Aplikasi Kasir Sederhana
untuk Coffeeshop

LAPORAN STUDI KASUS TUGAS AOL
MATA KULIAH COMP6360004 – ALGORITHM & PROGRAMMING
KELAS LC20

DISUSUN OLEH :

2602172233 – RAYNALDY DWI KHARISMA

COMPUTER SCIENCE

SEMESTER GANJIL 2022/2023

MALANG

Daftar Isi

BAB I
PENDAHULUAN

1. LATAR BELAKANG

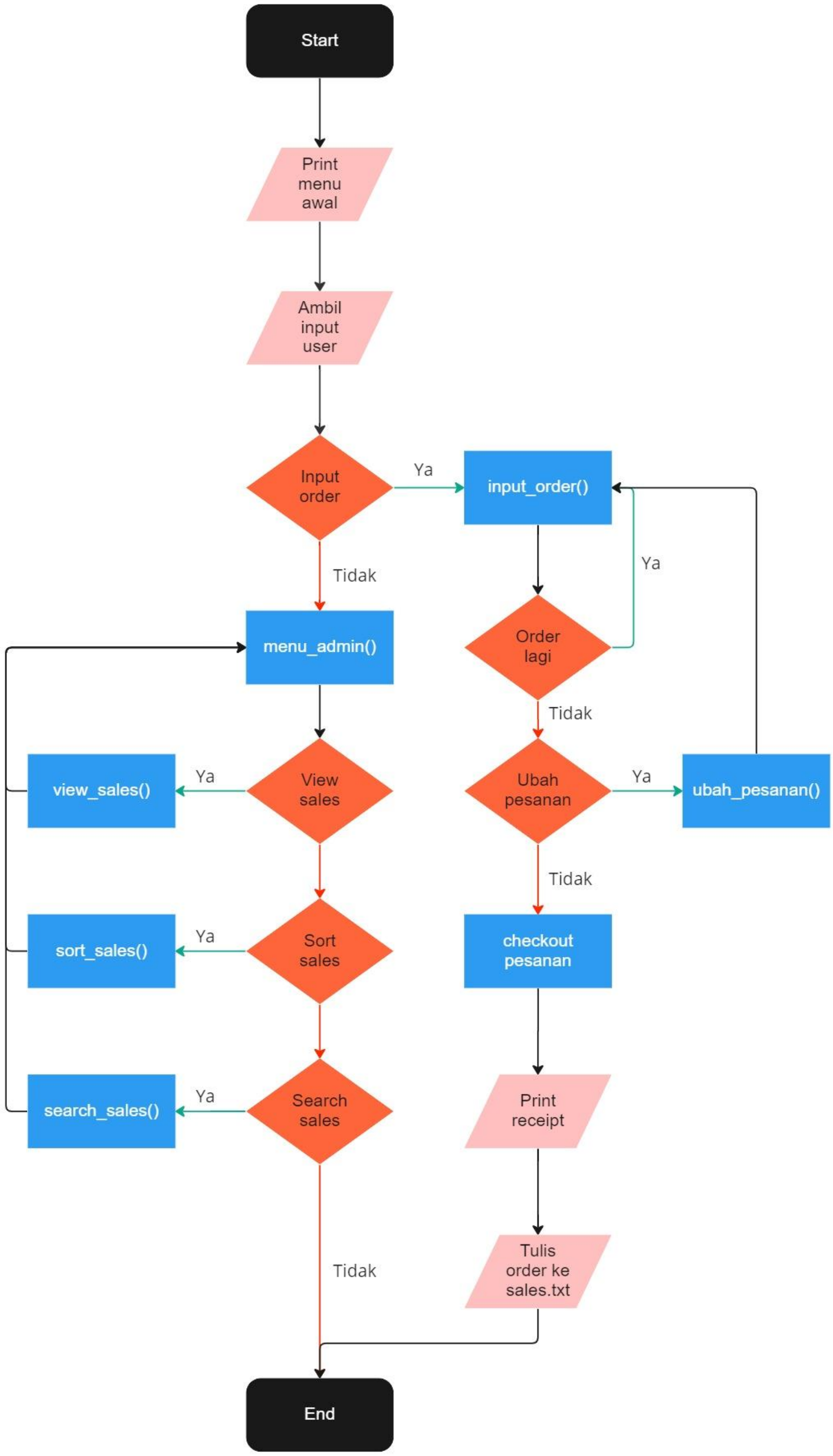
Di era yang semakin modern ini, saya sering kali melihat *coffeeshop* atau warung kopi yang masih manual dalam mencatat, merekap, dan memproses order pelanggan. Namun, sudah banyak juga *coffeeshop* yang sudah menggunakan mesin kasir otomatis yaitu aplikasi *Point of Sales* (POS). Hal ini membuat saya tertarik untuk mereplika aplikasi tersebut dalam skala kecil tetapi masih memiliki fungsi yang sama, yaitu otomatisasi order pelanggan.

2. KEBUTUHAN APLIKASI

Adapun beberapa hal yang saya butuhkan untuk membuat aplikasi tersebut:

1. Struktur kendali perulangan
    a. Do while loop
    b. For loop
2. Seleksi
    a. If else
    b. Switch case
3. Tipe data
    a. Empat buah struct untuk mencatat menu, order, sales, waktu, dan tanggal order
    b. Pointer bertipe const char untuk menyimpan format data read file
    c. Int, char, float, void untuk menyimpan sesuatu dalam variable
    d. Pointer FILE untuk membaca dan menulis file
4. Media penyimpanan
    a. File menu.txt untuk menyimpan menu *coffeeshop*
    b. File sales.txt untuk menyimpan sales *coffeeshop*
5. Algoritma sorting
    a. qsort *built-in* yang terdapat di library <stdlib.h>
    b. comparer nama – untuk sorting berdasarkan nama produk
    c. comparer qty – untuk sorting berdasarkan kuantitas produk terjual
    d. comparer total – untuk sorting berdasarkan total pembayaran
6. Algoritma searching
    a. Linear search menggunakan for loops dan if else statement untuk mencari data yang sesuai dengan kriteria

BAB II
DESAIN PROGRAM



BAB III
SOURCE CODE

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

const char *FORMAT_DATA_READ = "%d/%d/%d, %[^,], %c, %d, %d, %s\n";

struct Sales{
	int date[4];
	char nama[50];
	char size;
	int qty;
	int total_pembayaran;
	char metode_pembayaran[20];
} array_sales[100], temp;

struct Menu{
	char nama_menu[50];
	float harga_regular;
	float harga_large;
} array_menu[50];

struct Order{
	char date[20];
	int index_order;
	char ukuran_order;
	int qty_order;
} array_order[50];

void input_order();
void load_menu();
int index_counter(char FILE_DIR[]);
```

```
void print_menu_txt();
void scan_order(int index_temp);
void rekap_order(int index_temp);
void ubah_order();
void print_receipt(char sub_input3, int index_temp);
void write_to_file(char sub_input3, int index_temp);
void menu_admin();
void sort_sales();
void search_sales();
void print_sales();

int main(){
      char input;
      printf("Aplikasi Kasir\n");
      printf("--------------\n");
      printf("1. Input Order\n");
      printf("2. Admin\n");
      printf("0. Exit\n");
      printf("--------------\n");
      printf("Input: "); scanf("%d", &input); getchar();
      switch(input){
            case 1:
                  input_order();
                  break;
            case 2:
                  menu_admin();
                  break;
            case 0:
                  exit(0);
                  break;
            default:
                  printf("Input salah!\n");
                  system("pause");
                  system("cls");
                  main();
      }
      return 0;
}

void load_menu(){
      char buffer[256];
      FILE *fp;
      fp = fopen("menu.txt", "r");
      if(fp == NULL){
        printf("File menu.txt tidak ditemukan!\n");
    }
      fgets(buffer, sizeof(buffer), fp);
      int index=0;
    while(!feof(fp)){
      fscanf(fp, "%[^,], %f, %f\n",
                  &array_menu[index].nama_menu,
                  &array_menu[index].harga_regular,
                  &array_menu[index].harga_large);
            index++;
      }
      fclose(fp);
}

int index_counter(char FILE_DIR[]){
      FILE *fp;
    int count = 0;
    char c;
    fp = fopen(FILE_DIR, "r");
    if (fp == NULL){
        return 0;
    }
    for (c = getc(fp); c != EOF; c = getc(fp)){
      if (c == '\n'){
            count = count + 1;
            }
      }
    fclose(fp);
    return count;
}

void print_menu_txt(){
      int index = index_counter("menu.txt");
      printf("---------------------------------------------------\n");
      printf("| No |           Nama         | Regular |  Large  |\n");
      printf("---------------------------------------------------\n");
      for(int i=0; i<index; i++){
            printf("| %-2d | %-22s |  %.0f  |  %.0f  |\n",
                        i+1,
                        array_menu[i].nama_menu,
                        array_menu[i].harga_regular,
                        array_menu[i].harga_large);
      }
      printf("---------------------------------------------------\n");
}

void scan_order(int index_temp){
      printf("Pesan (No Produk): "); scanf("%d",
&array_order[index_temp].index_order); getchar();
      printf("Ukuran (R/L): "); scanf("%c",
&array_order[index_temp].ukuran_order); getchar();
      printf("Qty: "); scanf("%d", &array_order[index_temp].qty_order);
getchar();
}

void rekap_order(int index_temp){
      float total=0;
      printf("-------------------------------------------------------------
--\n");
      printf("| No |          Nama          | Ukuran | Harga | Qty | Total
|\n");
      printf("-------------------------------------------------------------
--\n");
      for(int i=0; i<index_temp; i++){
            if(array_order[i].ukuran_order == 'R'){
                  printf("| %-2d | %-22s |    %-3c | %.0f |  %-2d | %-6.0f
|\n",
                        i+1,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_menu[array_order[i].index_order-
1].harga_regular,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_regular * array_order[i].qty_order));
                        total+=(array_menu[array_order[i].index_order-
1].harga_regular * array_order[i].qty_order);
            }
            else if(array_order[i].ukuran_order == 'L'){
                  printf("| %-2d | %-22s |    %-3c | %.0f |  %-2d | %-6.0f
|\n",
                        i+1,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_menu[array_order[i].index_order-
1].harga_large,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_large * array_order[i].qty_order));
                        total+=(array_menu[array_order[i].index_order-
1].harga_large * array_order[i].qty_order);
            }
      }
      printf("------------------------------------------------------------
--\n");
      printf("\t\t\t\t    Total Pembayaran = %.0f\n", total);
}


void ubah_order(){
      int temp;
      printf("Pilih nomor pesanan yang ingin diubah!\n");
      printf("Input: "); scanf("%d", &temp); getchar();
      print_menu_txt();
      scan_order(temp-1);
}
```

```
void print_receipt(char sub_input3, int index_temp){
	printf("------------------------------------------------------------
--\n");
	printf("|\t\t\tIP Cafe 2\t\t\t      |\n");
	printf("------------------------------------------------------------
--\n");
	time_t t = time(NULL);
	struct tm tm = *localtime(&t);
	printf("| %d/%02d/%02d\t\t", tm.tm_mday, tm.tm_mon + 1, tm.tm_year +
1900);
	if(sub_input3 == '1'){
		printf("\t\t\t\t Cash |\n");
	}
	else if(sub_input3 == '2'){
		printf("\t\t\t\t QRIS |\n");
	}
	else if(sub_input3 == '3'){
		printf("\t\t\t     E-Wallet |\n");
	}
	printf("| %02d:%02d:%02d\t\t\t\t\t\t      |\n", tm.tm_hour,
tm.tm_min, tm.tm_sec);
	rekap_order(index_temp);
}


void write_to_file(char sub_input3, int index_temp){
	time_t t = time(NULL);
	struct tm tm = *localtime(&t);
	FILE *fp;
	fp = fopen("sales.txt", "a");
	for(int i=0; i<index_temp; i++){
		if(toupper(array_order[i].ukuran_order) == 'R' && sub_input3 ==
'1'){
			fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, Cash\n",
				tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
				array_menu[array_order[i].index_order-1].nama_menu,
				array_order[i].ukuran_order,
				array_order[i].qty_order,
				(array_menu[array_order[i].index_order-
1].harga_regular * array_order[i].qty_order));
```

```
            }
            else if(toupper(array_order[i].ukuran_order) == 'R' && 
sub_input3 =='2'){
                  fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, QRIS\n",
                        tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_regular * array_order[i].qty_order));
            }
            else if(toupper(array_order[i].ukuran_order) == 'R' && 
sub_input3 =='3'){
                  fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, E-Wallet\n",
                        tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_regular * array_order[i].qty_order));
            }
            else if(toupper(array_order[i].ukuran_order) == 'L' && 
sub_input3 == '1'){
                  fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, Cash\n",
                        tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_large * array_order[i].qty_order));
            }
            else if(toupper(array_order[i].ukuran_order) == 'L' && 
sub_input3 =='2'){
                  fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, QRIS\n",
                        tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_large * array_order[i].qty_order));
            }
            else if(toupper(array_order[i].ukuran_order) == 'L' &&
sub_input3 =='3'){
                  fprintf(fp, "%d/%d/%d, %s, %c, %d, %.0f, E-Wallet\n",
                        tm.tm_mday, tm.tm_mon + 1, tm.tm_year + 1900,
                        array_menu[array_order[i].index_order-1].nama_menu,
                        array_order[i].ukuran_order,
                        array_order[i].qty_order,
                        (array_menu[array_order[i].index_order-
1].harga_large * array_order[i].qty_order));
            }
      }
      fclose(fp);
}

void input_order(){
      char sub_input;
      char sub_input2;
      char sub_input3;
      int index_temp=0;
      do{
            system("cls");
            load_menu();
            print_menu_txt();
            scan_order(index_temp);
            index_temp++;
            printf("Pesan lagi? (Y/N)\n");
            printf("Input: "); scanf("%c", &sub_input); getchar();
      }while(toupper(sub_input)!='N');
      do{
            system("cls");
            rekap_order(index_temp);
            printf("\n");
            printf("1. Ubah pesanan\n");
            printf("2. Checkout\n");
```

```
        printf("Input: "); scanf("%c", &sub_input2); getchar();
        if(sub_input2 == '1'){
            ubah_order();
        }
    }while(sub_input2!='2');
    printf("Pilih metode pembayaran\n");
    printf("1. Cash\n");
    printf("2. QRIS\n");
    printf("3. E-Wallet\n");
    printf("Input: "); scanf("%c", &sub_input3); getchar();
    system("cls");
    print_receipt(sub_input3, index_temp);
    write_to_file(sub_input3, index_temp);
    system("pause");
    system("cls");
    main();
}

void print_array_sales(int i){
    printf("| %-2d | %d/%d/%d | %-22s |  %-3c | %-3d | %-6d | %-8s |\n",
        i+1,
        array_sales[i].date[0],
        array_sales[i].date[1],
        array_sales[i].date[2],
        array_sales[i].nama,
        array_sales[i].size,
        array_sales[i].qty,
        array_sales[i].total_pembayaran,
        array_sales[i].metode_pembayaran);
}

void initialize_data(){
    FILE *fp;
    int file_index = index_counter("sales.txt");
    fp = fopen("sales.txt", "r");
    for(int i=0; i<file_index; i++){
            fscanf(fp, FORMAT_DATA_READ,
                  &array_sales[i].date[0],
                  &array_sales[i].date[1],
                  &array_sales[i].date[2],
                  &array_sales[i].nama,
                  &array_sales[i].size,
                  &array_sales[i].qty,
                  &array_sales[i].total_pembayaran,
                  &array_sales[i].metode_pembayaran);
      }
      fclose(fp);
}

void view_sales(){
      int file_index = index_counter("sales.txt");
      initialize_data();
      system("cls");
      print_sales();
}

void print_sales(){
      int file_index = index_counter("sales.txt");
      printf("-------------------------------------------------------------
---------------\n");
      printf("| No |  Tanggal  |           Nama         | Size | Qty |
Total  |  Metode  |\n");
      printf("-------------------------------------------------------------
---------------\n");
      for(int i=0; i<file_index; i++){
            print_array_sales(i);
      }
      printf("-------------------------------------------------------------
---------------\n");
}

typedef int (*compfn)(const void*, const void*);

int compare_nama(struct Sales *elem1, struct Sales *elem2){
	if(strcasecmp(elem1->nama, elem2->nama) < 0){
		return -1;
	}
	else if(strcasecmp(elem1->nama, elem2->nama) > 0){
		return 1;
	}
	return 0;
}

int compare_qty(struct Sales *elem1, struct Sales *elem2){
	if(elem1->qty < elem2->qty){
		return -1;
	}
	else if(elem1->qty > elem2->qty){
		return 1;
	}
	return 0;
}

int compare_total(struct Sales *elem1, struct Sales *elem2){
	if(elem1->total_pembayaran < elem2->total_pembayaran){
		return -1;
	}
	else if(elem1->total_pembayaran > elem2->total_pembayaran){
		return 1;
	}
	return 0;
}

void sort_sales(){
	int file_index = index_counter("sales.txt");
	int sub_input;
	initialize_data();
	system("cls");
	printf("Sort berdasarkan:\n");
	printf("-----------------\n");
```

```
      printf("1. Nama Produk\n");
      printf("2. Qty\n");
      printf("3. Total\n");
      printf("0. Back\n");
      printf("-----------------\n");
      printf("Input: "); scanf("%d", &sub_input); getchar();
      switch(sub_input){
            case 1:
                  qsort((void *) &array_sales, file_index, sizeof(struct
Sales), (compfn)compare_nama);
                  system("cls");
                  printf("Sort Berdasarkan Nama Produk\n");
                  print_sales();
                  system("pause");
                  sort_sales();
                  break;
            case 2:
                  qsort((void *) &array_sales, file_index, sizeof(struct
Sales), (compfn)compare_qty);
                  system("cls");
                  printf("Sort Berdasarkan Qty Produk Terjual\n");
                  print_sales();
                  system("pause");
                  sort_sales();
                  break;
            case 3:
                  qsort((void *) &array_sales, file_index, sizeof(struct
Sales), (compfn)compare_total);
                  system("cls");
                  printf("Sort Berdasarkan Nama Produk\n");
                  print_sales();
                  system("pause");
                  sort_sales();
                  break;
            case 0:
                  menu_admin();
                  break;
            default:
                  printf("Input salah!\n");
                  system("pause");
                  sort_sales();
                  break;
      }
}

void search_by_name(){
      char key[25];
      int found;
      int file_index=index_counter("sales.txt");
      system("cls");
    printf("Masukkan nama produk yang akan dicari : ");
    scanf("%s", key);
    getchar();
    found=0;
    for(int i=0; i<file_index; i++){
      if(strstr(array_sales[i].nama, key) != NULL){
            found = 1;
            }
      }
      if(found==0){
        printf("Data tidak ditemukan\n");
        system("pause");
        search_sales();
    }
    else{
      printf("-------------------------------------------------------------
---------------\n");
            printf("| No |  Tanggal  |           Nama         | Size | Qty
| Total  |  Metode  |\n");
            printf("-------------------------------------------------------
---------------------\n");
        for(int i=0; i<file_index; i++){
            if(strstr(array_sales[i].nama, key) != NULL){
                  print_array_sales(i);
                  }
            }
            printf("--------------------------------------------------------
---------------------\n");
    }
    system("pause");
    search_sales();
}

void search_by_qty(){
      char key[25];
      int found, high, low;
      int file_index=index_counter("sales.txt");
      system("cls");
    printf("Masukkan qty terendah : ");
    scanf("%d", &low);
    printf("Masukkan qty tertinggi : ");
    scanf("%d", &high);
    for(int i = 0; i < file_index; i++){
        if(array_sales[i].qty >= low && array_sales[i].qty <= high){
            found=1;
            };
    }
    if (found == 0){
        printf("Data tidak ditemukan\n");
        system("pause");
        search_sales();
    }
    else{
      printf("--------------------------------------------------------------
---------------\n");
            printf("| No |  Tanggal  |            Nama         | Size | Qty
| Total  |  Metode   |\n");
            printf("--------------------------------------------------------
---------------------\n");
        for(int i = 0; i < file_index; i++){
            if(array_sales[i].qty >= low && array_sales[i].qty <= high){
                print_array_sales(i);
            }
        }
        printf("-----------------------------------------------------------
-----------------\n");
    }
}


void search_by_sales(){
      char key[25];
      int found, high, low;
      int file_index=index_counter("sales.txt");
      system("cls");
    printf("Masukkan pembayaran terendah : ");
    scanf("%d", &low);
    printf("Masukkan pembayaran tertinggi : ");
    scanf("%d", &high);
    for(int i = 0; i < file_index; i++){
        if(array_sales[i].total_pembayaran >= low &&
array_sales[i].total_pembayaran <= high){
            found=1;
            };
    }
    if (found == 0){
        printf("Data tidak ditemukan\n");
        system("pause");
        search_sales();
    }
    else{
      printf("-------------------------------------------------------------
---------------\n");
            printf("| No |  Tanggal  |          Nama           | Size | Qty
| Total  |  Metode  |\n");
            printf("-------------------------------------------------------
---------------------\n");
        for(int i = 0; i < file_index; i++){
            if(array_sales[i].total_pembayaran >= low &&
array_sales[i].total_pembayaran <= high){
                print_array_sales(i);
            }
        }
        printf("----------------------------------------------------------
-----------------\n");
    }
}

void search_by_method(){
      char key[25];
      int found;
      int file_index=index_counter("sales.txt");
      system("cls");
    printf("Masukkan metode pembayaran yang akan dicari : ");
    scanf("%s", key);
    getchar();
    found=0;
    for(int i=0; i<file_index; i++){
      if(strstr(array_sales[i].metode_pembayaran, key) != NULL){
            found = 1;
            }
      }
      if(found==0){
        printf("Data tidak ditemukan\n");
        system("pause");
        search_sales();
    }
    else{
      printf("------------------------------------------------------------
---------------\n");
            printf("| No |  Tanggal  |           Nama         | Size | Qty
| Total  |  Metode  |\n");
            printf("------------------------------------------------------
---------------------\n");
        for(int i=0; i<file_index; i++){
            if(strstr(array_sales[i].metode_pembayaran, key) != NULL){
                  print_array_sales(i);
                  }
            }
            printf("-------------------------------------------------------
---------------------\n");
    }
    system("pause");
    search_sales();
}

void search_sales(){
      int file_index = index_counter("sales.txt");
      int sub_input;
      char key[25];
      int found;
      int high, low;
      initialize_data();
      system("cls");
      printf("Search berdasarkan:\n");
      printf("-----------------\n");
      printf("1. Nama Produk\n");
      printf("2. Qty\n");
      printf("3. Total Pembayaran\n");
      printf("4. Metode Pembayaran\n");
      printf("0. Back\n");
      printf("-----------------\n");
      printf("Input: "); scanf("%d", &sub_input); getchar();
      switch(sub_input){
            case 1:
                  search_by_name();
                  break;
            case 2:
                  search_by_qty();
                  break;
            case 3:
                  search_by_sales();
                  break;
            case 4:
                  search_by_method();
```

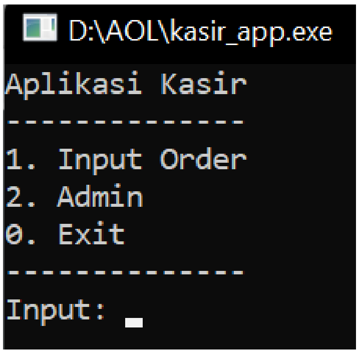
```
                        break;
                case 0:
                        menu_admin();
                        break;
                default:
                        printf("Input salah!\n");
                        system("pause");
                        sort_sales();
                        break;
        }
}

void menu_admin(){
        char sub_input;
        system("cls");
        printf("Menu Admin\n");
        printf("--------------\n");
        printf("1. View Sales\n");
        printf("2. Sort Sales\n");
        printf("3. Search Sales\n");
        printf("0. Back\n");
        printf("--------------\n");
        printf("Input: "); scanf("%d", &sub_input); getchar();
        switch(sub_input){
                case 1:
                        view_sales();
                        system("pause");
                        menu_admin();
                        break;
                case 2:
                        sort_sales();
                        break;
                case 3:
                        search_sales();
                        break;
                case 0:
```

```
            system("cls");
            main();
            break;
        default:
            printf("Input salah!\n");
            system("pause");
            menu_admin();
            break;
    }
}
```

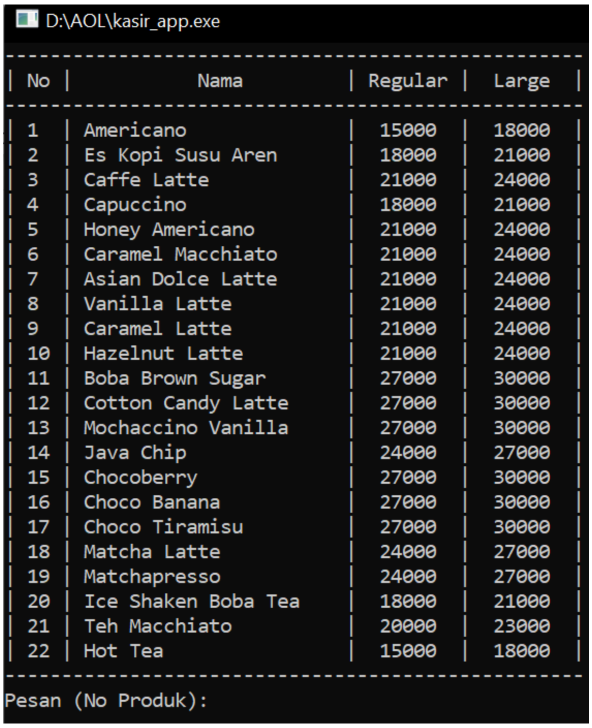
BAB IV
TAMPILAN HASIL

1. Tampilan Awal



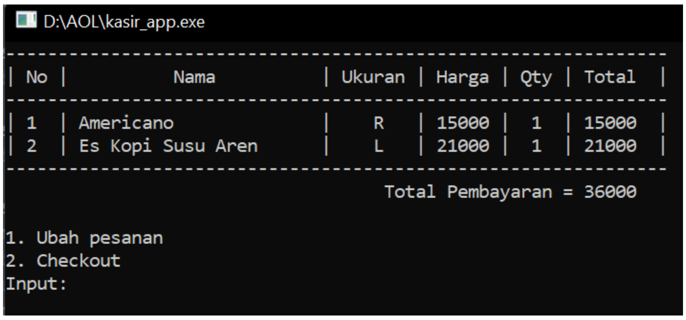
```
Aplikasi Kasir
--------------
1. Input Order
2. Admin
0. Exit
--------------
Input: _
```

2. Tampilan Input Order



```
---------------------------------------------------
| No |            Nama          | Regular |  Large  |
---------------------------------------------------
| 1  | Americano                |  15000  |  18000  |
| 2  | Es Kopi Susu Aren        |  18000  |  21000  |
| 3  | Caffe Latte              |  21000  |  24000  |
| 4  | Capuccino                |  18000  |  21000  |
| 5  | Honey Americano          |  21000  |  24000  |
| 6  | Caramel Macchiato        |  21000  |  24000  |
| 7  | Asian Dolce Latte        |  21000  |  24000  |
| 8  | Vanilla Latte            |  21000  |  24000  |
| 9  | Caramel Latte            |  21000  |  24000  |
| 10 | Hazelnut Latte           |  21000  |  24000  |
| 11 | Boba Brown Sugar         |  27000  |  30000  |
| 12 | Cotton Candy Latte       |  27000  |  30000  |
| 13 | Mochaccino Vanilla       |  27000  |  30000  |
| 14 | Java Chip                |  24000  |  27000  |
| 15 | Chocoberry               |  27000  |  30000  |
| 16 | Choco Banana             |  27000  |  30000  |
| 17 | Choco Tiramisu           |  27000  |  30000  |
| 18 | Matcha Latte             |  24000  |  27000  |
| 19 | Matchapresso             |  24000  |  27000  |
| 20 | Ice Shaken Boba Tea      |  18000  |  21000  |
| 21 | Teh Macchiato            |  20000  |  23000  |
| 22 | Hot Tea                  |  15000  |  18000  |
---------------------------------------------------
Pesan (No Produk):
```

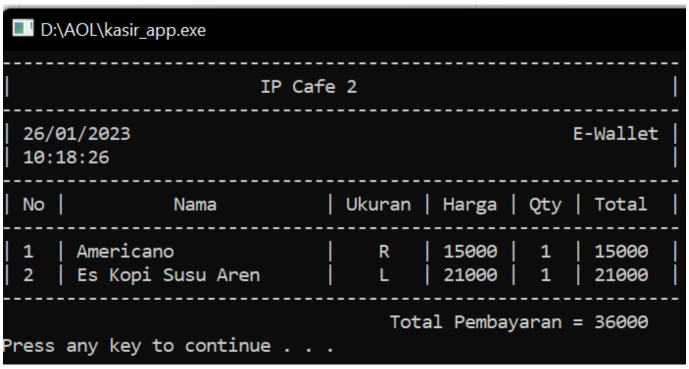
3. Tampilan Rekap Order



```
--------------------------------------------------------------
| No |          Nama           | Ukuran | Harga | Qty | Total  |
--------------------------------------------------------------
| 1  | Americano               |   R    | 15000 |  1  | 15000  |
| 2  | Es Kopi Susu Aren       |   L    | 21000 |  1  | 21000  |
--------------------------------------------------------------
                                    Total Pembayaran = 36000

1. Ubah pesanan
2. Checkout
Input:
```

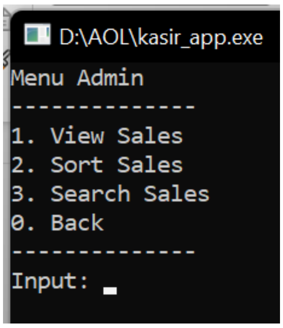
4. Tampilan Checkout



D:\AOL\kasir_app.exe

```
---------------------------------------------------------------
|                          IP Cafe 2                          |
---------------------------------------------------------------
| 26/01/2023                                        E-Wallet |
| 10:18:26                                                   |
---------------------------------------------------------------
| No |           Nama            | Ukuran | Harga | Qty | Total  |
---------------------------------------------------------------
| 1  | Americano                 |   R    | 15000 |  1  | 15000  |
| 2  | Es Kopi Susu Aren         |   L    | 21000 |  1  | 21000  |
---------------------------------------------------------------
                                 Total Pembayaran = 36000
Press any key to continue . . .
```

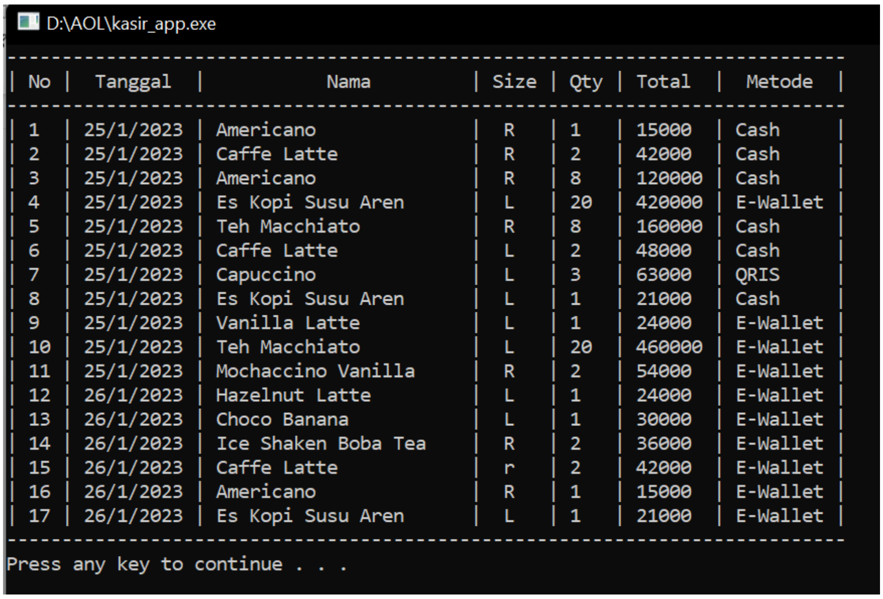
5. Tampilan Menu Admin



D:\AOL\kasir_app.exe

```
Menu Admin
-------------
1. View Sales
2. Sort Sales
3. Search Sales
0. Back
-------------
Input: _
```

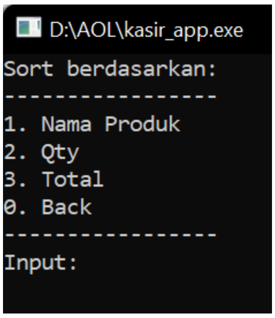
6. Tampilan View Sales



D:\AOL\kasir_app.exe

```
------------------------------------------------------------------------------
| No |  Tanggal  |           Nama           | Size | Qty | Total  |  Metode  |
------------------------------------------------------------------------------
| 1  | 25/1/2023 | Americano                |  R   | 1   | 15000  | Cash     |
| 2  | 25/1/2023 | Caffe Latte              |  R   | 2   | 42000  | Cash     |
| 3  | 25/1/2023 | Americano                |  R   | 8   | 120000 | Cash     |
| 4  | 25/1/2023 | Es Kopi Susu Aren        |  L   | 20  | 420000 | E-Wallet |
| 5  | 25/1/2023 | Teh Macchiato            |  R   | 8   | 160000 | Cash     |
| 6  | 25/1/2023 | Caffe Latte              |  L   | 2   | 48000  | Cash     |
| 7  | 25/1/2023 | Capuccino                |  L   | 3   | 63000  | QRIS     |
| 8  | 25/1/2023 | Es Kopi Susu Aren        |  L   | 1   | 21000  | Cash     |
| 9  | 25/1/2023 | Vanilla Latte            |  L   | 1   | 24000  | E-Wallet |
| 10 | 25/1/2023 | Teh Macchiato            |  L   | 20  | 460000 | E-Wallet |
| 11 | 25/1/2023 | Mochaccino Vanilla       |  R   | 2   | 54000  | E-Wallet |
| 12 | 26/1/2023 | Hazelnut Latte           |  L   | 1   | 24000  | E-Wallet |
| 13 | 26/1/2023 | Choco Banana             |  L   | 1   | 30000  | E-Wallet |
| 14 | 26/1/2023 | Ice Shaken Boba Tea      |  R   | 2   | 36000  | E-Wallet |
| 15 | 26/1/2023 | Caffe Latte              |  r   | 2   | 42000  | E-Wallet |
| 16 | 26/1/2023 | Americano                |  R   | 1   | 15000  | E-Wallet |
| 17 | 26/1/2023 | Es Kopi Susu Aren        |  L   | 1   | 21000  | E-Wallet |
------------------------------------------------------------------------------
Press any key to continue . . .
```

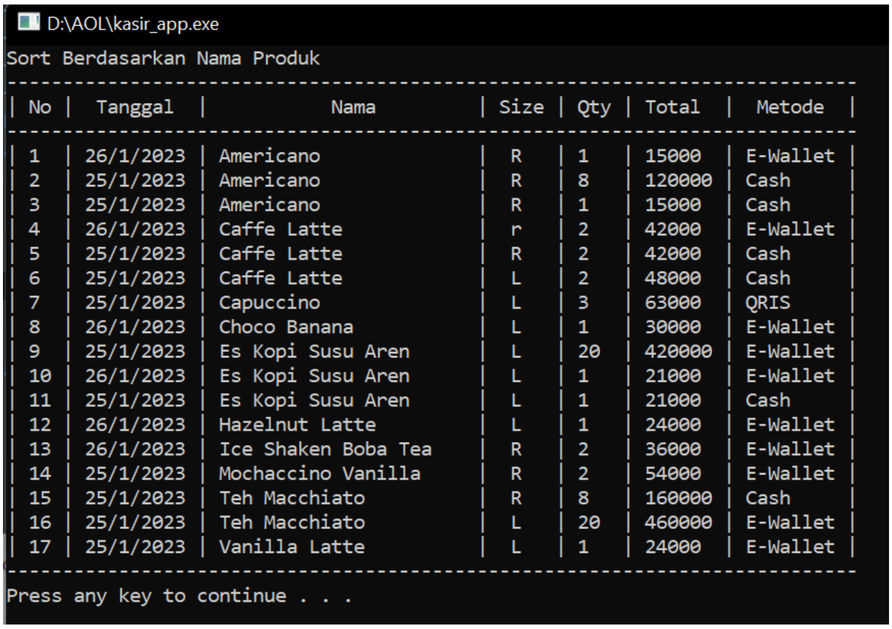
7. Tampilan Sort Sales



D:\AOL\kasir_app.exe

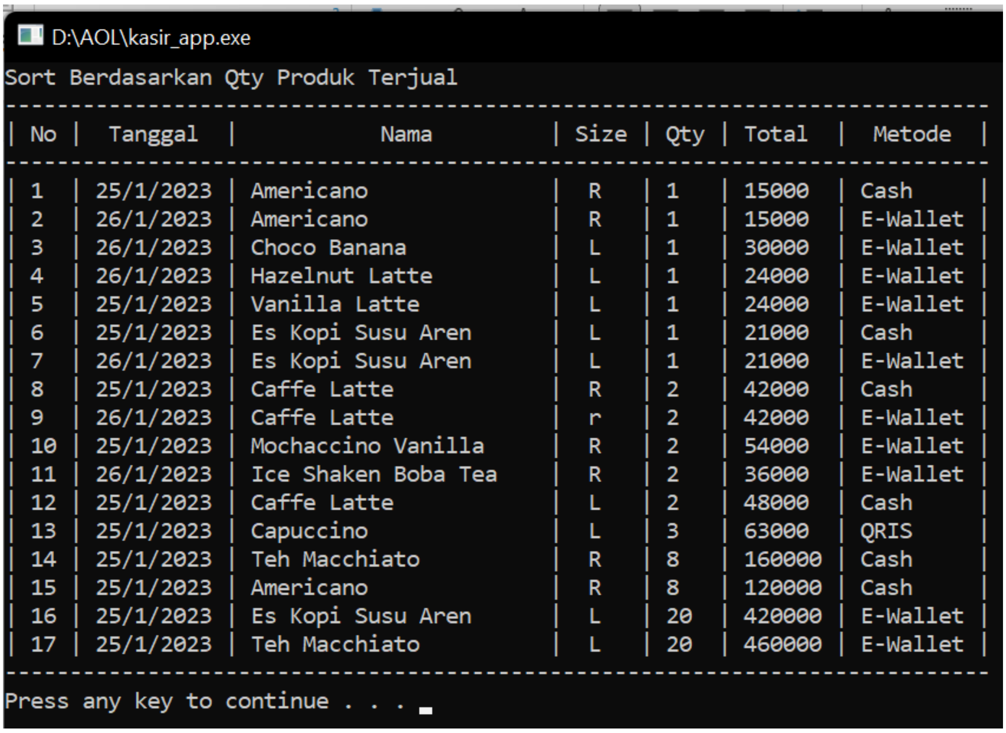
```
Sort berdasarkan:
----------------
1. Nama Produk
2. Qty
3. Total
0. Back
----------------
Input:
```

8. Tampilan Sort Sales Berdasarkan Nama Produk



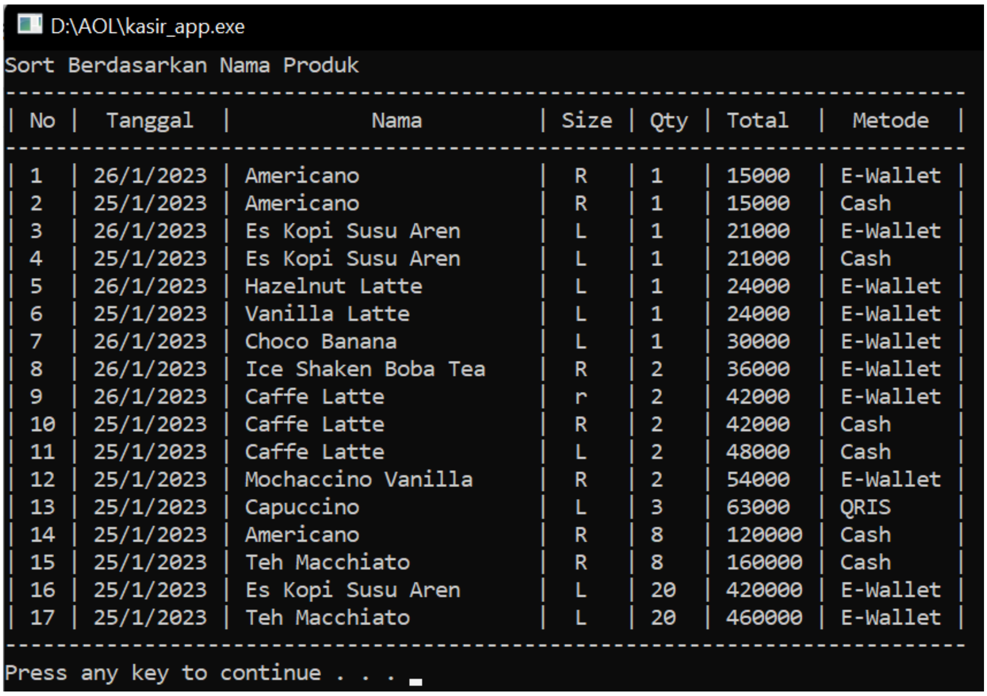
```
Sort Berdasarkan Nama Produk
----------------------------------------------------------------------------
| No |  Tanggal  |             Nama          | Size | Qty | Total  |  Metode  |
----------------------------------------------------------------------------
| 1  | 26/1/2023 | Americano                 |  R   | 1   | 15000  | E-Wallet |
| 2  | 25/1/2023 | Americano                 |  R   | 8   | 120000 | Cash     |
| 3  | 25/1/2023 | Americano                 |  R   | 1   | 15000  | Cash     |
| 4  | 26/1/2023 | Caffe Latte               |  r   | 2   | 42000  | E-Wallet |
| 5  | 25/1/2023 | Caffe Latte               |  R   | 2   | 42000  | Cash     |
| 6  | 25/1/2023 | Caffe Latte               |  L   | 2   | 48000  | Cash     |
| 7  | 25/1/2023 | Capuccino                 |  L   | 3   | 63000  | QRIS     |
| 8  | 26/1/2023 | Choco Banana              |  L   | 1   | 30000  | E-Wallet |
| 9  | 25/1/2023 | Es Kopi Susu Aren         |  L   | 20  | 420000 | E-Wallet |
| 10 | 26/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | E-Wallet |
| 11 | 25/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | Cash     |
| 12 | 26/1/2023 | Hazelnut Latte            |  L   | 1   | 24000  | E-Wallet |
| 13 | 26/1/2023 | Ice Shaken Boba Tea       |  R   | 2   | 36000  | E-Wallet |
| 14 | 25/1/2023 | Mochaccino Vanilla        |  R   | 2   | 54000  | E-Wallet |
| 15 | 25/1/2023 | Teh Macchiato             |  R   | 8   | 160000 | Cash     |
| 16 | 25/1/2023 | Teh Macchiato             |  L   | 20  | 460000 | E-Wallet |
| 17 | 25/1/2023 | Vanilla Latte             |  L   | 1   | 24000  | E-Wallet |
----------------------------------------------------------------------------
Press any key to continue . . .
```

9. Tampilan Sort Sales Berdasarkan Qty Produk Terjual



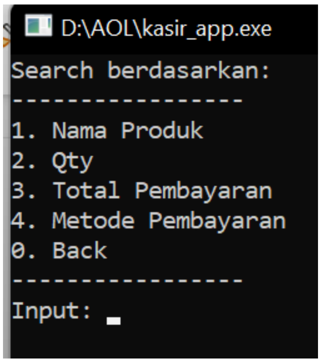
```
Sort Berdasarkan Qty Produk Terjual
----------------------------------------------------------------------------
| No |  Tanggal  |             Nama          | Size | Qty | Total  |  Metode  |
----------------------------------------------------------------------------
| 1  | 25/1/2023 | Americano                 |  R   | 1   | 15000  | Cash     |
| 2  | 26/1/2023 | Americano                 |  R   | 1   | 15000  | E-Wallet |
| 3  | 26/1/2023 | Choco Banana              |  L   | 1   | 30000  | E-Wallet |
| 4  | 26/1/2023 | Hazelnut Latte            |  L   | 1   | 24000  | E-Wallet |
| 5  | 25/1/2023 | Vanilla Latte             |  L   | 1   | 24000  | E-Wallet |
| 6  | 25/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | Cash     |
| 7  | 26/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | E-Wallet |
| 8  | 25/1/2023 | Caffe Latte               |  R   | 2   | 42000  | Cash     |
| 9  | 26/1/2023 | Caffe Latte               |  r   | 2   | 42000  | E-Wallet |
| 10 | 25/1/2023 | Mochaccino Vanilla        |  R   | 2   | 54000  | E-Wallet |
| 11 | 26/1/2023 | Ice Shaken Boba Tea       |  R   | 2   | 36000  | E-Wallet |
| 12 | 25/1/2023 | Caffe Latte               |  L   | 2   | 48000  | Cash     |
| 13 | 25/1/2023 | Capuccino                 |  L   | 3   | 63000  | QRIS     |
| 14 | 25/1/2023 | Teh Macchiato             |  R   | 8   | 160000 | Cash     |
| 15 | 25/1/2023 | Americano                 |  R   | 8   | 120000 | Cash     |
| 16 | 25/1/2023 | Es Kopi Susu Aren         |  L   | 20  | 420000 | E-Wallet |
| 17 | 25/1/2023 | Teh Macchiato             |  L   | 20  | 460000 | E-Wallet |
----------------------------------------------------------------------------
Press any key to continue . . .
```

10. Tampilan Sort Sales Berdasarkan Total Pembayaran



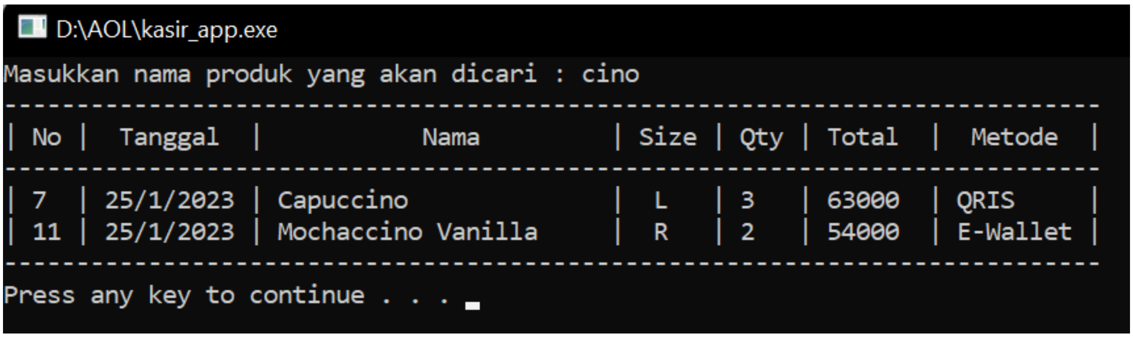
```
Sort Berdasarkan Nama Produk
-------------------------------------------------------------------------------
| No |  Tanggal  |             Nama         | Size | Qty | Total  |  Metode  |
-------------------------------------------------------------------------------
| 1  | 26/1/2023 | Americano                |  R   | 1   | 15000  | E-Wallet |
| 2  | 25/1/2023 | Americano                |  R   | 1   | 15000  | Cash     |
| 3  | 26/1/2023 | Es Kopi Susu Aren        |  L   | 1   | 21000  | E-Wallet |
| 4  | 25/1/2023 | Es Kopi Susu Aren        |  L   | 1   | 21000  | Cash     |
| 5  | 26/1/2023 | Hazelnut Latte           |  L   | 1   | 24000  | E-Wallet |
| 6  | 25/1/2023 | Vanilla Latte            |  L   | 1   | 24000  | E-Wallet |
| 7  | 26/1/2023 | Choco Banana             |  L   | 1   | 30000  | E-Wallet |
| 8  | 26/1/2023 | Ice Shaken Boba Tea      |  R   | 2   | 36000  | E-Wallet |
| 9  | 26/1/2023 | Caffe Latte              |  r   | 2   | 42000  | E-Wallet |
| 10 | 25/1/2023 | Caffe Latte              |  R   | 2   | 42000  | Cash     |
| 11 | 25/1/2023 | Caffe Latte              |  L   | 2   | 48000  | Cash     |
| 12 | 25/1/2023 | Mochaccino Vanilla       |  R   | 2   | 54000  | E-Wallet |
| 13 | 25/1/2023 | Capuccino                |  L   | 3   | 63000  | QRIS     |
| 14 | 25/1/2023 | Americano                |  R   | 8   | 120000 | Cash     |
| 15 | 25/1/2023 | Teh Macchiato            |  R   | 8   | 160000 | Cash     |
| 16 | 25/1/2023 | Es Kopi Susu Aren        |  L   | 20  | 420000 | E-Wallet |
| 17 | 25/1/2023 | Teh Macchiato            |  L   | 20  | 460000 | E-Wallet |
-------------------------------------------------------------------------------
Press any key to continue . . .
```

11. Tampilan Search Sales



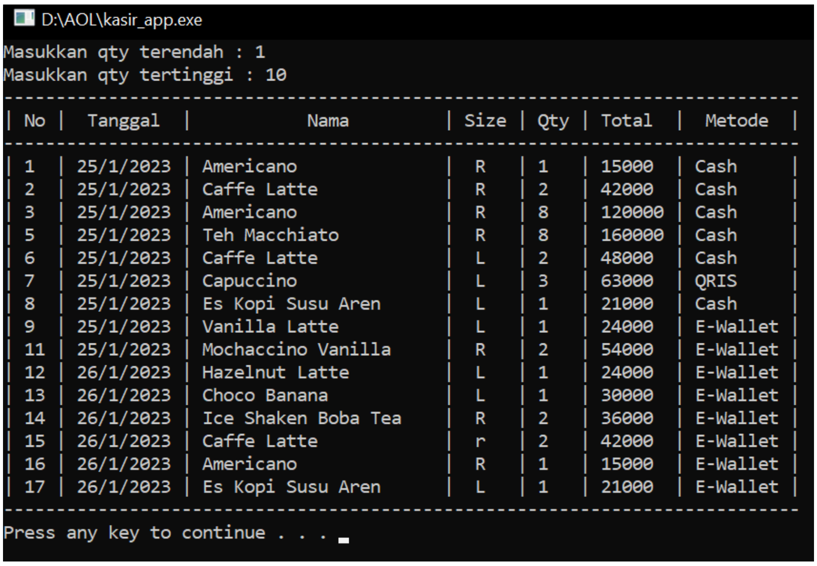
```
Search berdasarkan:
-----------------
1. Nama Produk
2. Qty
3. Total Pembayaran
4. Metode Pembayaran
0. Back
-----------------
Input:
```

12. Tampilan Search Berdasarkan Nama Produk



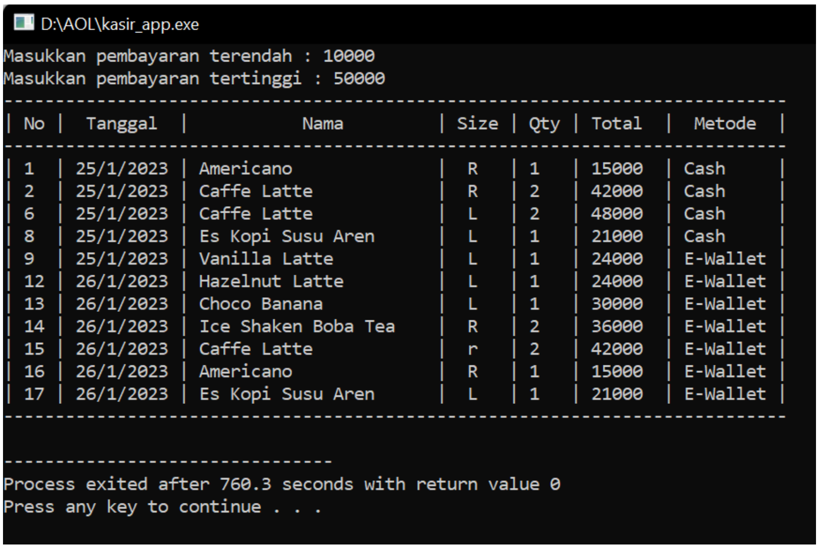
```
Masukkan nama produk yang akan dicari : cino
-------------------------------------------------------------------------------
| No |  Tanggal  |             Nama         | Size | Qty | Total  |  Metode  |
-------------------------------------------------------------------------------
| 7  | 25/1/2023 | Capuccino                |  L   | 3   | 63000  | QRIS     |
| 11 | 25/1/2023 | Mochaccino Vanilla       |  R   | 2   | 54000  | E-Wallet |
-------------------------------------------------------------------------------
Press any key to continue . . .
```

13. Tampilan Search Berdasarkan Qty Penjualan



```
Masukkan qty terendah : 1
Masukkan qty tertinggi : 10
------------------------------------------------------------------------------
| No |  Tanggal  |             Nama           | Size | Qty | Total  |  Metode  |
------------------------------------------------------------------------------
| 1  | 25/1/2023 | Americano                  |  R   | 1   | 15000  | Cash     |
| 2  | 25/1/2023 | Caffe Latte                |  R   | 2   | 42000  | Cash     |
| 3  | 25/1/2023 | Americano                  |  R   | 8   | 120000 | Cash     |
| 5  | 25/1/2023 | Teh Macchiato              |  R   | 8   | 160000 | Cash     |
| 6  | 25/1/2023 | Caffe Latte                |  L   | 2   | 48000  | Cash     |
| 7  | 25/1/2023 | Capuccino                  |  L   | 3   | 63000  | QRIS     |
| 8  | 25/1/2023 | Es Kopi Susu Aren          |  L   | 1   | 21000  | Cash     |
| 9  | 25/1/2023 | Vanilla Latte              |  L   | 1   | 24000  | E-Wallet |
| 11 | 25/1/2023 | Mochaccino Vanilla         |  R   | 2   | 54000  | E-Wallet |
| 12 | 26/1/2023 | Hazelnut Latte             |  L   | 1   | 24000  | E-Wallet |
| 13 | 26/1/2023 | Choco Banana               |  L   | 1   | 30000  | E-Wallet |
| 14 | 26/1/2023 | Ice Shaken Boba Tea        |  R   | 2   | 36000  | E-Wallet |
| 15 | 26/1/2023 | Caffe Latte                |  r   | 2   | 42000  | E-Wallet |
| 16 | 26/1/2023 | Americano                  |  R   | 1   | 15000  | E-Wallet |
| 17 | 26/1/2023 | Es Kopi Susu Aren          |  L   | 1   | 21000  | E-Wallet |
------------------------------------------------------------------------------
Press any key to continue . . .
```

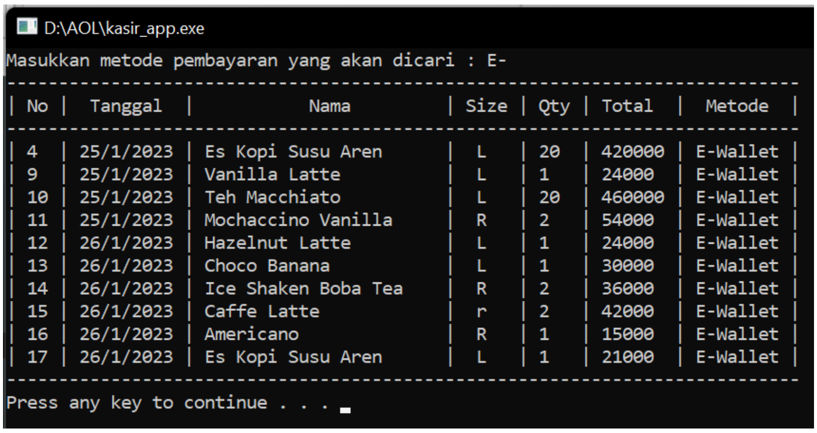
14. Tampilan Search Berdasarkan Total Pembayaran



```
Masukkan pembayaran terendah : 10000
Masukkan pembayaran tertinggi : 50000
----------------------------------------------------------------------------
| No |  Tanggal  |             Nama          | Size | Qty | Total  |  Metode  |
----------------------------------------------------------------------------
| 1  | 25/1/2023 | Americano                 |  R   | 1   | 15000  | Cash     |
| 2  | 25/1/2023 | Caffe Latte               |  R   | 2   | 42000  | Cash     |
| 6  | 25/1/2023 | Caffe Latte               |  L   | 2   | 48000  | Cash     |
| 8  | 25/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | Cash     |
| 9  | 25/1/2023 | Vanilla Latte             |  L   | 1   | 24000  | E-Wallet |
| 12 | 26/1/2023 | Hazelnut Latte            |  L   | 1   | 24000  | E-Wallet |
| 13 | 26/1/2023 | Choco Banana              |  L   | 1   | 30000  | E-Wallet |
| 14 | 26/1/2023 | Ice Shaken Boba Tea       |  R   | 2   | 36000  | E-Wallet |
| 15 | 26/1/2023 | Caffe Latte               |  r   | 2   | 42000  | E-Wallet |
| 16 | 26/1/2023 | Americano                 |  R   | 1   | 15000  | E-Wallet |
| 17 | 26/1/2023 | Es Kopi Susu Aren         |  L   | 1   | 21000  | E-Wallet |
----------------------------------------------------------------------------

--------------------------------
Process exited after 760.3 seconds with return value 0
Press any key to continue . . .
```

15. Tampilan Search Berdasarkan Metode Pembayaran



```
Masukkan metode pembayaran yang akan dicari : E-
------------------------------------------------------------------------------
| No |  Tanggal  |             Nama           | Size | Qty | Total  |  Metode  |
------------------------------------------------------------------------------
| 4  | 25/1/2023 | Es Kopi Susu Aren          |  L   | 20  | 420000 | E-Wallet |
| 9  | 25/1/2023 | Vanilla Latte              |  L   | 1   | 24000  | E-Wallet |
| 10 | 25/1/2023 | Teh Macchiato              |  L   | 20  | 460000 | E-Wallet |
| 11 | 25/1/2023 | Mochaccino Vanilla         |  R   | 2   | 54000  | E-Wallet |
| 12 | 26/1/2023 | Hazelnut Latte             |  L   | 1   | 24000  | E-Wallet |
| 13 | 26/1/2023 | Choco Banana               |  L   | 1   | 30000  | E-Wallet |
| 14 | 26/1/2023 | Ice Shaken Boba Tea        |  R   | 2   | 36000  | E-Wallet |
| 15 | 26/1/2023 | Caffe Latte                |  r   | 2   | 42000  | E-Wallet |
| 16 | 26/1/2023 | Americano                  |  R   | 1   | 15000  | E-Wallet |
| 17 | 26/1/2023 | Es Kopi Susu Aren          |  L   | 1   | 21000  | E-Wallet |
------------------------------------------------------------------------------
Press any key to continue . . .
```