```
#include<dos.h>
#include<time.h>
#include<stdio.h>
#include<fstream.h>
#include<math.h>
#include<graphics.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
#include<iomanip.h>

void pg3(void);
void ty(void);
void login(void);
void ig(void);
void graph(int t,float op[],float cp[],int nshares[],int n);
void sgraph(int);
int res[6],finres=0,invcash=0,fincash=0,midx,midy,flag=0,pos;
char cfinresP[180],cfinresN[180],cinvcash[180],cfincash[180],username[50],password[50];

class user
{
    public:
        char name[80];   // username
        char pass[80];   // password
        int invc;                   // invested cash
        int finc;                   // final cash
        int finr;                   // final result
} user1,user2;

struct share
 {
  char  *sn[6][20];         //for share names
  float op[5];              //share open price
  float cp[5];              //share close price
  float gl[5];              //share gain or loss
 } ob1;

float g,p,usd,gbp;

void ig()
{
   int gdriver=DETECT,gmode;
    initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI");
}

void preface()
{
  clrscr();
  ig();
  textcolor(BLACK);
 settextstyle(1,HORIZ_DIR, 3);
 setcolor(12);
 outtextxy(270,0,"NEXTRON");
```

```
 cout<<endl<<endl<<endl;
 cout<<"         Welcome user"<<endl<<" Here at NEXTRON, we help you buy"<<endl<<"  and track your shares , their"<<endl<<" profits and losses along with"<<endl<<" providing  you a detailed analysis"<<endl<<"   of each ups and downs of the market."<<endl<<"     Good to talk-feedback@nexttron.com"<<endl;
 getch();
 closegraph();
}

void arr(void)
{
    int c,n=random(1000);
    n>500?c=1:c=-1;
    g=33000+(c*(n/10));
    p=3000+(c*n);
    n=random(1000);
    n>500?c=1:c=-1;
    usd=70+(c*(n/100));
    gbp=90+(c*(n/100));
}

void commodities(void);
void currencies(void);

int ran(void)
 {
  int c=random(1001);
  return c;
 }

void shares()
 {
  ob1.sn[0][20]="DRAX Pharma";
  ob1.sn[1][20]="EMPIRE estates";
  ob1.sn[2][20]="CIGATI cars";
  ob1.sn[3][20]="QUANBY systems";
  ob1.sn[4][20]="ORFOB apparels";
   int c=ran();
    c>500?c=1:c=-1;
   for(int i=0;i<5;i++)
   {
    ob1.op[i]=300+(c*(ran()/10));
    ob1.cp[i]=300+(c*(ran()/10));
    ob1.gl[i]=10*((ob1.cp[i]-ob1.op[i])/ob1.op[i]);
   }
}

void soc(int a)
 {
  textcolor(BLACK);
  setbkcolor(BLACK);
  for(int j=0;j<3;j++)
  cout<<endl;
   if(a==1)
```

```
    {
      setfillstyle(3,5);
      bar(0,77,700,97);
    }
    else
    {
      setfillstyle(3,9);
      bar(0,47,700,67);
    }
    textcolor(BLACK);
    setbkcolor(BLACK);
    cout<<"SHARES"<<endl<<endl;
  cout<<"Share Name"<<'\t'<<'\t'<<"Open price"<<'\t'<<"Close price"<<endl;
  for(int i=0;i<5;i++)
   {
    cout<<ob1.sn[i][20]<<'\t'<<'\t'<<ob1.op[i]<<'\t'<<'\t'<<ob1.cp[i]<<endl;
   }
 }

 void marquee()
  {
   textcolor(BLACK);
   clrscr();
   for(int i=0;i<5;i++)
     {
      int x=56;
       while(x>0)
    {
      gotoxy(x--,3);
      ob1.gl[i]<0?setfillstyle(10,RED):setfillstyle(10,GREEN);
      bar(0,30,700,50);
      if(ob1.gl[i]<0)
      {
        cout<<ob1.sn[i][20];
        cout<<" \\/ ";
        cout<<setprecision(3)<<fabs(ob1.gl[i])<<"%";
      }
      else
      {
        cout<<ob1.sn[i][20];
        cout<<" /\\ ";
        cout<<setprecision(3)<<fabs(ob1.gl[i])<<"%";
      }
      setcolor(WHITE);
      soc(1);
      commodities();
      currencies();
      delay(10);
      clrscr();
    }
     }
     closegraph();
     ig();
}
```

```
void commodities()
 {
     setfillstyle(3,14);
     bar(0,220,700,240);
     cout<<endl<<"COMMODITIES"<<endl;
     cout<<endl<<endl;
      cout<<"Gold : "<<setprecision(5)<<g<<endl;
     cout<<setprecision(3)<<"Petroleum : "<<fabs(p)<<endl;
 }

void currencies()
{
     setfillstyle(3,11);
     bar(0,317,700,337);
     cout<<endl<<"CURRENCIES"<<endl;
     cout<<endl<<endl;
     cout<<"USD : "<<::usd<<endl;
     cout<<"GBP : "<<::gbp<<endl;
     delay(100);
}

void pg2()
{
 clrscr();
 closegraph();
 ig();
 textcolor(BLACK);
  AGAIN:                    //for reprinting if user enters invalid input
   char ch[40];          //string for user input of buy or name
   soc(0);
   textcolor(BLACK);
   cout<<endl<<endl;
   cout<<"Press BUY to buy shares"<<endl;
   cout<<"Or enter any SHARE NAME to view stats"<<endl;//take user's choice.
   cout<<"Or enter USERSTATS to view your balance amount"<<endl;
   cout<<"Or enter EXIT to leave"<<endl;
   gets(ch);
   int c=0;
   if(strcmpi(ch,"buy")==0)
    {
     c++;
     closegraph();
     ig();
     pg3();
    }

    else if(strcmpi(ch,"userstats")==0)
    {
      c++;
      closegraph();
      cout<<"Current Balance of "<<user1.name<<" : "<<user1.finc;
      getch();
      clrscr();
```

```
      ig();
      goto AGAIN;
    }

    else if(strcmpi(ch,"EXIT")==0)
     {
      c++;
      closegraph();
      ig();
      ty();
     }

    else
      for(int i=0;i<5;i++)
      {
    if(strcmpi(ch,ob1.sn[i][20])==0)
     {
      closegraph();
      ig();
      sgraph(i);
      c++;
     }
      }
    if(c==0)
     {
       clrscr();
       setfillstyle(10,RED);
       bar(0,0,700,700);
       settextstyle(SANS_SERIF_FONT,HORIZ_DIR,5);
       setcolor(YELLOW);
       outtextxy(200,200,"Invalid input");
       delay(1000);
       clrscr();
       textcolor(BLACK);
       setbkcolor(BLACK);
       goto AGAIN;
     }
   getch();
}

void logo()
{
    settextjustify(CENTER_TEXT, CENTER_TEXT);
    textcolor(BLACK);
    for(int i=15;i>0;i--)
    {
        int j=((i%2)!=0?10:12);
        setcolor(j);
        int midx=getmaxx()/2,midy=getmaxy()/2,t=(i==1?1000:pow(i,2.5));
        char title[]="NEXTRON";
        settextstyle(DEFAULT_FONT,HORIZ_DIR,8);
        outtextxy(midx+5,midy,title);
        delay(t);
        clrscr();
```

```
        delay(i==2?1000:t);
    }
}

void pg3(void)
{
 clrscr();
  textcolor(BLACK);//black screen
  int k=3,nos[6],n,tt;
  char str[20];
    soc(0);//print share table
    while(k-->0)//loop to leave spaces
    cout<<endl;
    cout<<"Enter the no. of companies: "<<endl;
    cin>>n;
    for(int i=0;i<n;i++)
    {
        cout<<"Enter the company"<<endl;
        gets(str);
        cout<<"Enter the no of shares"<<endl;
        cin>>nos[i];
    }
    cout<<"enter the track time(in sec)"<<endl;
    cin>>tt;
    cout<<"Share colors :- 1)Magenta,2)Brown,3)Light Grey,4)Dark Grey ,5)Light Blue"<<endl;
 getch();
 closegraph();
 ig();
 graph(tt,ob1.op,ob1.cp,nos,n);
 getch();
 closegraph();
 ig();
 pg2();
}

void sgraph(int p)
{
    int gdriver=DETECT,gmode;
    initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI");
    textcolor(BLACK);
    int h=10,k=400,a,d;     //ORIGIN IS SET AS (int h,int k). Lines are plotted relative to these points.
    int ord[11]={k,k,k,k,k,k,k,k,k,ob1.op[p]};
    for(int b=0;b<9;b++)
    {
     d=ran();
     d>500?d*=1:d*=-1;
     d/=10;
     ord[b]=300+d;
    }
    randomize();
    for(int i=0;i<7;i++)
    {
```

```
        a=400-random(400);

        setbkcolor(BLACK);
        setlinestyle(SOLID_LINE,USERBIT_LINE,2);
        line(h,k,h+600,k); // Y-axis
        line(h,k,h,k-400); // X-axis

        //putpixel(h+150,k,GREEN); // Colourised Blimps on the X and Y axes
        //putpixel(h,k-150,GREEN); // Value of Color depends on SV gain or loss

        struct point{   int x;
                int y;
                }A,B,C,D,E,F,G,H,I,J;

        // Static Points on X-axis

        A.x=h;     A.y=ord[0];
        B.x=h+60;  B.y=ord[1];
        C.x=h+120; C.y=ord[2];
        D.x=h+240; D.y=ord[3];
        E.x=h+300; E.y=ord[4];
        F.x=h+360; F.y=ord[5];
        G.x=h+420; G.y=ord[6];
        H.x=h+480; H.y=ord[7];
        I.x=h+540; I.y=ord[8];
        J.x=h+600; J.y=ord[9];

        // Plotting the graph

        //setcolor(GREEN);
        setlinestyle(USERBIT_LINE,USERBIT_LINE,2);
        for(int b=10;b<=600;b+=25)
            line(b,k,b,k-400);

        setlinestyle(SOLID_LINE,USERBIT_LINE,2);
        line(A.x,A.y,B.x,B.y);
        line(B.x,B.y,C.x,C.y);
        line(C.x,C.y,D.x,D.y);
        line(D.x,D.y,E.x,E.y);
        line(E.x,E.y,F.x,F.y);
        line(F.x,F.y,G.x,G.y);
        line(G.x,G.y,H.x,H.y);
        line(H.x,H.y,I.x,I.y);
        line(I.x,I.y,J.x,J.y);

        // Updating ordinates
        cout<<"                                "<<ob1.sn[p][20]<<" - "<<400-ord[9]<<"   ↵
";
        sleep(2);
        clrscr();
        setbkcolor(BLACK);
        for(int j=0;j<=9;j++) // This is to be moved to the last along with the sleep   ↵
statement
        {
```

```
            ord[j]=ord[j+1];
        }

        ord[9]=a;             //Random ordinate allotted to point J

    }

    getch();
    closegraph();
    ig();
    pg2();
}

void login()
{
    user_input:
    flag=0;
    ig();
    textcolor(BLACK);
    setbkcolor(RED);
    setcolor(GREEN);
    for(int i=0;i<10;i++)
    {   line(0,i,getmaxx(),i);
        line(i,0,i,getmaxy());
        line(getmaxx()-i,0,getmaxx()-i,getmaxy());
        line(0,getmaxy()-i,getmaxx(),getmaxy()-i);
    }
    midx=getmaxx()/2; midy=getmaxy()/2;
    setlinestyle(SOLID_LINE,1,15);
    settextstyle(DEFAULT_FONT,HORIZ_DIR,3);
    setcolor(WHITE);
    line(midx-140,midy-70,midx-140,midy+70);
    line(midx-140,midy-70,midx+140,midy-70);
    line(midx+140,midy-70,midx+140,midy+70);
    line(midx+140,midy+70,midx-140,midy+70);
    setcolor(WHITE);
    outtextxy(midx-55,midy-50,"LOGIN");
    settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
    outtextxy(midx-130,midy+5,"USERNAME: ");//============================
    outtextxy(midx-130,midy+40,"PASSWORD: ");
    //gotoxy(midx-100,midy+5);

    ofstream fout("userdata.txt",ios::app|ios::binary);
    ifstream fin("userdata.txt",ios::in|ios::binary);
    gotoxy(34,16);                                      /*CHANGE!!!*/
    gets(user1.name);
    while(!fin.eof())
    {
        fin.read((char*)&user2,sizeof(user2));
        if(!strcmp(user2.name,user1.name))
        {
            flag=1;
            pos=fin.tellg();
        }
```

```
    }
    if(flag)
    {
        for(int i=0;i<=5;i++)
        {
            if(i==5)
            {
                clrscr();
                cout<<endl<<"Limit of incorrect attempts exceeded! You will now be
locked out of the program";
                getch();
                closegraph();
                exit(0);
            }
            password:
            gotoxy(34,18);                  /*CHANGE!!! (WAS WINDOW)*/
            gets(user1.pass);
            fin.seekg(pos-(sizeof(user2)));
            if(!strcmp(user1.pass,user2.pass))
            {
                clrscr();
                cout<<"ACCESS GRANTED!"<<endl;
                sleep(3);
                user1.finc=user2.finc;
                getch();
            }
            else if(strcmp(user1.pass,user2.pass))
            {
                clrscr();
                cout<<endl<<"Incorrect password! Please enter correct password";
                sleep(3);
                closegraph();
                fin.close();
                fout.close();
                login();
            }
        }
    }

    else if(!flag)
    {
        gotoxy(34,18);                  /*CHANGE!!! (WAS WINDOW)*/
        gets(user1.pass);
        clrscr();
        cout<<"Your account has been created!";
    }

    fout.close();
    fin.close();
    getch();
    closegraph();
}

void graph(int t,float op[],float cp[],int nshares[],int n)
```

```
{
    int gdriver=DETECT,gmode;
    initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BGI");
    textcolor(BLACK);
    int h=10,k=400;
    int ord[6][6]={{0,0,0,0,0,op[0]},{0,0,0,0,0,op[1]},{0,0,0,0,0,op[2]},
{0,0,0,0,0,op[3]},{0,0,0,0,0,op[4]},{0,0,0,0,0,op[5]}};

    for(int i=0;i<t;i++)
    {
        setbkcolor(BLACK);
        setlinestyle(SOLID_LINE,USERBIT_LINE,3);
        setcolor(WHITE);
        line(h,k,h+600,k);
        line(h,k,h,k-400);

        struct point{   int x;
                int y;
                }A,B,C,D,E,F;

         // setlinestyle(USERBIT_LINE,USERBIT_LINE,2);
         // for(int b=10;b<=625;b+=25)
         // line(b,k,b,k-400);

        for(int j=0;j<n;j++)
        {
            int a=random(400);
            ord[j][5]=(i==0?op[j]:a);

             // if(i==(t-1))
              //        ord[j][5]=300+(j*5);

            setcolor(j+5);

            //putpixel(h+150,k,GREEN); // Colourised Blimps on the X and Y axes
            //putpixel(h,k-150,GREEN); // Value of Color depends on SV gain or loss

            A.x=h;     A.y=400-ord[j][0];
            B.x=h+120; B.y=400-ord[j][1];
            C.x=h+240; C.y=400-ord[j][2];
            D.x=h+360; D.y=400-ord[j][3];
            E.x=h+480; E.y=400-ord[j][4];
            F.x=h+600; F.y=400-ord[j][5];

            setlinestyle(SOLID_LINE,USERBIT_LINE,3);
            line(A.x,A.y,B.x,B.y);
            line(B.x,B.y,C.x,C.y);
            line(C.x,C.y,D.x,D.y);
            line(D.x,D.y,E.x,E.y);
            line(E.x,E.y,F.x,F.y);

            //cout<<"                                        "<<ord[9]<<"
";
            //setbkcolor(BLACK);
```

```
            for(int k=0;k<5;k++)
            {
                ord[j][k]=ord[j][k+1];
            }
                    }
        sleep(2);
        if(i!=(t-1))
            clrscr();
    }

int opp[]={250,200,150,300,389,23};
    for(i=0;i<n;i++)
    {
        //res[i]=ord[i][5]-op[i];
        invcash+=(op[i]*nshares[i]);
        fincash+=(cp[i]*nshares[i]);   //should be ord[i][5];
        finres=-(fincash-invcash);
    }

    if(fincash>=invcash)
    {   setcolor(GREEN);
        sprintf(cfinresP,"TOTAL PROFIT: $%d",finres);
        outtextxy(h+435,k+27,cfinresP);
    }
    else if(fincash<=invcash)
    {   setcolor(RED);
        sprintf(cfinresN,"TOTAL LOSS: $%d",finres);
        outtextxy(h+435,k+27,cfinresN);
    }

    line(h,k+10,h+600,k+10);
    line(h,k+10,h,k+50);
    line(h+600,k+10,h+600,k+50);
    line(h,k+50,h+600,k+50);

    settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
    sprintf(cinvcash,"INVESTED CASH: $%d",invcash);
    outtextxy(h+10,k+27,cinvcash);
    line(h+200,k+10,h+200,k+50);


    sprintf(cfincash,"CURRENT BALANCE: $%d",fincash);
    outtextxy(h+210,k+27,cfincash);
    line(h+425,k+10,h+425,k+50);

    user1.invc=invcash;
    user1.finc+=fincash;
    user1.finr=finres;

    ofstream fout("userdata.txt",ios::app|ios::binary);
    fout.write((char*)&user1,sizeof(user1));
    fout.close();

    getch();
```

```
    closegraph();
    clrscr();

 }

void ty()
{
 closegraph();
 ig();
  char ty[20]="THANK YOU";
  clrscr();
   for(int i=0;i<9;i++)
   {
     gotoxy(30+(2*i),14);
     cout<<ty[i];
     delay(500);
   }
 getch();
 exit(1);
}

void main()
{
 clrscr();
 ig();
 randomize();
 getch();
 logo();
 getch();
 preface();
 login();
 ig();
 arr();
 shares();
 marquee();
 pg2();
 closegraph();
}
```