**Experiment :1**

SHOPPING CART

**ALGORITHM**

Step 1: Start

Step 2: Declare and define functions “additem”, “deleteitem”, “updatebill” and “displaycart”

Step 3: Declare structure cart\_items with variables “code”, “quantity” and “cost”

Step 4: Declare variables stock[i][j], choice, item\_number, total.

Step 5: Print the stock items.

Step 6: Read choice from the user.  
 Case ‘1’ – call “additem” function which adds item in your cart.

Case ‘2’ – call “deleteitem” function which deletes items in your cart

Case ‘3’ – call “displaycart” function which displays your carts

Case ‘default’ – GOTO 8

Step 7: Call “updatebill” function to display items in your cart and the total bill.

Step 8: Stop

**Program**

#include<stdio.h>

#include<stdlib.h>

void additem();//function to add item

void deleteitem();//function to delete item

void updatebill();//function update cart after the add or deletion of item

void displaycart();//function to display cart

int stock[5][3]={{0,0,0},{1,10,100},{2,12,200},{3,14,150},{4,21,250}};//array of stocks present

typedef struct{

int code;

int quantity;

float cost;

}cartitems;

cartitems c[10]; //array of structure

int inum=0;//to keep count of number of items purchased

float total=0;//to calculate total cost for a item

int main()

{

int i,j,n,choice;

printf("\*\*\*\*\*\*\*\*\*SHOPING CART\*\*\*\*\*\*\*\*\*");

do{

printf("\n\nItem code\tQuantity\tPrice\n");//cart menu display

for(i=1;i<5;i++)

{

for(j=0;j<3;j++)

printf("%d\t\t",stock[i][j]);

printf("\n");

}

//Inputing choice from user

printf("\nPlease enter your choice:");

printf("\n\n 1: add item to cart\n 2:delete item from cart\n 3:display\n 4:exit:\n");

scanf("%d",&choice);

switch(choice)//switch to the entered choice

{

case 1:additem();

break;

case 2:deleteitem();

break;

case 3:displaycart();

break;

case 4: exit(0);

break;

default:printf("Enter the correct choice\n");

};

}while(1);

return 0;

}

void additem()

{

//Input item code and quantity

printf("\nEnter the code and quantity of the item to be added to your cart:");

scanf("%d %d",&c[inum].code,&c[inum].quantity);

if(c[inum].quantity<stock[c[inum].code][1])//check for quantity availability

{

c[inum].cost=c[inum].quantity\*stock[c[inum].code][2];//calculate cost

printf("\n the item with code%d is added to the cart\n", c[inum].code);

printf("\n your cart contains....\n");

printf("\n item code\t quantity\tcost\n");

printf("%d\t\t%d\t\t%0.2f",c[inum].code,c[inum].quantity,c[inum].cost);

stock[c[inum].code][1]=stock[c[inum].code][1]-c[inum].quantity;//update quantity

inum++;

updatebill();//update stock array

}

else//executes when quantity is not sufficient

{

printf("\nStock does not has sufficient quantity\n");

}

return;

}

void deleteitem()

{

if(inum>0)//check for item in stock is purchased or not

{

printf("\n last item from your cart deleted\n");

inum--;

stock[c[inum].code][1]=stock[c[inum].code][1]+c[inum].quantity;//delete item

updatebill();

}

else

printf("\nItem is not added to cart for deletion\n");

return;

}

void updatebill()// function for bill calculation after each operation

{

int i;

total=0;

printf("\n there are %d items in your cart...\n\n",inum);

for(i=0;i<inum;i++)

total=total+c[i].cost;

return;

}

void displaycart()//Display purchased items

{

int i;

printf("\n there are %d items in your cart...\n\n",inum);

printf("\n item code\tquantity\tamount\n");

for(i=0;i<inum;i++)

printf("\n%d\t\t%d\t\t%5.2f",c[i].code,c[i].quantity,c[i].cost);

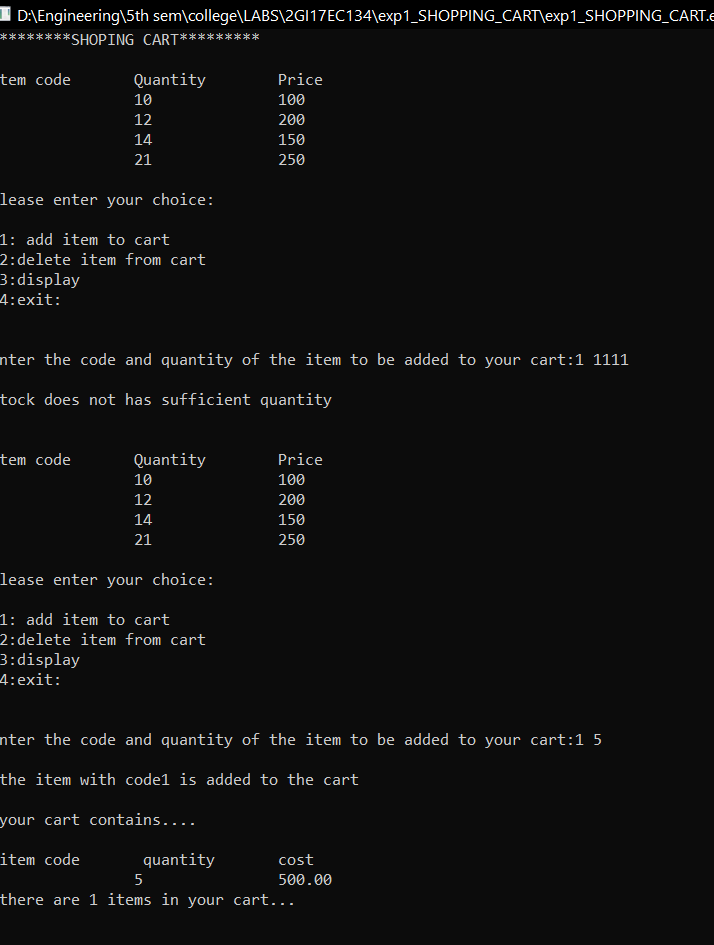
printf("\n\n\n\t\t\tGrand total is:%5.2f\n",total);

return;

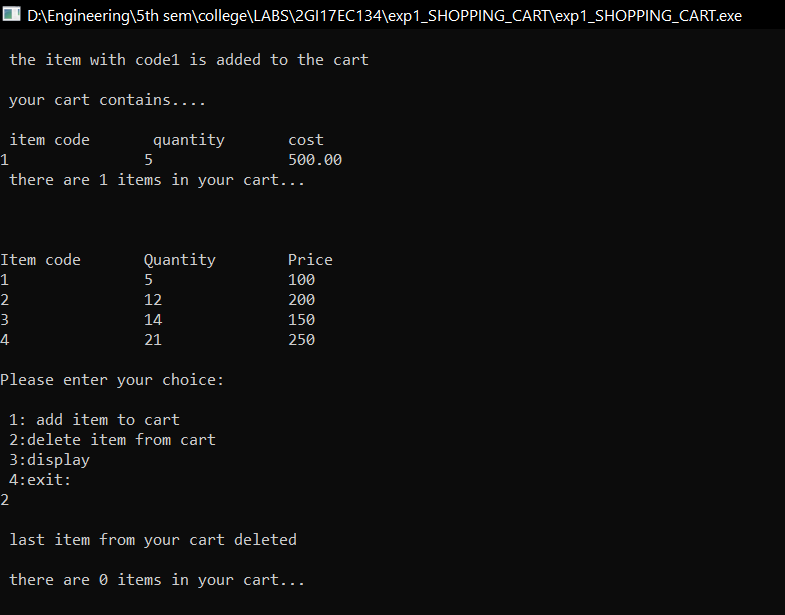
}

**SAMPLE INPUT /OUPUT**

**Add item**



**Delete item**



**Display item**