

**LAPORAN PRAKTIKUM**  
**SESI 1**  
**PRAKTIKUM COMP6362 – DATA STRUCTURES**  
**KELAS BE20**



Oleh :  
2440008600 – Andru Baskara Putra

**SEMESTER GENAP 2020/2021**  
**BINA NUSANTARA UNIVERSITY**  
**MALANG**

## A. Kode Program

### Source Code

```
#include <stdio.h>
#include <string.h>
#include <stdlib.h>

int i;

typedef struct chip{
    char name[200];
    char code[200];
    int product;
    int price;
}chip; chip count[7];

void store(){
    strcpy(count[1].code, "CP001");
    strcpy(count[1].name, "Banana Chips");
    count[1].product=25;
    count[1].price=10000;

    strcpy(count[2].code, "CP005");
    strcpy(count[2].name, "Apple Chips");
    count[2].product = 45;
    count[2].price = 12000;

    strcpy(count[3].code, "CP009");
    strcpy(count[3].name, "MeLon Chips");
    count[3].product = 32;
    count[3].price = 14000;

    strcpy(count[4].code, "CP014");
    strcpy(count[4].name, "Jackfruit Chips");
    count[4].product = 20;
    count[4].price = 15000;

    strcpy(count[5].code, "CP015");
    strcpy(count[5].name, "Pineapple Chips");
    count[5].product = 27;
    count[5].price = 16000;

    strcpy(count[6].code, "CP017");
    strcpy(count[6].name, "Dragon Fruit Chips");
    count[6].product = 29;
    count[6].price = 21000;

    strcpy(count[7].code, "CP025");
    strcpy(count[7].name, "Durian Chips");
```

```

        count[7].product = 14;
        count[7].price = 27500;
    }

    //untuk sementara saya gunakan binary
    /*int finding(struct chip count[],int find, int slave, int noble){
        int cent;
        while(slave<=noble){
            cent=(slave+noble)/2;

            if(strcmp(count[cent].code,find)==1){
                noble=cent-1;
            }else if(strcmp(count[cent].code,find)==-1){
                slave=cent+1;
            }else{
                return cent;
            }
            return -1;

        }

    }*/

    //pakai linier
    int finding(chip count[],int find,int arraysize){
        for ( int i = 1; i <= arraysize; ++i) {
            if (strcmp(count[i].code,find)==0)
                return i;
        }

        //ga ketemu
        return -1;
    }

    void line(){
        printf("_____
        _____\n");
    }

    void table(){
        line();
        printf("\n/No.      /      Code      /      Chips Name      /      Available      /
        Price      /\n");
    }

```

```

        line();
        printf("\n/01.    |    CP001    |    Banana Chips    |    25
/ Rp. 10.000,- | \n");
        line();
        printf("\n/02.    |    CP005    |    Apple Chips    |    45
/ Rp. 12.000,- | \n");
        line();
        printf("\n/03.    |    CP009    |    Melon Chips    |    32
/ Rp. 14.000,- | \n");
        line();
        printf("\n/04.    |    CP014    |    Jackfruit Chips    |    20
/ Rp. 15.000,- | \n");
        line();
        printf("\n/05.    |    CP015    |    Pineapple Chips    |    27
/ Rp. 16.000,- | \n");
        line();
        printf("\n/06.    |    CP017    |    Dragon Fruit Chips    |    29
/ Rp. 21.000,- | \n");
        line();
        printf("\n/07.    |    CP025    |    Durian Chips    |    14
/ Rp. 27.500,- | \n");
        line();
    }

int main()
{
    int a; int b;
    table();
    store();
    a:
    printf("\n===== \n");
    printf("                Welcome! \n");
    printf("\n===== \n");
    printf("1. Jual \n");
    printf("2. Tambah Stock \n");
    printf("3. Kill Program \n");
    printf("Mau Ngapain? \n");
    printf(">>");
    scanf("%d",&a);
    if(a<1 || a>3){
        printf("pilihan yang sangat tidak valid, coba diisi kembali 1-
3 pilihannya \n");
        goto a;
    }
    while(a != 3){
        if(a==1){
            char numcode[7];
            int numfinding,var,total,greattotal,n=0;

```

```

re:
printf("Silahkan Masukan Kode produk \n");
printf(">>"); scanf("%s",&numcode);
var=finding(count,numcode,7);
if(var==-1){
    printf("Maaf, kode yang anda masukan tidak tersedia, mohon masuk
an kembali dengan benar, lihat tabelnya\n");
    printf("Penggunaan capslock perlu di perhatikan\n");
    goto re;
}

ra:
printf("Mau berapa?\n");
printf(">>"); scanf("%d",&b);
if(b<0 || b>count[var].product){
    printf("Sepertinya jumlah yang anda masukan terlalu kurang ataup
un kelebihan, mohon diisi sesuai tabel\n");
    goto ra;
}

printf("Harga Total = %d * %d\n",count[var].price,b);

total= count[var].price*b;
printf("Harga Total = %d\n",total);
printf("====Mantab Laris Manis====\n");
printf("\n====\n");
printf("\n      Current Stock\n");
printf("\n====\n");
count[var].product -= b;
for (i=1; i<=7; i++){
    line();
    printf("/%d. ",i);
    printf(" / %s ",count[i].code);
    int panjangstring=strlen(count[i].name);
    if(panjangstring<8)
    {
        printf("/ %s\t\t");
    }else if(panjangstring>=8 && panjangstring<16)
    {
        printf("/ %s\t\t",count[i].name);
    }else if(panjangstring>=16 && panjangstring<24)
    {
        printf("/ %s\t",count[i].name);
    }else if(panjangstring>=24 && panjangstring<32)
    {
        printf("/ %s\t",count[i].name);
    }else if(panjangstring>=32 && panjangstring<40)
    {

```

```

        printf("/ %s\t",count[i].name);
    }

    printf("/ %d ",count[i].product);
    printf(" / Rp. %d,- \n",count[i].price);
    line();
}
goto a;

}else if(a==2){
    int itemfinding,updated,total,n=0;
    char numcode[7];
ro:
    printf("Masukan Kode produk\n");
    printf(">>"); scanf("%s",&numcode);
    itemfinding=finding(count,numcode,8);
    if(itemfinding==-1){
        printf("Maaf, kode yang anda masukan tidak tersedia, mohon masuk
an kembali dengan benar, Lihat tabelnya\n");
        printf("Penggunaan capslock perlu di perhatikan\n");
        goto ro;
    }

ru:
    printf("Mau Berapa?[1-100] \n");
    printf(">>");
    scanf("%d",&updated);
    if(updated<0 || updated>100){
        printf("Sepertinya jumlah yang anda masukan terlalu kurang ataupun
kelebihan, Value penambahan stock yaitu 1-
100 tidak kurang tidak lebih\n");
        goto ru;
    }
    printf("\n Berhasil!\n");
    printf("\n===== \n");
    printf("\n Current Stock\n");
    printf("\n===== \n");

    count[itemfinding].product += updated;

    for(i=1; i<=7; i++){
        line();
        printf("/%d. ",i);
        printf(" / %s ",count[i].code);
        int panjangstring=strlen(count[i].name);
        if(panjangstring<8)
        {
            printf("/ %s\t\t ");

```

```

    }else if(panjangstring>=8 && panjangstring<16)
    {
        printf("/ %s\t\t",count[i].name);
    }else if(panjangstring>=16 && panjangstring<24)
    {
        printf("/ %s\t",count[i].name);
    }else if(panjangstring>=24 && panjangstring<32)
    {
        printf("/ %s\t",count[i].name);
    }else if(panjangstring>=32 && panjangstring<40)
    {
        printf("/ %s\t",count[i].name);
    }

    printf("/ %d ",count[i].product);
    printf(" / Rp. %d,- \n",count[i].price);
    line();
}
goto a;
}

}
printf("\n===== \n");
printf("\n      Terimakasih telah \n");
printf("\n Menyaksikan saya presentasi \n");
printf("\n===== \n");
printf("      Bismillah Nilai Bagus \n");

exit(0);

return 0;
}

```

## B. Deskripsi Program

### 1. **typedef struct chip**

Didalam struct chip terdapat beberapa data yang dibutuhkan dalam program ini, Typdef digunakan agar nanti di proses selanjutnya tidak perlu declare kembali struct nya

### 2. **void store**

Berisikan data-data yang diisi sesuai dengan study case yang diperintahkan.

### 3. **int finding**

Berisikan proses dan perintah pencarian data menggunakan metode linier search, parameter yang digunakan adalah chip count[], int find dan int arraysize

### 4. **void line**

Berisikan output garis, untuk tampilan UI, diisi dalam fungsi agar program main terlihat rapi dan berisikan full logic.

### 5. **void table**

Berisikan output tampilan UI dari table daftar produk, diisi dalam fungsi agar program main terlihat rapi dan berisikan full logic

### 6. **int main / program utama / main program**

Bersisikan urutan proses dan perintah yang terjadi dalam program, berikut proses nya ;

1. Mendeklarasi variabel input a dan b
2. Lalu melakukan pemanggilan fungsi **void line** dan **void table** untuk tampilan UI
3. Membuat output berupa UI daftar menu utama, yaitu pilihan jual, tambah stock dan kill program
4. Input pilihan kedalam variabel a
5. Terapat proses kondisi if else, dimana jika variabel a memiliki value diantara 1-3, maka program akan melanjutkan perintah ke proses selanjutnya, namun jika tidak, program akan Kembali ke tampilan daftar menu awal menggunakan goto.
6. Program berjalan seperti semestinya dan bertemu dengan kondisi while, dimana selama value dari variabel a bukan bernilai 3, maka program akan menjalankan sebuah proses berdasarkan pilihan user, namun jika value a menyentuh 3, maka program akan menampilkan UI perpisahan dan langsung menghentikan program
7. Kondidi if else dimana jika user memilih pilihan pertama, maka program akan memberikan tampilan dan perintah input kode produk, jika kode produk salah maka program akan memberitahu user dan memerintahkan untuk mengulangi input Kembali, jika benar, maka program akan melakukan perintah penginputan jumlah produk yang akan dijual, Lalu program memberikan perhitungan **Harga x jumlah barang yang akan dijual** , lalu program meng output tampilan UI current stock yang berisikan table yang sudah di update dan juga total harga



8. Jika user memilih pilihan kedua maka program akan melakukan perintah input kode produk dan jumlah nya, jika salah maka akan diberi peringatan dan mengulang input seperti step 7, jika benar maka program akan melakukan perhitungan **Jumlah saat ini + Jumlah yang ditambahkan**. Lalu program akan meng output tampilan table yang telah di output

## C. Bukti Presentasi

