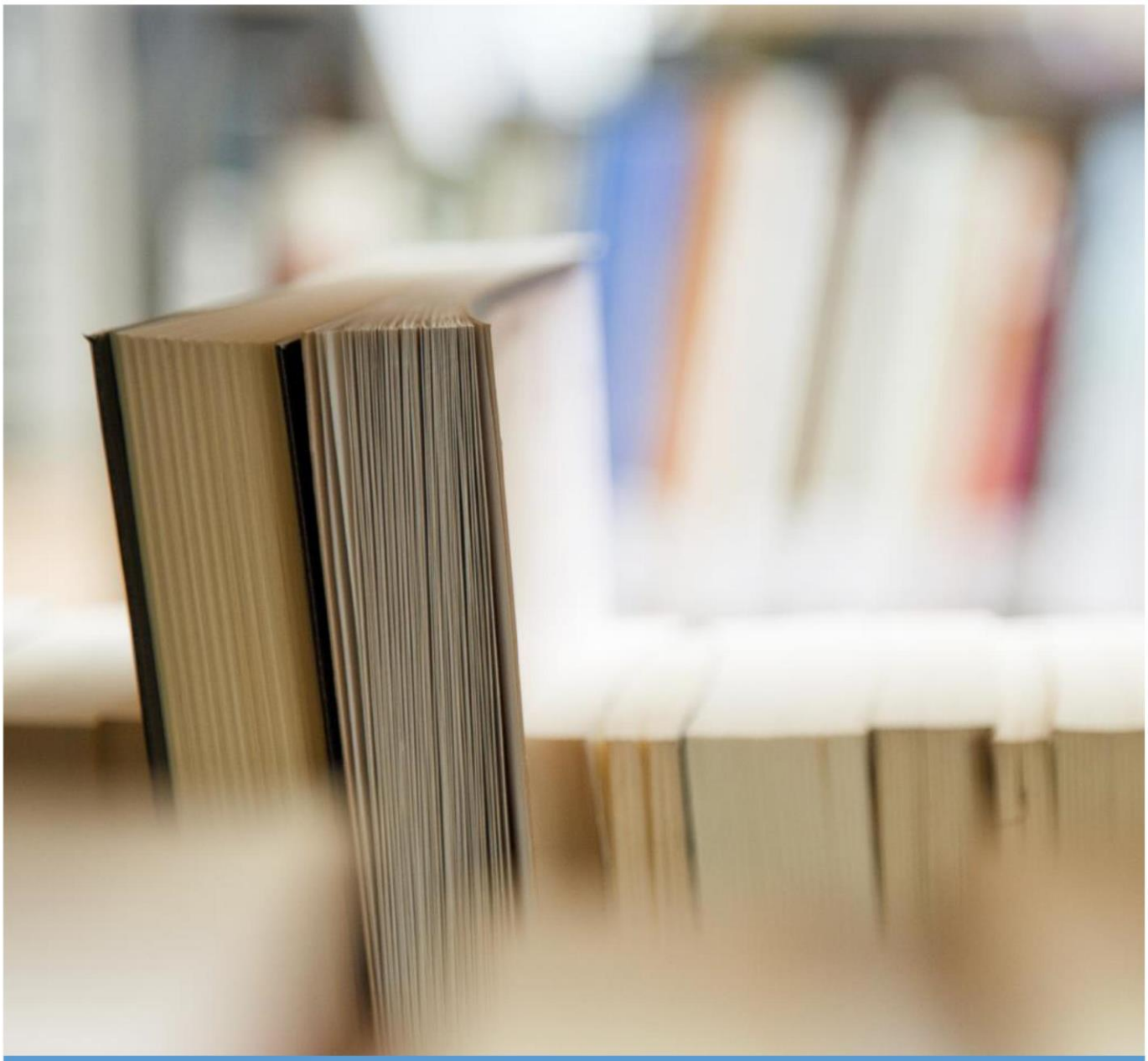


Computer Science Project

CANTEEN MANAGEMENT

HARSHIT & ABHIJIT XII-B

HIRANANDANI UPSCALE SCHOOL

DEPARTMENT OF COMP.SC

CERTIFICATE

This is to certify that Abhijit and Harshit, students of class XII-B have successfully completed the research on the below mentioned project under the guidance of Mrs. Suman Edison (Subject Teacher) during the year 2015-16 in partial fulfilment of Comp.Sc Practical examination conducted by AISSCE, Chennai.

Signature of External Examiner Signature of Comp.Sc Teacher

ACKNOWLEDGEMENT

Primarily we would thank God for being able to complete this project with success. Then we would like to thank my Comp.Sc teacher Mrs. Suman Edison , whose valuable guidance has been the ones that helped us patch this project and make it full proof success, her suggestions and her instructions has served as the major contributor towards the completion of the project.

Then we would like to thank our parents and friends who have helped us with their valuable suggestions and guidance has been helpful in various phases of the completion of the project.

Last but not the least we would like to thank our classmates who have helped us a lot.

Abhijit & Harshit

INDEX

AIM OF THE PROJECT

To write a C++ program for Canteen Management.

Introduction

C++ is a programming language which allows you to control your computer, making it do what you want it to do.

A C++ program is a collection of commands, which tell the computer to do "something". This collection of commands is usually called C++ source code, source code or just code. Commands are either "functions" or "keywords". Keywords are a basic building block of the language, while functions are, in fact, usually written in terms of simpler functions

Every program in C++ has one function, always named main, that is always called when your program first executes. From main, you can also call other functions whether they are written by us or, as mentioned earlier, provided by the compiler.

HARDWARE AND SOFTWARE SPECIFICATIONS

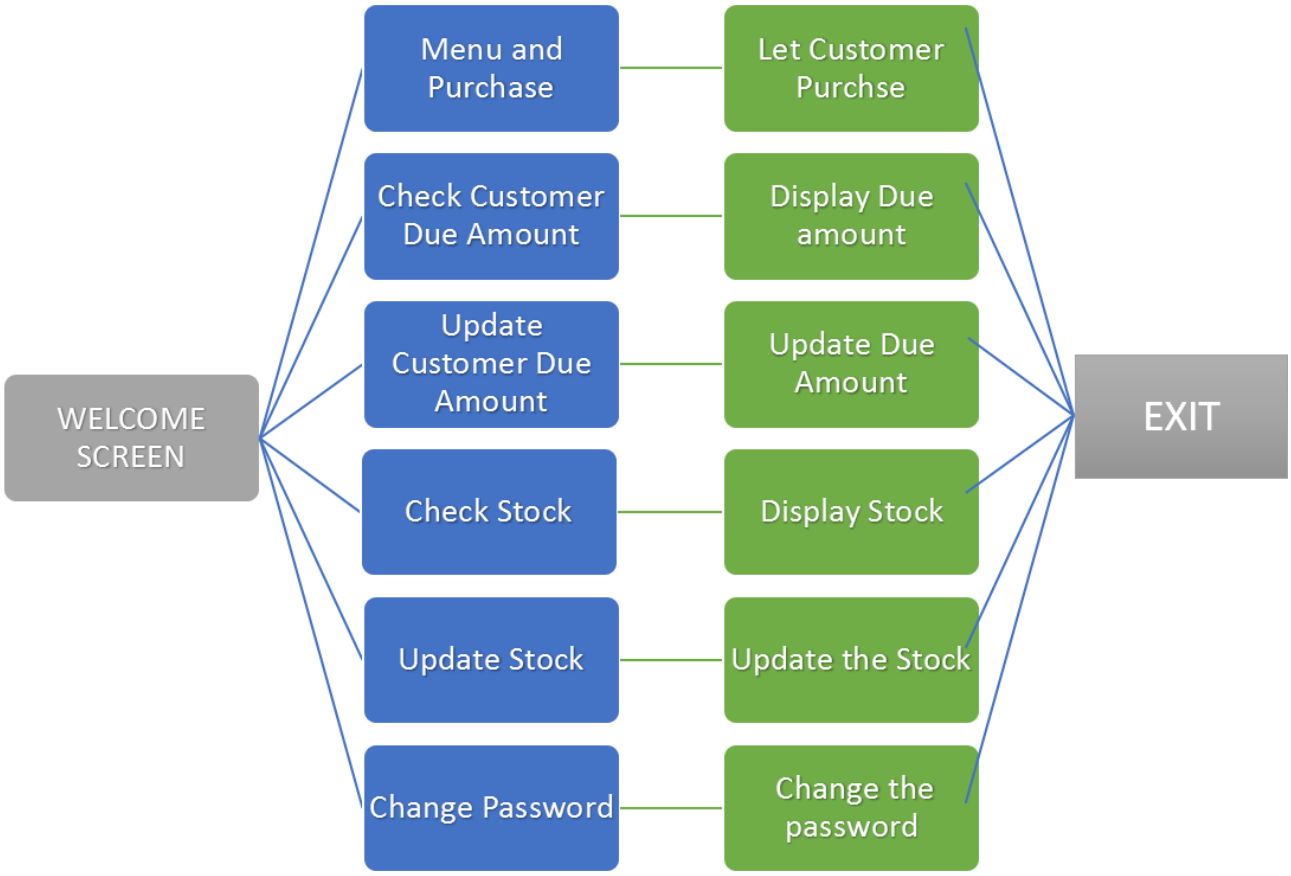
Hardware:

Model Name: IBALL PC
ProcessorName: Intel Core2Duo
Processor Speed: 2.8 GHz
Number of Processors: 1
Total Number of Cores: 2
L2 Cache (per Core): 256 KB
L3 Cache: 3 MB
Memory: 4 GB

Software:

Codeblocks 13.10
WINDOWS 7 ULTIMATE

SYSTEM DESIGN



About our Program

Canteen management system project is able to provide very fast service to their customers by using their records which have saved previously in database. And new records are added when new customer are visited to canteen shop.

Canteen management system project in c++ is used to uniquely identify customers by their customer's id. By using customer id we can enter data in database.

In Canteen management system project we can also maintain the payment record which is not cleared date. In canteen management system documentation customer also check their account, which service they have taken and can access the customer information in any time.

The canteen management system is also capable of maintaining a record of the stock available and updates the stock automatically when a purchase is made.

The program is safeguarded by a login id and password which will be given to the cashiers on the billing desk for them to access the program.

Program Functions

Header Files	Functions
<fstream.h>	open(),close()
<cstdlib.h>	atoi(),exit()
<conio.h>	getch()
<windows.h>	system("cls"),system("pause"),sleep()

void menu() - shows the menu for purchase

void password() - to check the password

void changepassword() – to change the password

void setcolor(int) - to set the colour of the text on the output screen

void welcome() - displays the welcome screen

void name() - displays our canteen's name

void upgradestockmanual() - to update the stock by the user

void purchase() - the main function which performs the purchase of the product

void updatebalance(int,int) - to update the customer's due amount

void checkstock() - displays the stock

void menu1() - displays the menu without prices

void upgradestock(int,int&,int) - updates the stock after a purchase is made

void thanks() - the final screen

void getbalance(int) - displays the due amount of a particular customer

void account(int,int) - updates customer's due amount after the purchase

SOURCE CODE

```
#include<iostream>
#include<fstream>
#include<windows.h>
#include<cstdlib>
#include<conio.h>


/*
1: Blue
2: Green
3: Cyan
4: Red
5: Purple
6: Yellow (Dark)
7: Default white
8: Gray/Grey
9: Bright blue
10: Brigth green
11: Bright cyan
12: Bright red
13: Pink/Magenta
14: Yellow
15: Bright white*/


using namespace std;

void menu();//shows the menu for purchase
void password();//to check the password
void changepassword();
void setcolor(int);//to set the colour of the text on the output screen
void welcome();//displays the welcome screen
void name();//displays our project name
void upgradestockmanual();//to update the stock by the user
void purchase();//the main function which performs the purchase of the product
void updatebalance(int,int);//to update the customer balance
void checkstock();//displays the stock
void menu1();//displays the menu without prices
void upgradestock(int,int&,int);//updates the stock after a purchase is made
void thanks();//the end screen
void getbalance(int);//displays the balance of a particular customer
void account(int,int);//updates their due amount after the purchase
int vp=15,ep=20,sw=30,vr=25,cpz=70,don=30,sam=10, chib=30,crmb=15,chocof=50;//the prices of the item
int sum,add,i,j,e,f,g,h,k,l,w,x,y,z,choice,admn,main1,serial,newamnt1,option1,func,d;
int m=0,n=0,o=0,p=0,q=0,r=0,s=0,t=0,u=0,v=0,m1=0,n1=0,o1=0,p1=0,q1=0,r1=0,s1=0,t1=0,u1=0,v1=0;//maintains the

char pu1[50],pu,d1[50],serial1[50];
```

```
int main()
{
setcolor(10);
welcome();
setcolor(12);
password();
do
{
system("cls");
name();
setcolor(15);
cout<<"\a";

cout<<"\n\n\n\t\t\t  1.MENU & PURCHASE";
cout<<"\n\n\n\t\t\t  2.CHECK CUSTOMER DUE AMOUNT";
cout<<"\n\n\n\t\t\t  3.UPDATE CUSTOMER DUE AMOUNT";
cout<<"\n\n\n\t\t\t  4.CHECK STOCK";
cout<<"\n\n\n\t\t\t  5.UPDATE STOCK";
cout<<"\n\n\n\t\t\t  6.CHANGE PASSWORD";
cout<<"\n\n\n\t\t\t  7.EXIT";
cout<<"\n\n\n\t\t\t  ENTER YOUR CHOICE(1-7): ";
cin>>d1;
d=atoi(d1);
system("cls");
name();
switch(d)
{
case 1:purchase();
break;


case 2:system("cls");
name();
cout<<"\n\n  Enter your admission no. : ";
cin>>admn;
getbalance(admn);
break;

case 3:system("cls");name();
cout<<"\n\n  Enter your Admission No.: ";
cin>>admn;
cout<<"\n\n  Enter the amount paid : ";
cin>>sum;
updatebalance(admn,sum);
break;

case 4:checkstock();
break;

case 5:upgradestockmanual();
break;

case 6:changepassword();
break;

case 7:main1=0;
```

```
90.  thanks();
91.  break;
92.  break;

93.  default: cout<<"\n\n\n\t\t\t Your choice is wrong"<<"\n\n\n\t\t\t";
94.  system("pause");
95.  main1=1;
96.  }
97.  }while(main1==1);
98.  }

99.  void getbalance(int w)
100. {
101. int flag=0;
102. int bal[10];
103. int checkroll[10];
104. int loop=0;
105. ifstream f3;
106. f3.open("balance.txt");
107. while(!f3.eof())
108. {
109. f3>>checkroll[loop];
110. f3>>bal[loop];
111. loop++;
112. }
113. f3.close();
114. do
115. {
116. for(int check=0;check<10;check++)
117. {
118. if(checkroll[check]==w)
119. {
120. flag=1;
121. break;
122. }
123. if(check==9)

124. {
125. cout<<"\n\n\n  You have entered the wrong admn no.";
126. cout<<"\n\n  Enter admission no. : ";
127. cin>>w;
128. break;
129. }
130. }
131. }while(flag==0);
132. ifstream f4;
133. f4.open("name.txt");
134. char checkname[10][10];
135. loop=0;
136. while(!f4.eof())
137. {
138. f4>>checkroll[loop];
139. f4>>checkname[loop];
140. loop++;
141. }
142. f4.close();
143. for(int loop3=0;loop3<10;loop3++)
```

```
{
if(checkroll[loop3]==w)
{
system("cls");name();
cout<<"\n\n\n\t\tWelcome "<<checkname[loop3];
cout<<"\n\n\t\tYour current due amount is: "<<bal[loop3]<<endl;
cout<<"\n\n\t\t";
break;
}
}
main1=1;
system("pause");
}

void menu()
{
setcolor(14);
cout<<"\n\n";
cout<<"\n Sno.   ITEMS              RATE(RS)";
cout<<"\n\n\n  1    Veg Puff            "<<vp;
cout<<"\n\n  2    Egg Puff            "<<ep;
cout<<"\n\n  3    Sandwich            "<<sw;
cout<<"\n\n  4    Veg Roll            "<<vr;
cout<<"\n\n  5    Cheese Pizza        "<<cpz;
cout<<"\n\n  6    Doughnut            "<<don;
cout<<"\n\n  7    Samosa              "<<sam;
cout<<"\n\n  8    Chilli Bun          "<<chib;
cout<<"\n\n  9    Cream Bun           "<<crmb;
cout<<"\n\n  10   Chocolate Fudge      "<<chocof;
}

void account(int x,int money)
{
system("cls");name();
int flag=0;
ifstream file1;
file1.open("name.txt");
char name[10][20];
int admn1[10];
int j=0;
while(!file1.eof())
{
file1>>admn1[j];
file1>>name[j];
j++;
}
do
{
for(int check=0;check<10;check++)
{
if(admn1[check]==x)
{
flag=1;
break;
}
if(check==9)
{
```

```
cout<<"\n\n\n  You have entered the wrong admn no.";
cout<<"\n\n  Enter admission no. : ";
cin>>x;
break;
}
}
}while(flag==0);
system("cls");

for(int i=0;i<10;i++)
{
if(x==admn1[i])
{
//name();
cout<<"\n\n\n\n\t\tWelcome  "<< name[i]<<endl;
break;
}
}
ifstream file2;
file2.open("balance.txt");
int balance[10];
j=0;
while(!file2.eof())
{
file2>>admn1[j];
file2>>balance[j];
j++;
}
file2.close();
std::ofstream f2;
f2.open("balance.txt",std::ofstream::out|std::ofstream::trunc);
f2.close();
for(int k=0;k<10;k++)
{
if(x==admn1[k])
{
balance[k]=balance[k]+money;
cout<<"\n\n\t"<<name[k]<<" your total due amount is Rs. "<<balance[k]<<endl;
cout<<"\n\n\tThis amount has been updated in our database"<<endl;
break;
}

}
ofstream f1;
f1.open("balance.txt");
for(int l=0;l<10;l++)
{
f1<<admn1[l]<<"  ";
f1<<balance[l]<<"\n";
}
cout<<"\n\n";
main1=1;
system("pause");
}

void welcome()
{
```

```
cout<<"\n\n\t\t\t  WELCOME TO BITS AND BYTES";
cout<<"\n\t\t\t  _____";
cout<<"\n\n\n\n\n\n\t\t\t   PROJECT DONE BY \n\n\t\t\t   ABHIJIT AND HARSHIT";
cout<<"\n\t\t\t   *******    *******";
cout<<"\n\n\t\t\t   CLASS XII - B"<<endl;
cout<<"\n\t\t   ";
system("pause");
}

void purchase()
{
setcolor(12);
do
{
system("cls");
name();
menu();
setcolor(12);
cout<<"\n\n\n\n  DO YOU WANT TO PURCHASE(Y/N)... ";
cin>>pu1;
pu=pu1[0];
if(pu!='n'&&pu!='y'&&pu!='Y'&&pu!='N')
{
cout<<"\n  Your choice is wrong"<<endl;
system("pause");
}
}while(pu!='n'&&pu!='y'&&pu!='Y'&&pu!='N');
while(pu=='Y'||pu=='y')
{
system("cls");
name();
menu();
setcolor(12);
char choice1[10];
int check;
do
{
cout<<"\n\n\n  Enter the serial no.(1-10): ";
cin>>choice1;
choice=atoi(choice1);
for(int i=1;i<11;i++)
{
if(choice==i)
{
check=1;
}
}
if(check!=1)
{
cout<<"\n  INVALID SERIAL NO\n\n  PLEASE ENTER VALID SERIAL NO.";
cout<<"\n\n";
system("pause");
system("cls");
name();
menu();
}
}while(check!=1);
```

```
cout<<"\n\n";
switch(choice)
{
case 1: cout<<"  How many Veg Puff : ";
cin>>m;
m=m+m1;
upgradestock(1,m,2);
break;
case 2: cout<<"  How many Egg Puff : ";
cin>>n;
n=n+n1;
upgradestock(2,n,2);
break;
case 3: cout<<"  How many Sandwich : ";
cin>>o;
o=o+o1;
upgradestock(3,o,2);
break;
case 4: cout<<"  How many Veg Roll : ";
cin>>p;
p=p+p1;
upgradestock(4,p,2);
break;
case 5: cout<<"  How many Cheese Pizza : ";
cin>>q;
q=q+q1;
upgradestock(5,q,2);
break;
case 6: cout<<"  How many Doughnut : ";
cin>>r;
r=r+r1;
upgradestock(6,r,2);
break;
case 7: cout<<"  How many Samosa : ";
cin>>s;
s=s+s1;
upgradestock(7,s,2);
break;
case 8: cout<<"  How many Chilli Bun : ";
cin>>t;
t=t+t1;
upgradestock(8,t,2);
break;
case 9: cout<<"  How many Cream Bun : ";
cin>>u;
u=u+u1;
upgradestock(9,u,2);
break;
case 10: cout<<"  How many Chocolate Fudge : ";
cin>>v;
v=v+v1;
upgradestock(10,v,2);
break;
default : cout<<"\n  Wrong Choice";
}
cout<<"\n\n  Do you want to purchase more items(Y/N)...  ";
cin>>pu1;
```

```
366. pu=pu1[0];
367. system("cls");
368. name();
369. m1=m;
370. n1=n;
371. o1=o;
372. p1=p;
373. q1=q;
374. r1=r;
375. s1=s;
376. t1=t;
377. u1=u;
378. v1=v;
379. }
380. cout<<"\n";
381. e=m*vp;
382. f=n*ep;
383. g=o*sw;
384. h=p*vr;
385. k=q*cpz;
386. l=r*don;
387. w=s*sam;
388. x=t*chib;
389. y=u*crmb;
390. z=v*chocof;
391. sum=e+f+g+h+k+l+w+x+y+z;
392. add=m+n+o+p+q+r+s+t+u+v;
393. if(add!=0)
394. {
395. system("cls");
396. name();
397. setcolor(15);
398. cout<<"\n\n\t\t\t\tHUS CANTEEN\n";
399. cout<<"\n\n\t\t\t\t  CASH MEMO\n";
400. cout<<"================================================================================
      ==";
401. cout<<"\nITEMS"          <<"          QUANTITY              PRICE(Rs.)";
402. cout<<"\n================================================================================
      ====";
403. if(m>0)
404. cout<<"\n\nVeg Puff            "<<m<<"                 Rs."<<m*vp;
405. if(n>0)
406. cout<<"\n\nEgg Puff            "<<n<<"                 Rs."<<n*ep;
407. if(o>0)
408. cout<<"\n\nSandwich             "<<o<<"                 Rs."<<o*sw;
409. if(p>0)
410. cout<<"\n\nVeg Roll             "<<p<<"                 Rs."<<p*vr;
411. if(q>0)
412. cout<<"\n\nCheese Pizza         "<<q<<"                 Rs."<<q*cpz;
413. if(r>0)
414. cout<<"\n\nDoughnut             "<<r<<"                 Rs."<<r*don;
415. if(s>0)
416. cout<<"\n\nSamosa               "<<s<<"                 Rs."<<s*sam;
417. if(t>0)
418. cout<<"\n\nChilli Bun           "<<t<<"                 Rs."<<t*chib;
419. if(u>0)
420. cout<<"\n\nCream Bun            "<<u<<"                 Rs."<<u*crmb;
```

```
421. if(v>0)
422. cout<<"\n\nChocolate Fudge      "<<v<<"                  Rs."<<v*chocof;
423. cout<<"\n=============================================================================
     ====";
424. cout<<"\nTOTAL:              "<<add<<"                  Rs."<<sum;
425. m=0,n=0,o=0,p=0,q=0,r=0,s=0,t=0,u=0,v=0;
426. m1=0,n1=0,o1=0,p1=0,q1=0,r1=0,s1=0,t1=0,u1=0,v1=0;//to clear the buffer
427. do
428. {
429. cout<<"\n\n  Do you want to add the bill amount to your account(y/n)...";
430. cin>>pu1;
431. pu=pu1[0];
432. if(pu!='y'&&pu!='Y'&&pu!='n'&&pu!='N')
433. {
434. cout<<"\n\n  Your choice is wrong"<<endl;
435. system("pause");
436. }
437. }while(pu!='y'&&pu!='Y'&&pu!='n'&&pu!='N');
438. if(pu=='y'||pu=='Y')
439. {
440. cout<<"\n\n\n  Enter your Admission No.: ";
441. cin>>admn;
442. account(admn,sum);
443. }
444. else if(pu=='n'||pu=='N')
445. {
446. main1=1;
447. }
448. }
449. main1=1;//if no purchase is done
450. }



451. void thanks()
452. {
453. system("cls");name();
454. cout<<"\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
455. Sleep(500);
456. system("cls");name();
457. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
458. Sleep(500);
459. system("cls");name();
460. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
461. Sleep(500);
462. system("cls");name();
463. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
464. Sleep(500);
465. system("cls");name();
466. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
467. Sleep(500);
468. system("cls");name();
469. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t   THANKS";
470. Sleep(500);
471. system("cls");name();
472. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t
     THANKS";
```

```
473. Sleep(500);
474. system("cls");name();
475. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\t\t\t\t\t
     THANKS";
476. Sleep(500);
477. system("cls");name();
478. cout<<"\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n\n
     \t\t\t\t   THANKS";
479. }
480. void password()
481. {
482. main1=1;
483. ifstream f1;
484. f1.open("login.txt");
485. int loginid[3];
486. char password[3][10];
487. int k=0;
488. while(!f1.eof())
489. {
490. f1>>loginid[k];
491. f1>>password[k];
492. k++;
493. }
494. int login,flag,pos;
495. char login1[10];
496. do
497. {
498. system("cls");
499. name();
500. cout<<"\n\n\n\n\n\n\n\n\t\t\t\tLOGIN ID : ";
501. cin>>login1;
502. login=atoi(login1);
503. for(int i=0;i<3;i++)
504. {
505. if(login==loginid[i])
506. {
507. flag=1;
508. pos=i;
509. }
510. }
511. if(flag!=1)
512. {
513. cout<<"\n\n\t\t\t\tINVALID LOGIN ID.\n\n\t\t\t\tPLEASE TRY AGAIN";
514. cout<<"\n\n\t\t\t\t";
515. system("pause");
516. }

517. }while(flag!=1);
518. char pass[100];
519. do
520. {
521. cout<<"\n\n\t\t\t\tPASSWORD : ";
522. for(int j=0; j<10;j++)
523. {
524. pass[j]=getch();
525. cout<<"*";
```

```
526. if(pass[j]=='\r')//check if enter key is pressed
527. {
528. pass[j]='\0';
529. break;
530. }
531. else if(pass[j]=='\b')
532. {
533. if(j==0)
534. cout<<"\b"<<" "<<"\b";
535. else if(j>=1)
536. {
537. pass[j-1]='\0';//make the previous byte null if backspase is pressed
538. j=j-2;
539. cout<<"\b"<<" "<<"\b\b"<<" "<<"\b";
540. }

541. }
542. }
543. flag=strcmp(pass,password[pos]);
544. if(flag!=0)
545. {
546. cout<<"\n\n\t\t\t\tINVALID PASSWORD.\n\n\t\t\t\tPLEASE TRY AGAIN";
547. cout<<"\n\n\t\t\t\t";
548. system("pause");
549. }
550. }while(flag!=0);
551. }

552. void checkstock()
553. {
554. system("cls");name();
555. int loop4=0;
556. ifstream f5;
557. int st[10],sn[10];
558. f5.open("stock.txt");
559. while(!f5.eof())
560. {
561. f5>>sn[loop4];
562. f5>>st[loop4];
563. loop4++;
564. }
565. cout<<"\n\n\n\t\t";
566. setcolor(14);
567. cout<<"\n Sno.   ITEMS            STOCK";
568. cout<<"\n\n\n  1    Veg Puff  "<<"          "<<st[0];
569. cout<<"\n\n  2    Egg Puff    "<<"        "<<st[1];
570. cout<<"\n\n  3    Sandwich    "<<"        "<<st[2];
571. cout<<"\n\n  4    Veg Roll    "<<"        "<<st[3];
572. cout<<"\n\n  5    Cheese Pizza"<<"        "<<st[4];
573. cout<<"\n\n  6    Doughnut    "<<"        "<<st[5];
574. cout<<"\n\n  7    Samosa      "<<"        "<<st[6];
575. cout<<"\n\n  8    Chilli Bun  "<<"        "<<st[7];
576. cout<<"\n\n  9    Cream Bun   "<<"        "<<st[8];
577. cout<<"\n\n  10   Chocolate Fudge"<<"      "<<st[9];
578. cout<<"\n\n  ";
579. system("pause");
580. main1=1;
```

```
581. }

582. void upgradestock(int check1,int &newamnt,int option)
583. {
584. ifstream f4;
585. int sn[10],loop4=0;
586. char me[10][40];
587. f4.open("menu.txt");
588. while(!f4.eof())
589. {
590. f4>>sn[loop4];
591. f4>>me[loop4];
592. loop4++;
593. }
594. loop4=0;
595. ifstream f5;
596. int st[10];
597. f5.open("stock.txt");
598. while(!f5.eof())
599. {
600. f5>>sn[loop4];
601. f5>>st[loop4];
602. loop4++;
603. }
604. f4.close();
605. f5.close();
606. if(option==2)
607. {
608. for(int i=0;i<10;i++)
609. {
610. if(check1==sn[i])
611. {
612. if(newamnt>st[i])
613. {
614. cout<<"\n\n  SORRY!!!  INSUFFICIENT QUANTITY";
615. cout<<"\n\n  We have only "<<st[i]<<" in our stock"<<"\n\n  ";
616. system("pause");
617. newamnt=0;
618. purchase();
619. break;
620. }
621. st[i]=st[i]-newamnt;
622. break;
623. }
624. }
625. }
626. std::ofstream f6;
627. f6.open("stock.txt",std::ofstream::out|std::ofstream::trunc);
628. f6.close();
629. ofstream f7;
630. f7.open("stock.txt");
631. for(int j=0;j<10;j++)
632. {
633. f7<<sn[j]<<" ";
634. f7<<st[j]<<"\n";
635. }
636. f7.close();
```

```
main1=1;
}

void upgradestockmanual()
{
system("cls");name();
ifstream f9;
int sn1[10],loop5=0;
char me1[10][40];
f9.open("menu.txt");
while(!f9.eof())
{
f9>>sn1[loop5];
f9>>me1[loop5];
loop5++;
}
loop5=0;
ifstream f10;
int st1[10];
f10.open("stock.txt");
while(!f10.eof())
{
f10>>sn1[loop5];
f10>>st1[loop5];
loop5++;
}
f9.close();
f10.close();
int check2,option1,newamnt1,flag;
char option2[100],newamnt[100];
do
{
system("cls");
name();
menu1();
cout<<"\n\n\t\tEnter the serial no of the product : ";
cin>>serial1;
check2=atoi(serial1);
for(int i=0;i<10;i++)
{
if(check2==i+1)
{
flag=1;
cout<<"\n\n\t\tThere are "<<st1[i]<<" "<<me1[i]<<" in stock";
}
}
if(flag!=1)
{
cout<<"\n\n\t\tINVALID SERIAL NO.\n\n\t\tPLEASE TRY AGAIN";
cout<<"\n\n\t\t";
system("pause");
}
}while(flag!=1);
do
{
cout<<"\n\n\t\tThe quantity is to be\n\n\t\t1.ADDED\n\n\t\t2.DEDUCTED ";
cout<<"\n\n\t\t(1/2).... ";
```

```
693. cin>>option2;
694. option1=atoi(option2);
695. if(option1!=1&&option1!=2)
696. {
697. cout<<"\n\n\t\tINVALID OPTION\n\n\t\tPLEASE TRY AGAIN";
698. system("pause");
699. }
700. }while(option1!=1&&option1!=2);
701. do
702. {
703. cout<<"\n\n\t\tEnter the quantity : ";
704. cin>>newamnt;
705. if(isdigit(newamnt[0]))
706. {
707. newamnt1=atoi(newamnt);
708. flag=0;
709. }
710. else
711. {
712. cout<<"\n\n\t\tINVALID QUANTITY\n\n\t\tPLEASE TRY AGAIN";
713. }
714. }while(flag!=0);
715. if(option1==1)
716. {
717. for(int i=0;i<10;i++)
718. {
719. if(check2==sn1[i])
720. {
721. st1[i]=st1[i]+newamnt1;
722. cout<<"\n\n\t\tSTOCK UPDATED!!!!!";
723. cout<<"\n\n\t\tThere are "<<st1[i]<<" "<<me1[i]<<" in stock";
724. cout<<"\n\n\t\t";
725. system("pause");
726. break;
727. }
728. }
729. }
730. if(option1==2)
731. {
732. for(int i=0;i<10;i++)
733. {
734. if(check2==sn1[i])
735. {
736. if(newamnt1>st1[i])
737. {
738. cout<<"\n\n  SORRY!!!  INSUFFICIENT QUANTITY";
739. cout<<"\n\n  We have only "<<st1[i]<<" in our stock"<<"\n\n  ";
740. cout<<"\n\n  GOING TO MAIN MENU";
741. cout<<"\n\n  ";
742. main1=1;
743. system("pause");
744. break;
745. }
746. st1[i]=st1[i]-newamnt1;
747. cout<<"\n\n\t\tSTOCK UPDATED!!!!!!";
748. cout<<"\n\n\t\tThere are "<<st1[i]<<" "<<me1[i]<<" in stock";
749. cout<<"\n\n\t\t";
```

```
750. system("pause");
751. break;
752. }
753. }
754. }
755. std::ofstream f11;
756. f11.open("stock.txt",std::ofstream::out|std::ofstream::trunc);
757. f11.close();
758. ofstream f12;
759. f12.open("stock.txt");
760. for(int j=0;j<10;j++)
761. {
762. f12<<sn1[j]<<" ";
763. f12<<st1[j]<<"\n";
764. }
765. main1=1;
766. }

767. void setcolor(int value)
768. {
769. SetConsoleTextAttribute(GetStdHandle(STD_OUTPUT_HANDLE),  value);
770. }

771. void menu1()
772. {
773. setcolor(14);
774. cout<<"\n Sno.   ITEMS";
775. cout<<"\n\n\n  1    Veg Puff  ";
776. cout<<"\n\n  2    Egg Puff    ";
777. cout<<"\n\n  3    Sandwich    ";
778. cout<<"\n\n  4    Veg Roll    ";
779. cout<<"\n\n  5    Cheese Pizza";
780. cout<<"\n\n  6    Doughnut    ";
781. cout<<"\n\n  7    Samosa      ";
782. cout<<"\n\n  8    Chilli Bun  ";
783. cout<<"\n\n  9    Cream Bun   ";
784. cout<<"\n\n  10   Chocolate Fudge";
785. }


786. void updatebalance(int x,int money)
787. {
788. system("cls");name();
789. int flag=0;
790. ifstream file1;
791. file1.open("name.txt");
792. char name1[10][20];
793. int admn1[10];
794. int j=0;
795. while(!file1.eof())
796. {
797. file1>>admn1[j];
798. file1>>name1[j];
799. j++;
800. }
801. do
802. {
```

```
803. for(int check=0;check<10;check++)
804. {
805. if(admn1[check]==x)
806. {
807. flag=1;
808. break;
809. }
810. if(check==9)
811. {
812. cout<<"\n  You have entered the wrong admn no.";
813. cout<<"\n\n  Enter admission no. : ";
814. cin>>x;
815. break;
816. }
817. }
818. }while(flag==0);
819. system("cls");
820. name();
821. for(int i=0;i<10;i++)
822. {
823. if(x==admn1[i])
824. {
825. //name();
826. cout<<"\n\n\t\tWelcome  "<< name1[i]<<endl;
827. break;
828. }
829. }
830. ifstream file2;
831. file2.open("balance.txt");
832. int balance[10];
833. j=0;
834. while(!file2.eof())
835. {
836. file2>>admn1[j];
837. file2>>balance[j];
838. j++;
839. }
840. file2.close();
841. std::ofstream f2;
842. f2.open("balance.txt",std::ofstream::out|std::ofstream::trunc);
843. f2.close();
844. for(int k=0;k<10;k++)
845. {
846. if(x==admn1[k])
847. {
848. balance[k]=balance[k]-money;
849. cout<<"\n\n\t"<<name1[k]<<" your total due amount is Rs. "<<balance[k]<<endl;
850. cout<<"\n\n\tThis amount has been updated in our database"<<endl;
851. break;
852. }

853. }
854. ofstream f1;
855. f1.open("balance.txt");
856. for(int l=0;l<10;l++)
857. {
858. f1<<admn1[l]<<"  ";
```

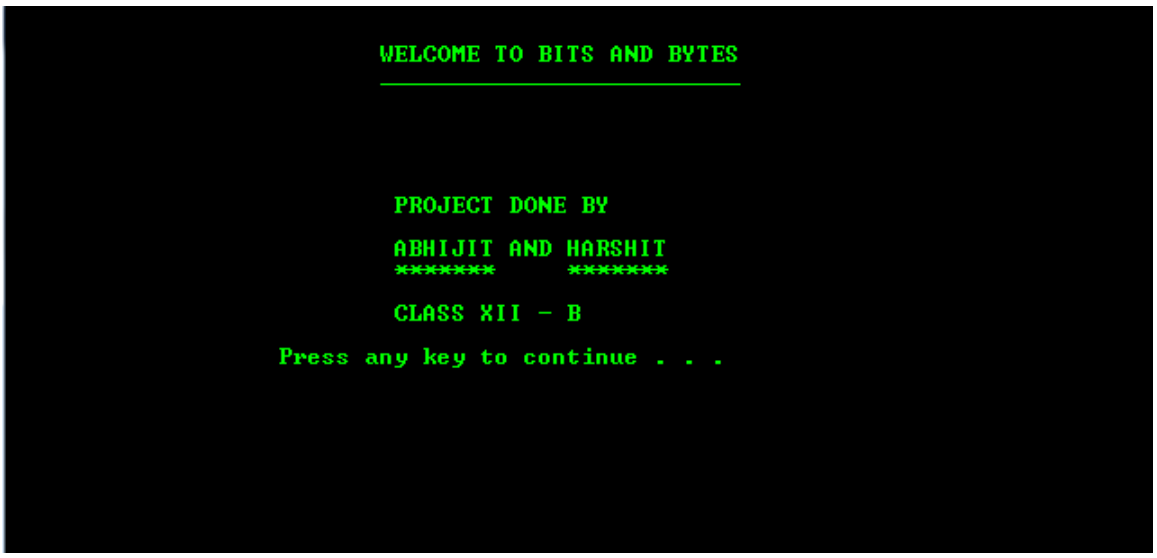
```
f1<<balance[l]<<"\n";
}
cout<<"\n\n";
main1=1;
system("pause");
}

void name()
{
setcolor(12);
cout<<"\n\n\n\t\t\t\tBITS AND BYTES";
cout<<"\n\t\t\t\t_____";
}

void changepassword()
{
ifstream f1;
f1.open("login.txt");
int loginid[3];
char password[3][10];
int k=0;
while(!f1.eof())
{
f1>>loginid[k];
f1>>password[k];
k++;
}
f1.close();
int login,flag,pos;
char login1[10];
do
{
system("cls");
name();
cout<<"\n\n\n\n\n\n\n\n\n\t\t\t\tLOGIN ID : ";
cin>>login1;
login=atoi(login1);
for(int i=0;i<3;i++)
{
if(login==loginid[i])
{
flag=1;
pos=i;
}
}
if(flag!=1)
{
cout<<"\n\n\t\t\t\tINVALID LOGIN ID.\n\n\t\t\t\tPLEASE TRY AGAIN";
cout<<"\n\n\t\t\t\t";
system("pause");
}

}while(flag!=1);
char pass[100];
do
{
cout<<"\n\n\t\t\t\tENTER YOUR OLD PASSWORD : ";
```

```
913. for(int j=0; j<10;j++)
914. {
915. pass[j]=getch();
916. cout<<"*";

917. if(pass[j]=='\r')//check if enter key is pressed
918. {
919. pass[j]='\0';
920. break;
921. }
922. else if(pass[j]=='\b')
923. {
924. if(j==0)
925. cout<<"\b"<<" "<<"\b";
926. else if(j>=1)
927. {
928. pass[j-1]='\0';//make the previous byte null if backspase is pressed
929. j=j-2;
930. cout<<"\b"<<" "<<"\b\b"<<" "<<"\b";
931. }

932. }
933. }
934. flag=strcmp(pass,password[pos]);
935. if(flag!=0)
936. {
937. cout<<"\n\n\t\t\t\tINVALID PASSWORD.\n\n\t\t\t\tPLEASE TRY AGAIN";
938. cout<<"\n\n\t\t\t\t";
939. system("pause");
940. }
941. }while(flag!=0);
942. system("cls");
943. name();
944. char pass1[10],pass2[10];
945. do
946. {
947. cout<<"\n\n\n\n\n\n\n\n\t\t\t\tENTER YOUR NEW PASSWORD : ";
948. for(int j=0; j<10;j++)
949. {
950. pass1[j]=getch();
951. cout<<"*";

952. if(pass1[j]=='\r')//check if enter key is pressed
953. {
954. pass1[j]='\0';
955. break;
956. }
957. else if(pass1[j]=='\b')
958. {
959. if(j==0)
960. cout<<"\b"<<" "<<"\b";
961. else if(j>=1)
962. {
963. pass1[j-1]='\0';//make the previous byte null if backspase is pressed
964. j=j-2;
965. cout<<"\b"<<" "<<"\b\b"<<" "<<"\b";
966. }
```

```
967. }
968. }
969. cout<<"\n\n\t\t\t\tCONFIRM YOUR NEW PASSWORD : ";
970. for(int j=0; j<10;j++)
971. {
972. pass2[j]=getch();
973. cout<<"*";

974. if(pass2[j]=='\r')//check if enter key is pressed
975. {
976. pass2[j]='\0';
977. break;
978. }
979. else if(pass2[j]=='\b')
980. {
981. if(j==0)
982. cout<<"\b"<<" "<<"\b";
983. else if(j>=1)
984. {
985. pass2[j-1]='\0';//make the previous byte null if backspase is pressed
986. j=j-2;
987. cout<<"\b"<<" "<<"\b\b"<<" "<<"\b";
988. }

989. }
990. }
991. flag=strcmp(pass1,pass2);
992. if(flag!=0)
993. {
994. cout<<"\n\n\t\t\t\tPASSWORDS DONT MATCH.\n\n\t\t\t\tPLEASE TRY AGAIN";
995. cout<<"\n\n\t\t\t\t";
996. system("pause");
997. system("cls");
998. name();
999. }
1000.    }while(flag!=0);
1001.    std::ofstream f6;
1002.    f6.open("login.txt",std::ofstream::out|std::ofstream::trunc);
1003.    f6.close();
1004.    ofstream f7;
1005.    f7.open("login.txt");
1006.    for(int j=0;j<3;j++)
1007.    {
1008.    f7<<loginid[j]<<" ";
1009.    if(j==pos)
1010.    f7<<pass1<<"\n";
1011.    else
1012.    f7<<password[j]<<"\n";
1013.    }
1014.    f7.close();
1015.    cout<<"\n\n\t\t\t\tPASSWORDS CHANGED SUCCESSFULLY";
1016.    cout<<"\n\n\n\t\t\t\t";
1017.    system("pause");
1018.    main1=1;
1019.    }
```

OUTPUTS

Welcome Screen



Login Screen

Main Menu

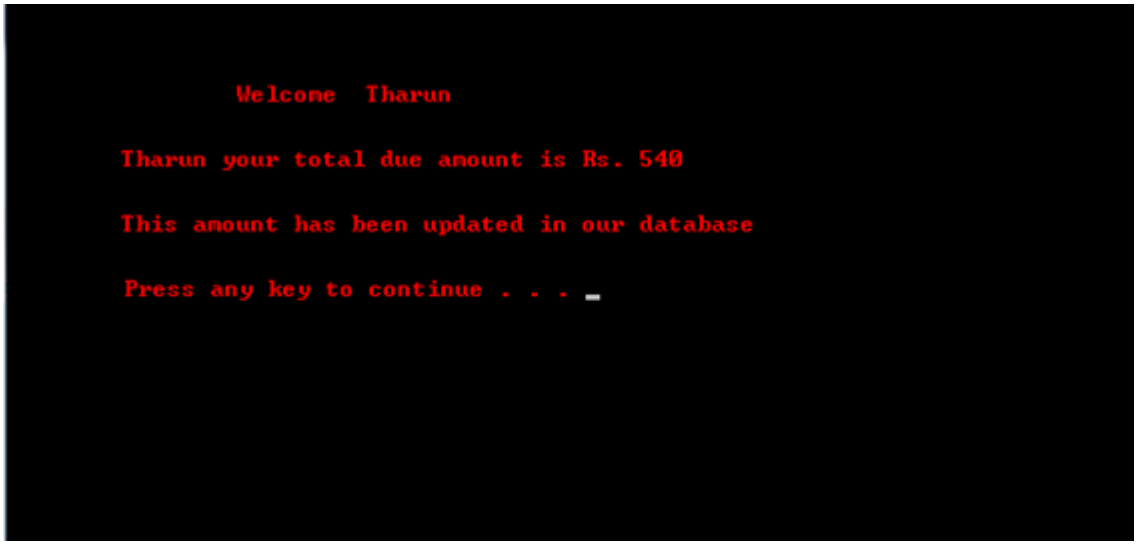
Option 1 - Purchase

```
                          BITS AND BYTES

Sno.    ITEMS                RATE(RS)

 1     Veg Puff               15
 2     Egg Puff               20
 3     Sandwich               30
 4     Veg Roll               25
 5     Cheese Pizza           70
 6     Doughnut               30
 7     Samosa                 10
 8     Chilli Bun             30
 9     Cream Bun              15
 10    Chocolate Fudge        50

 Enter the serial no.(1-10): 5

 How many Cheese Pizza : 6

 Do you want to purchase more items(Y/N)...
```

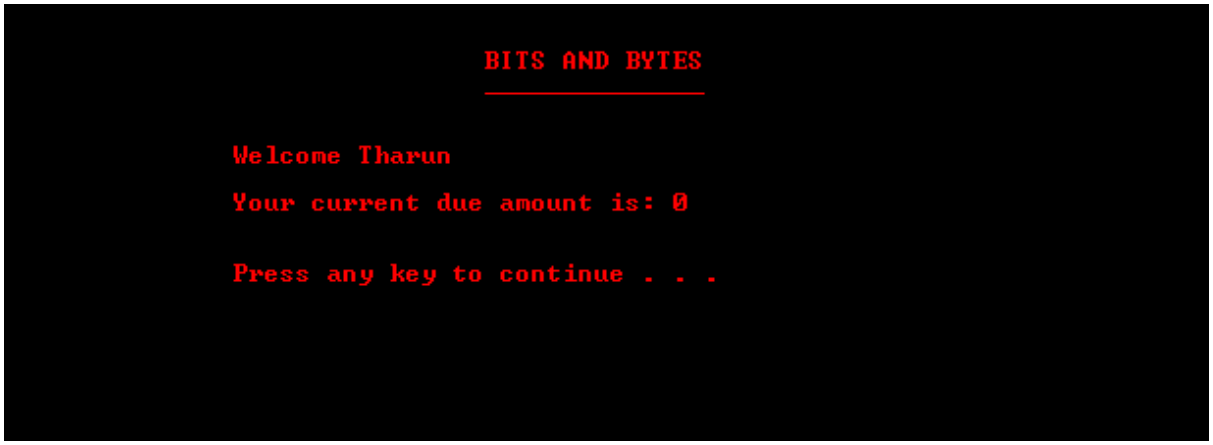
Cash Memo

```
                          BITS AND BYTES

                          HUS CANTEEN

                            CASH MEMO
================================================================================
ITEMS              QUANTITY                  PRICE(Rs.)
================================================================================

Egg Puff              6                       Rs.120

Cheese Pizza          6                       Rs.420
================================================================================
TOTAL:                12                       Rs.540

  Do you want to add the bill amount to your account(y/n)...
```

Customer's Account Details



Option -2



Option – 3

```
                              BITS AND BYTES

 Enter your Admission No.: 103

 Enter the amount paid : 1000
```

```
                              BITS AND BYTES

                 Welcome  Tharun

        Tharun your total due amount is Rs. 1010

        This amount has been updated in our database

Press any key to continue . . .
```

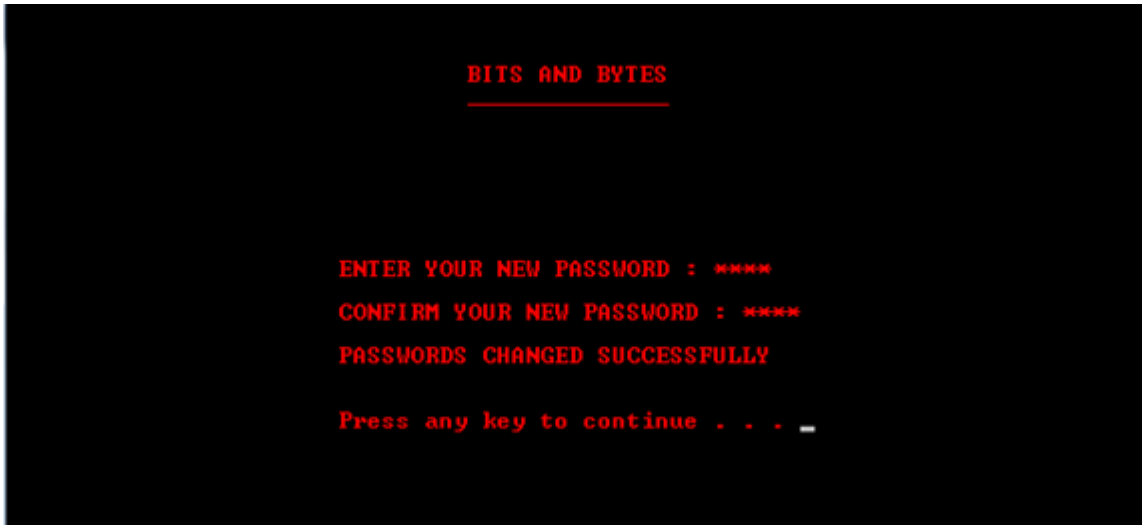
Option - 4

```
                              BITS AND BYTES


Sno.    ITEMS               STOCK

 1      Veg Puff             43
 2      Egg Puff             260
 3      Sandwich             76
 4      Veg Roll             56
 5      Cheese Pizza         66
 6      Doughnut             55
 7      Samosa               47
 8      Chilli Bun           67
 9      Cream Bun            80
 10     Chocolate Fudge      55
 Press any key to continue . . .
```

Option - 5

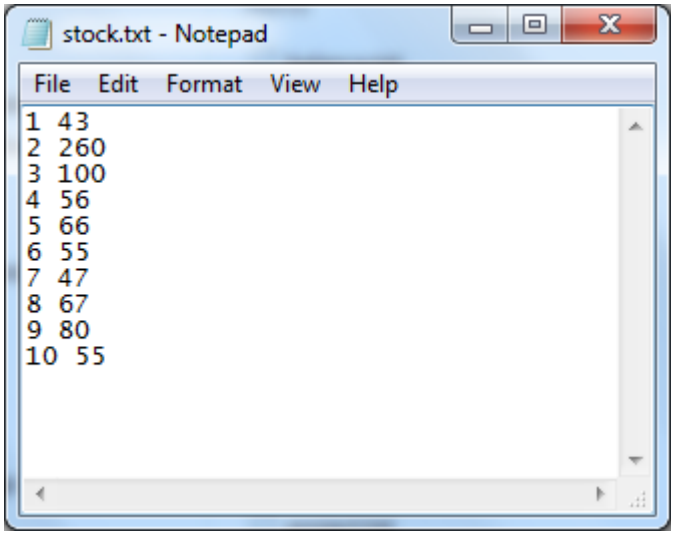
```
                              BITS AND BYTES
Sno.    ITEMS

 1     Veg Puff
 2     Egg Puff
 3     Sandwich
 4     Veg Roll
 5     Cheese Pizza
 6     Doughnut
 7     Samosa
 8     Chilli Bun
 9     Cream Bun
 10    Chocolate Fudge
                Enter the serial no of the product : 3

                There are 76 Sandwich in stock
                The quantity is to be
                1.ADDED
                2.DEDUCTED
                (1/2).... 1

                Enter the quantity : 24

                STOCK UPDATED!!!!!
                There are 100 Sandwich in stock
                Press any key to continue . . .
```

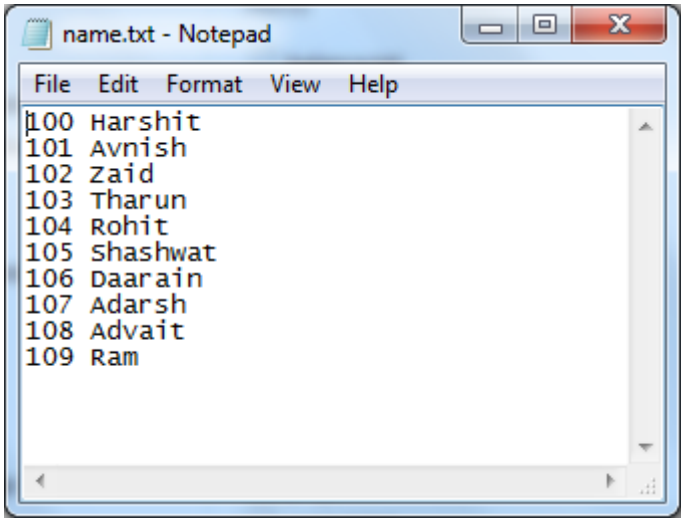
Option - 6



Option - 7

TEXT FILES USED





menu.txt - Notepad

File Edit Format View Help

```
01 VegPuff
02 EggPuff
03 Sandwich
04 VegRoll
05 CheesePizza
06 Doughnut
07 Samosa
08 ChilliBun
09 CreamBun
10 ChocolateFudge
```

login.txt - Notepad

File Edit Format View Help

```
123 hus
231 hus1
321 hus2
```

balance.txt - Notepad

File Edit Format View Help

```
100   0
101   0
102   0
103   0
104   0
105   0
106   0
107   0
108   0
109   0
```

LIMITATIONS OF THE SYSTEM

- Order is not being displayed while the customer is placing a order
- There is no option for cancelling a order
- The quantity of items is fixed in the code
- There is no scope for changing the price of the item

BIBLIOGRAPHY

- Google
- Projectsyapa.com
- Computer science with c++ by sumita arora