**Experiment** :1 SHOPPING CART

ALGORITHM

Step 1: Start

Step 2: Defining functions to add item to the cart, delete item from the cart, display the items in the cart and update the bill generated.

Step 3: Predefining item code, quantity and cost of the item using array.

Step 4: Declaring cartitems variable of data type struct.

Step 5: Print the stock items.

Step 6: Accept the input/choice from the user to perform the operation.  
 Case ‘1’:Add item in the cart.

Case ‘2’:Delete items in the cart

Case ‘3’:Displays items in the cart

Case ‘default’:exits the loop

Step 7: Update the bill and display items in the 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 to update the bill

void displaycart(); //function to display items in cart

int stock[5][3]={{0,0,0},{1,12,100},{2,15,200},{3,17,150},{4,25,250}}; //predefind stock

typedef struct{

int code;

int quantity;

float cost;

}cartitems;

cartitems c[10]; //array of structure

int inum=0;

float total=0;

int main()

{

int i,j,n,choice;

do{ //continous loop

printf("\n\nitem code\tquantity\tprice\n");

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

{

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

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

printf("\n");

}

printf("\n please enter your choice:");

printf("\n\n 1: add item to cart \t2:delete item from cart \t3:display \t4:exit:"); //choice

scanf("%d",&choice);

switch(choice)

{

case 1:additem(); //add function

break;

case 2:deleteitem(); //delete function

break;

case 3:displaycart(); //display function

break;

default: exit(0);

};

}while(1);

return 0;

}

void additem()

{

//printf("\n you have called additem\n");

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

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

c[inum].cost=c[inum].quantity\*stock[c[inum].code][2]; //for adding items

printf("\n the item with code%d is added to the cast\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); //displaying the item which has been added

stock[c[inum].code][1]=stock[c[inum].code][1]-c[inum].quantity; //Updating the predefined stock list

inum++;

updatebill();

return;

}

void deleteitem()

{

printf("\n your have called deleteitem()\n");

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

inum--;

stock[c[inum].code][1]=stock[c[inum].code][1]+c[inum].quantity; //deletion of the given item

updatebill();

return;

}

void updatebill()

{

int i;

total=0;

printf("\n you have called updatebill\n");

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

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

total=total+c[i].cost; //calculating total cost

return;

}

void displaycart()

{

int i;

//printf("\n you have called displaycart()\n");

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

printf("\n itemcode\tquantity\tamount\n");

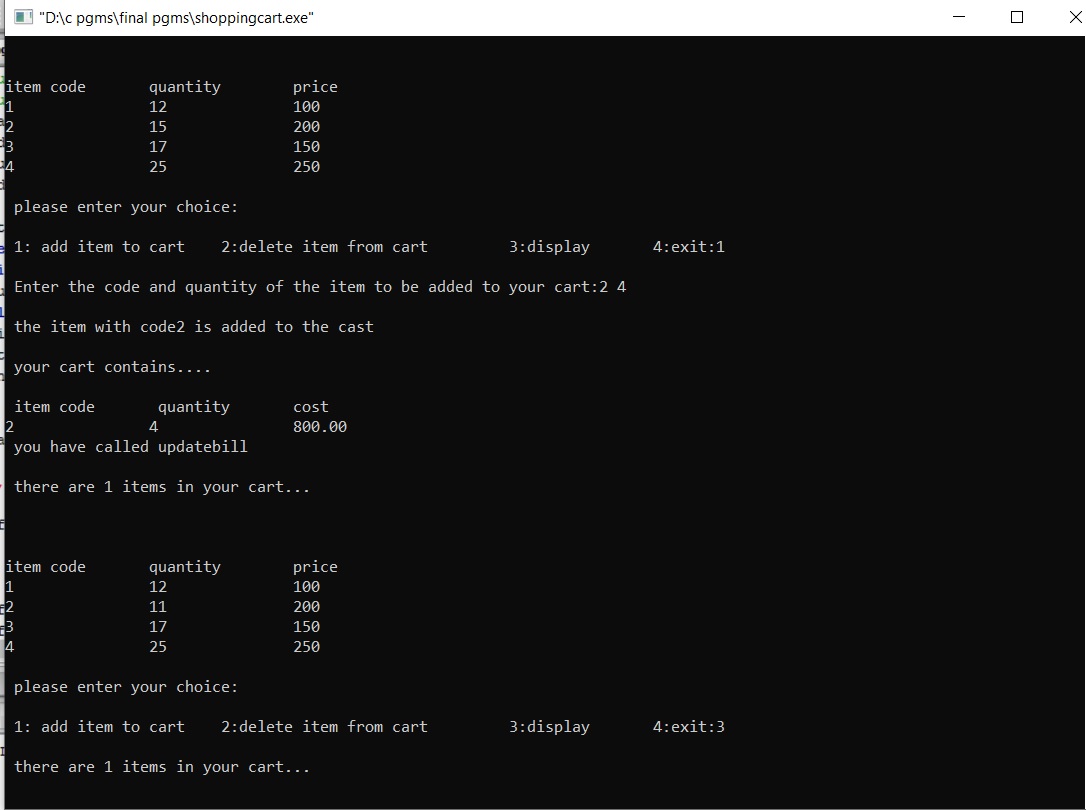
for(i=0;i<inum;i++)

printf("\n%d\t\t%d\t\t%5.2f",c[i].code,c[i].quantity,c[i].cost); //display code,quantity and price of item

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

return;

}

OUTPUTS:

