Coba Lagi=====||"<<endl;
        cout<<endl;
        goto viewmasuk;
    }

    system("cls");
    menus();
}

//MENU UTAMA PROGRAM
void datas::menus(){
    system("cls");
    intomenu:
    pilih pil; //GUNAKAN FUNGSI TIPE DATA ABSTACT (ADT) yang dideklarasikan di header
file
    cout<<"=====MENU===== "<<endl;
    cout<<"1.Beli dari distributor"<<endl;
    cout<<"2.Saldo"<<endl;
    cout<<"3.Tampilkan"<<endl;
    cout<<"4.Jual Beras"<<endl;
    cout<<"===== "<<endl;
    cout<<"Masukkan Pilihan : ";cin>>pil;
    if(pil == 1){
        distrib();
    }
    else if(pil == 2){
        saldo();
    }
    else if(pil == 3){
        tampilkan();
    }
    else if(pil == 4){
        beliberas();
    }
    else{
        system("cls");
        cout<<"===Pilihan pilihan menu tidak ada==="<<endl;
        cout<<"=====Mohon masukkan kembali===== "<<endl<<endl;
        goto intomenu;
    }
    cout<<endl;
}

//PROSEDUR/FUNGSI PENGISIAN SALDO
void datas::saldo(){
    pilih2 pil; //(ADT)
    long long int input;
    long long int output;
    ///insaldo
    atop:
    ifstream insaldo("Saldo.txt"); //panggil saldo untuk ifstream
    insaldo>>output; //masukkan text yang didalam saldo ke output
    insaldo.close();
    /////

```

```

system("cls");
cout<<"Saldo anda saat ini adalah : Rp."<<output<<endl; // cek saldo
cout<<"Apa anda ingin mengisi saldo saat ini ? (Y/T) : ";cin>>pil;
if(pil == 'y' || pil == 'Y'){ // apabila ya maka
    cout<<"Masukkan saldo yang akan di tambahkan : ";cin>>input;
    int hasil = input+output; // input saldo dan kalkulasikan dengan output
    ofstream outsaldo;
    outsaldo.open("Saldo.txt", ios::out);
    outsaldo<<hasil; //masukkan hasil tadi ke saldo kembali
    outsaldo.close();
    goto atap; // kembali ke atap (redirect atap)
}
if(pil == 't' || pil == 'T'){ //apabila tidak maka kembali ke menu
    system("cls");
    menus();
}
}

//Sub menu untuk pembelian beras
void datas::distrib(){
    system("cls");
    pilih pil; //(ADT)
    //Pilih beras yang ingin dipilih
    cout<<"=====DISTRIBUTOR===== "<<endl;
    cout<<"1.Beras C4 (Kualitas Super)"<<endl;
    cout<<"2.Beras Delanggu (Biasa)"<<endl;
    cout<<"3.Beras Merah"<<endl;
    cout<<"Masukkan pilihan : ";cin>>pil;
    masuk(pil);
}

main(){
    datas read; //AMBIL CLASS YANG DI HEADER TADI
    read.read(); // BACA TERLEBIH DAHULU FILE TXT SAAT MEMBUKA PROGRAM
    read.menus(); //Deklarasikan Menunya
}

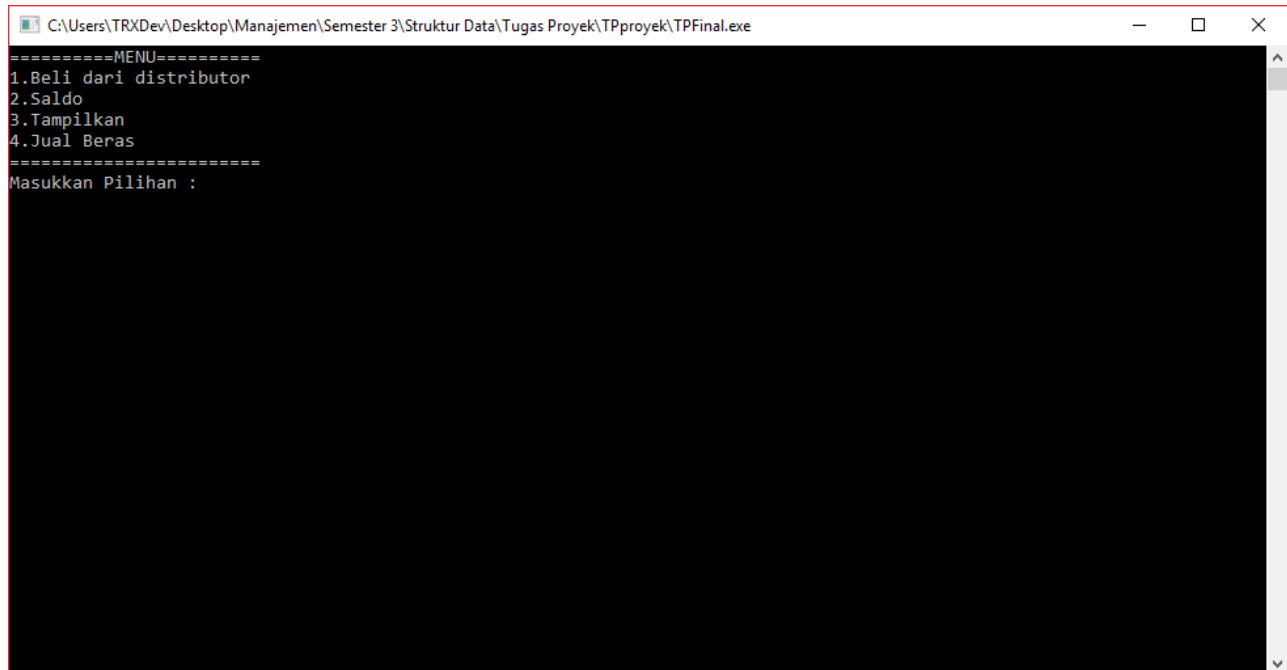
```

---

END OF TPAkhir.cpp (Body)

## C.Uji Coba Dan Dokumentasi Program

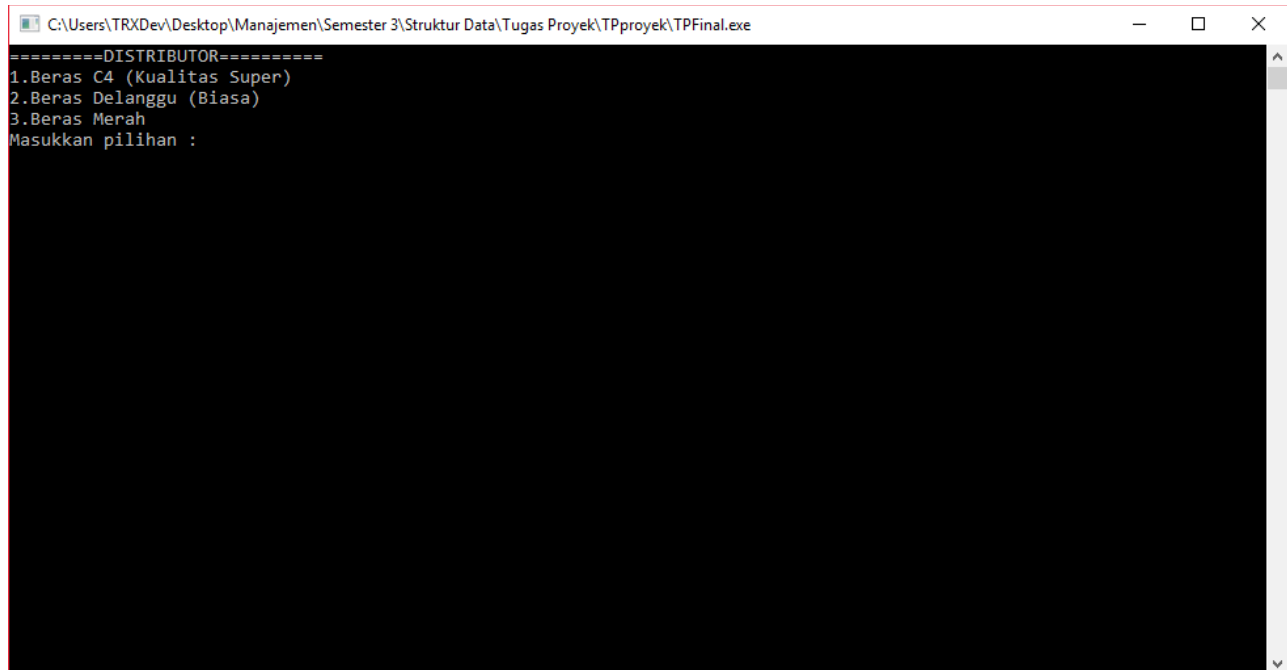
- Menu Awal :



A screenshot of a Windows application window titled "C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe". The window contains a black console area with white text. The text displays a menu with four options: "1.Beli dari distributor", "2.Saldo", "3.Tampilkan", and "4.Jual Beras". The menu is enclosed in a box of equals signs. Below the menu, it says "Masukkan Pilihan :".

```
=====MENU=====
1.Beli dari distributor
2.Saldo
3.Tampilkan
4.Jual Beras
=====
Masukkan Pilihan :
```

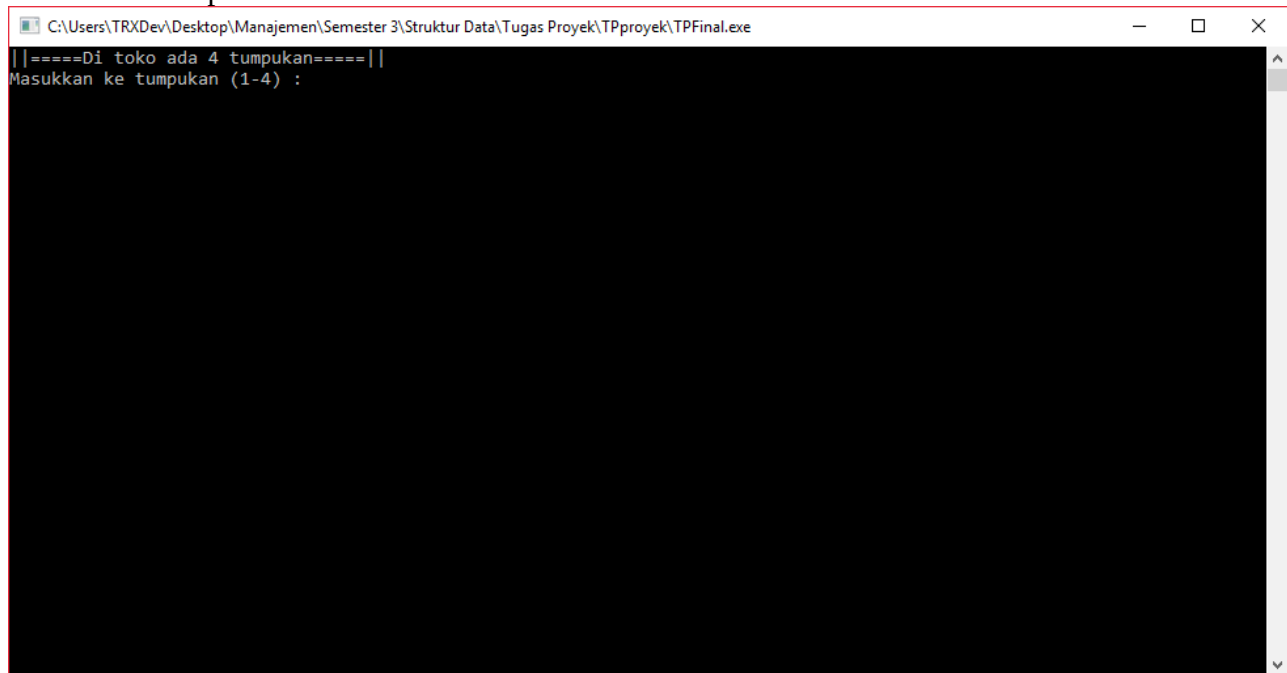
- Menu Beli :



A screenshot of the same Windows application window, now showing the "Menu Beli" screen. The text displays a list of three rice types: "1.Beras C4 (Kualitas Super)", "2.Beras Delanggu (Biasa)", and "3.Beras Merah". This list is also enclosed in a box of equals signs. Below the list, it says "Masukkan pilihan :".

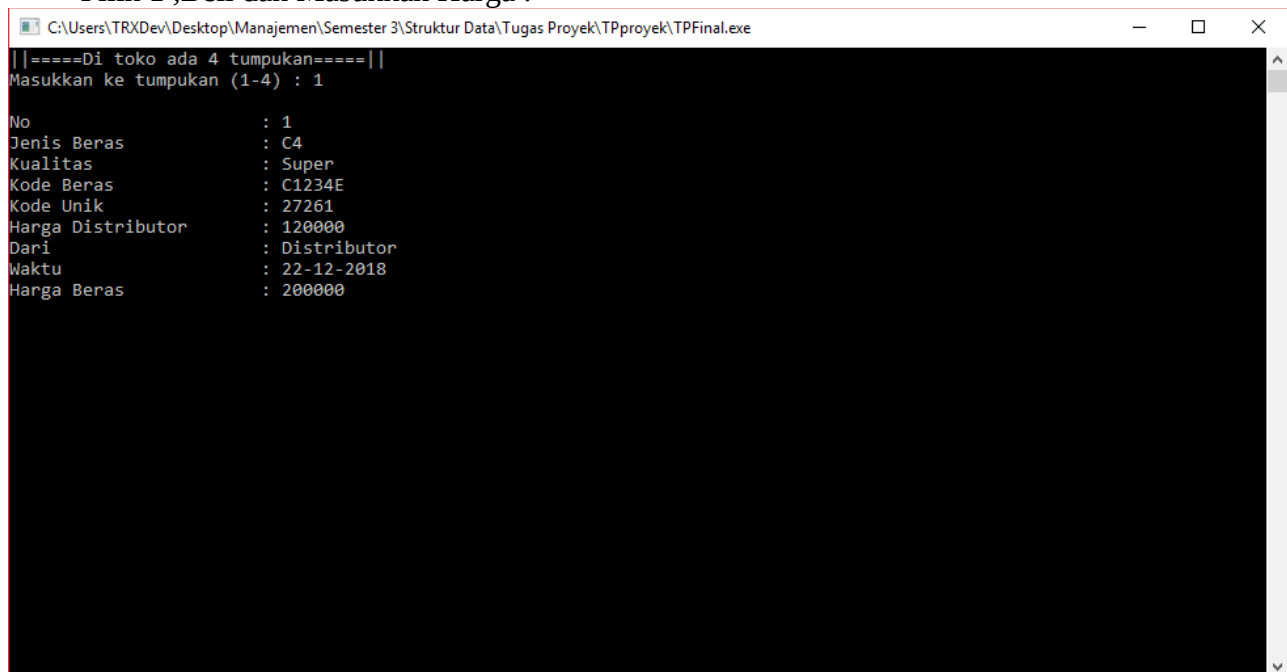
```
=====DISTRIBUTOR=====
1.Beras C4 (Kualitas Super)
2.Beras Delanggu (Biasa)
3.Beras Merah
Masukkan pilihan :
```

- Pilih Tumpukan :



```
C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||=====Di toko ada 4 tumpukan=====||
Masukkan ke tumpukan (1-4) :
```

- Pilih 1 ,Beli dan Masukkan Harga :

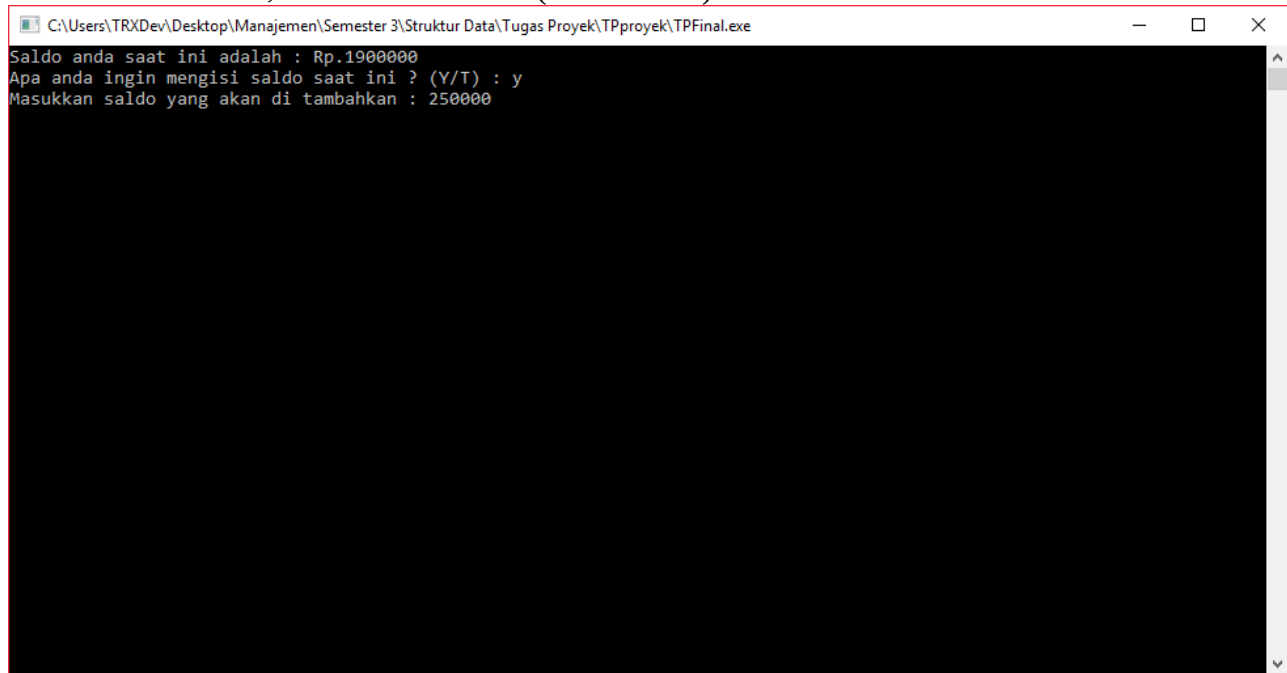


```
C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||=====Di toko ada 4 tumpukan=====||
Masukkan ke tumpukan (1-4) : 1

No                : 1
Jenis Beras       : C4
Kualitas          : Super
Kode Beras        : C1234E
Kode Unik         : 27261
Harga Distributor : 120000
Dari              : Distributor
Waktu             : 22-12-2018
Harga Beras       : 200000
```



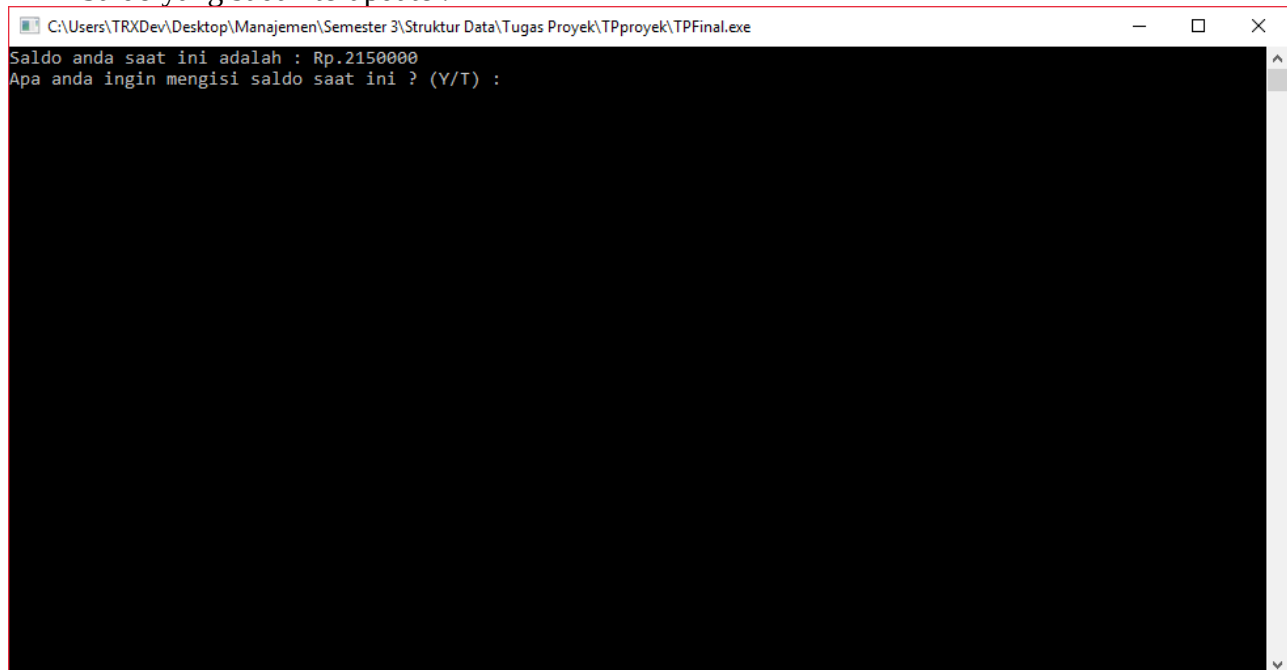
- Balik ke menu, Pilih saldo dan isi : (Menu Saldo)



A screenshot of a Windows command prompt window titled "C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe". The window has a black background with white text. The text displayed is as follows:

```
Saldo anda saat ini adalah : Rp.1900000
Apa anda ingin mengisi saldo saat ini ? (Y/T) : y
Masukkan saldo yang akan di tambahkan : 250000
```

- Saldo yang sudah terupdate :



A screenshot of a Windows command prompt window titled "C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe". The window has a black background with white text. The text displayed is as follows:

```
Saldo anda saat ini adalah : Rp.2150000
Apa anda ingin mengisi saldo saat ini ? (Y/T) :
```

- Menu Tampilan :

```

C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||=====TAMPILKAN=====||
1.Data Tumpukan 1 (Toko)
2.Data Tumpukan 2 (Toko)
3.Data Tumpukan 3 (Toko)
4.Data Tumpukan 4 (Toko)
5.Data Tumpukan 1 (Gudang)
6.Data Tumpukan 2 (Gudang)

Masukkan Pilihan :

```

- Menu Tampilan (misal pilih 1) :

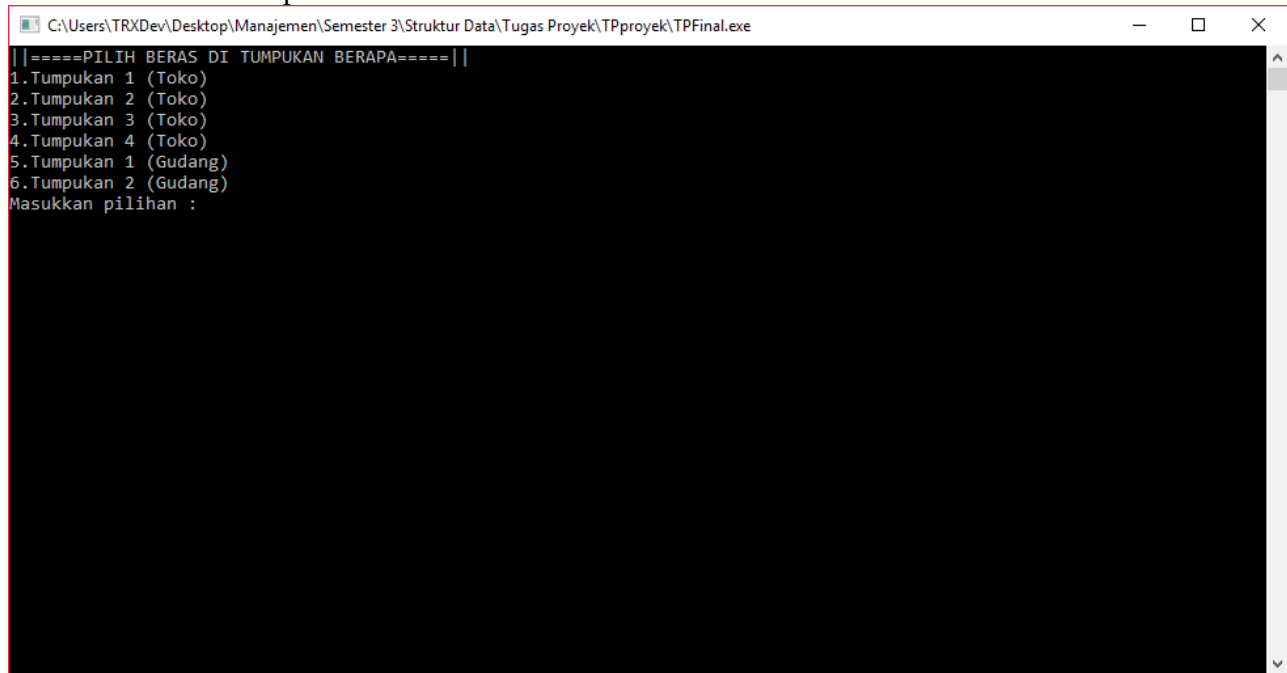
```

C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||=====TAMPILKAN=====||
1.Data Tumpukan 1 (Toko)
2.Data Tumpukan 2 (Toko)
3.Data Tumpukan 3 (Toko)
4.Data Tumpukan 4 (Toko)
5.Data Tumpukan 1 (Gudang)
6.Data Tumpukan 2 (Gudang)

Masukkan Pilihan : 1
=====
| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga | Dari | DD-MM-YYYY |
=====
| 3 | Delanggu | Standart | D23HJA | 27992 | 300000 | Distributor | 22-12-2018 |
| 2 | C4 | Super | C1234E | 27970 | 250000 | Distributor | 22-12-2018 |
| 1 | C4 | Super | C1234E | 27261 | 200000 | Distributor | 22-12-2018 |
=====
Jumlah Semua : 3
Apa anda ingin kembali ke menu ? (Y/T) :

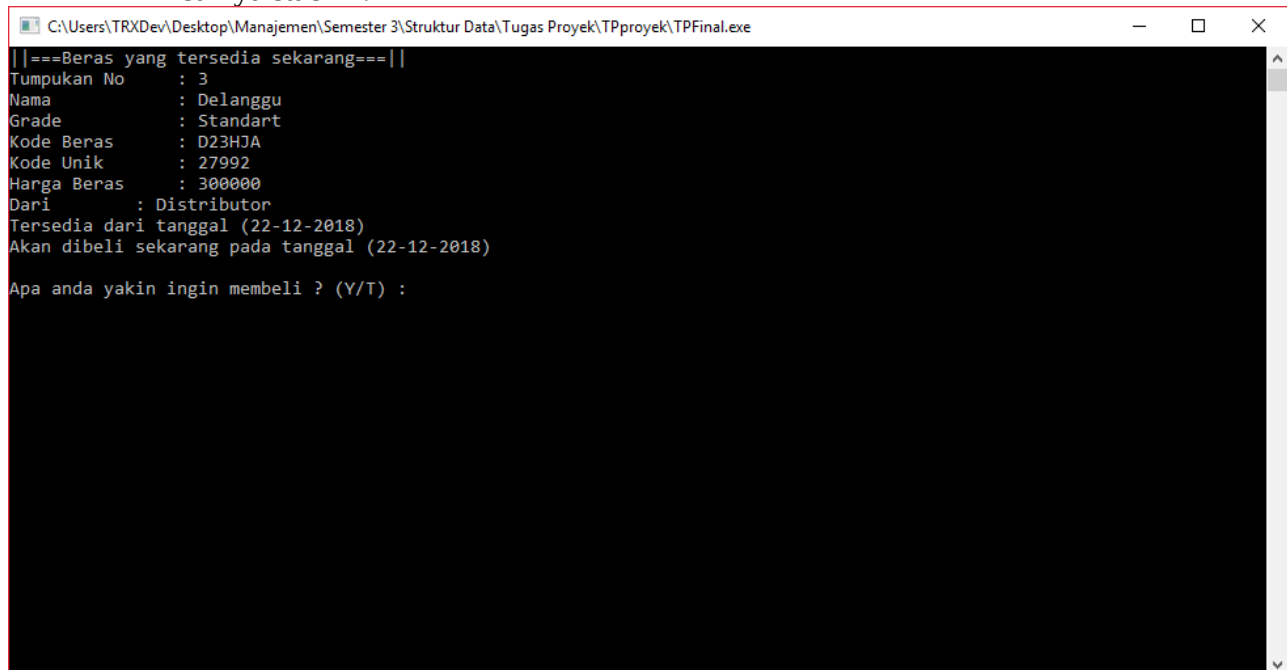
```

- Menu Beli Tumpukan :



```
C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||====PILIH BERAS DI TUMPUKAN BERAPA====||
1.Tumpukan 1 (Toko)
2.Tumpukan 2 (Toko)
3.Tumpukan 3 (Toko)
4.Tumpukan 4 (Toko)
5.Tumpukan 1 (Gudang)
6.Tumpukan 2 (Gudang)
Masukkan pilihan :
```

- Pilih misalnya stack1 :



```
C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||===Beras yang tersedia sekarang===||
Tumpukan No      : 3
Nama             : Delanggu
Grade            : Standart
Kode Beras       : D23HJA
Kode Unik        : 27992
Harga Beras      : 300000
Dari             : Distributor
Tersedia dari tanggal (22-12-2018)
Akan dibeli sekarang pada tanggal (22-12-2018)
Apa anda yakin ingin membeli ? (Y/T) :
```

- Beli yang ada di stack1 :

```

C:\Users\TRXDev\Desktop\Manajemen\Semester 3\Struktur Data\Tugas Proyek\TPproyek\TPFinal.exe
||===Beras yang tersedia sekarang===||
Tumpukan No      : 3
Nama             : Delanggu
Grade            : Standart
Kode Beras       : D23HJA
Kode Unik        : 27992
Harga Beras      : 300000
Dari             : Distributor
Tersedia dari tanggal (22-12-2018)
Akan dibeli sekarang pada tanggal (22-12-2018)

Apa anda yakin ingin membeli ? (Y/T) : y
===Beras sudah dibeli===

=====
| No. | Jenis Beras | Grade | Kode Beras | Kode Unik | Harga | Dari | DD-MM-YYYY |
=====
| 2 | C4 | Super | C1234E | 27970 | 250000 | Distributor | 22-12-2018 |
| 1 | C4 | Super | C1234E | 27261 | 200000 | Distributor | 22-12-2018 |
=====
Jumlah Semua : 2
Apa anda ingin kembali ke menu ? (Y/T) :

```

## BAB 3

### Kesimpulan

#### A.Kesimpulan Isi

Kesimpulannya pada program ini adalah untuk membantu pemilik beras/pengguna agar bisa manage stack/tumpukan berasnya agar lebih baik, lebih cepat, dan optimal serta membuat proses jual beli jadi lebih mudah pendataannya dengan menggunakan program ini. Pada program ini sangatlah sederhana namun setidaknya ini cukup berguna bagi pemilik/pengguna.

Kami berharap program ini bisa bermanfaat bagi semuanya yang ingin menggunakannya (dalam manage stack dan proses jual beli dalamnya).

#### B.Kendala yang ada

Sejujurnya kendala yang ada hanyalah pemilik merupakan orang yang sibuk sehingga sedikit susah mendapatkan informasinya namun syukur kami masih bisa menanyakan setidaknya sedikit hal-hal penting dan juga diperbolehkan dalam mengambil gambar yang ada di toko terutama pada tumpukan beras.

Selain itu kesusahan ada pada programnya yang ada, program ini berisikan setidaknya 2300+ baris, yang didalamnya banyak setidaknya spam. Walaupun banyak spam, sesungguhnya spam ini diperlukan contohnya dengan membuat setiap variabel yang sama pada setiap stacknya dan semuanya bersifat individual, semisal artinya jika 1 stack membutuhkan setidaknya 800 kode untuk komplit maka didalam cpp kami ada 6 stack maka  $800 \times 6$ , bisa dibayangkan banyaknya, namun walaupun seperti itu semuanya berjalan dengan semestinya.

## BAB 4

### Penutup



Alhamdulillah tugasnya sudah selesai dan semuanya baik baik saja, mulai dari tahap awal mencari tempat untuk diteliti sampai programnya jadi hingga presentasi sampai ke makalah ini. Semoga ini bisa menjadi pelajaran dan pengalaman bagi kami dan semoga ini bisa bermanfaat bagi orang banyak terutama pada aplikasinya.

Jika kami ada salah kami mohon maaf, akhir kata kami ucapkan Terima Kasih .

### Struktur Data