SOURCE CODE

```
#include<conio.h>
#include<graphics.h>
#include<stdio.h>

typedef struct member {
        int id;
        int tf,ts,tr;
        char name[10],password[10];
        char dob[11];
        char gender;
        char answer[10];
        struct status *status;
        struct id *req;
        struct id *friend;
        struct member *next;
} member;

typedef struct status {
        char show[160];
        struct status *next;
} status;

typedef struct id {
        int id;
        struct id *next;
} id_struct;

void save_db();
void initialize();
void main_register();
```

```
void main_login();
void insert_register(struct member *,int);
void main_reset();
void dashboard();
void newsfeed(struct member *);
void member_stats(struct member *);
void add_status(struct member *);
void show_status(struct member *);
void send_req(struct member *);
void add_req(struct member *,int);
void show_req(struct member *);
void viewprofile(struct member *);
void friend_profile(struct member *);
void accept_req(struct member *,int);
void show_friends(struct member *);
void load_db(void);
void wheel(void);
int home_page(void);
void newbar(int,int,int,int);
void skbly7(int,int,char*);
void encrypt(char *);
struct member * search_by_id(int);
struct member * search_by_name(char []);

member *top[27];
int id=1001;
struct member *logined;
int run=0;

void main()
{
        char o[10];
        int gd,gm,i,choice;
        detectgraph(&gd,&gm);
        initgraph(&gd,&gm,"E:\\TC\BGI");
        clrscr();
        strcpy(o,"Shivam");
        encrypt(o);
        //puts(o);
        //getch();
        if(run==0)
        {
                initialize();
            //  wheel();
                load_db();
                run++;
```

```
        }
        //load_db();
        do
        {
    //wheel();
    //outtextxy(30,400,"Press a key to continue....");                    //for loading
    //getch();
        flushall();
        choice=home_page();
        /*      flushall();
                //getch();
                clrscr();
                printf("MAIN MENU :\n1. Register\n2. Login\n3. Forget Password\n4.
Exit\n\nEnter your choice : ");
                scanf("%d",&choice);
        */
                switch(choice)
                {
                        case 1: main_register();
                        break;
                        case 2: main_login();
                        break;
                        case 3: main_reset();
                        break;
                        case 4:
                        break;
                        default : printf("Wrong chice entered...!!\nPlease try again... :)");
                        getch();
                        break;
                }
        } while(choice!=4);
        //save_db();
}

void initialize()
{
        int i;
        for(i=0;i<26;i++)
        {
                top[i]=NULL;
        }
}


void main_register()
{
```

```
        struct member *new_member;
        int i,sucess;
        clrscr();
        flushall();
        setfillstyle(SOLID_FILL,BLACK);
        bar(0,0,700,480);
        setfillstyle(SOLID_FILL,WHITE);
        bar(0,400,700,600);
        setcolor(100);
        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
        outtextxy(100,400,"REGISTER PAGE");
        printf("                WELCOME TO REGISTRATION
PAGE\n_____\
n\n");

        new_member=(member *)malloc(sizeof(member));
        new_member->id=id++;
        printf("Enter your name :");
        gets(new_member->name);
        //new_member->name=string_up(new_member->name);
        printf("Enter your desired password :");
        gets(new_member->password);
        printf("Enter your DOB (DD/MM/YYYY) :");
        gets(new_member->dob);
        printf("Enter your gender (M/F) :");
        scanf("%c",&new_member->gender);
        flushall();
        printf("Security question in case you forget your password will be :\nWhat is your mother
maiden name ?\nEnter your desired answer for it :");
        gets(new_member->answer);
        new_member->ts=0;
        new_member->tf=0;
        new_member->tr=0;
        new_member->next=NULL;
        new_member->status=NULL;
        new_member->req=NULL;
        new_member->friend=NULL;
        sucess=0;
        for(i=65;i<91;i++)
        {       //printf("%c %c\n",i,i+32);
                if(((int)new_member->name[0]==i)||((int)new_member->name[0]==i+32))
                {
                        sucess=1;
                        insert_register(new_member,i-65);
                }
        }
```

```
	if(sucess==0)
	{
		clrscr();
		printf("Hey it seems you entered wrong name....\nPlease take care while entering
it, all the best...!!\n\nLets try again... :)");
		free(new_member);
		getch();
	}
}

void insert_register(struct member *new,int i)
{
	struct member *current_top;
	current_top=top[i];

	if(top[i]==NULL)
	{
		top[i]=new;
		//printf("%s %s %d",top[i]->name,top[i]->password,i);

	}
	else
	{
		while(current_top->next!=NULL)
		{
			current_top=current_top->next;
		}
		current_top->next=new;
	}
}

void main_login()
{
	char name[10],password[10];
	int sucess=0,i,j;
	struct member *temp;
	int login_userr=0;
	int test[100];
	//clrscr();
	flushall();
	setfillstyle(SOLID_FILL,BLACK);
	bar(0,0,700,480);
	setfillstyle(SOLID_FILL,WHITE);
	bar(0,400,700,600);
	setcolor(100);
	settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
```

```
	outtextxy(100,400,"LOGIN PAGE");
	printf("		WELCOME TO OUR LOGIN
PAGE\n_____\
n\n");
	printf("Enter your username : ");
	//gotoxy(30,150);
	gets(name);
	printf("Enter your password :");
	gets(password);
	for(i=65;i<91;i++)
	{
		if((name[0]==i)||(name[0]==i+32))
		{
			j=i-65;
			sucess=1;
		}
	}
	if(sucess==0)
	{
		clrscr();
		printf("Hey it seems you entered wrong name....\nPlease take care while entering
it, all the best...!!\n\nLets try again... :)");
		getch();
	}
	else if(sucess==1)
	{
		sucess=0;
		temp=top[j];
		while(temp!=NULL)
		{
			if(!strcmp(name,temp->name))
			{
				sucess=1;
				if(!strcmp(password,temp->password))
				{
					printf("LOGIN SUCCESSFULL TO YOUR A/C WITH ID
: %d",temp->id);
					logined=temp;
					login_userr=1;
					getch();
				}
			}
		temp=temp->next;
		}
	}
```

```
	if(sucess==0)
	{
		printf("Sorry, we were unable to find your a/c. Please contact the admin at
skbly7@gmail.com");
		getch();
	}

	if(login_userr==1)
	{
		dashboard();
	}

}


void main_reset()
{
	char name[10],answer[10];
	int sucess=0,i,j;
	struct member *temp;
	clrscr();
	flushall();
	setfillstyle(SOLID_FILL,BLACK);
	bar(0,0,700,480);
	setfillstyle(SOLID_FILL,WHITE);
	bar(0,400,700,600);
	setcolor(100);
	settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
	outtextxy(60,400,"PASS RESET PAGE");
	printf("                WELCOME TO PASSWORD RESET
PAGE\n_____\
n\n");
	printf("Enter your username :");
	gets(name);
	printf("Security question :\nWhat is your mother`s Maiden Name : ");
	gets(answer);
	for(i=65;i<91;i++)
	{
		if((name[0]==i)||(name[0]==i+32))
		{
			j=i-65;
			sucess=1;
		}
	}
	if(sucess==0)
	{
```

```
                clrscr();
                setfillstyle(SOLID_FILL,BLACK);
                bar(0,0,700,480);
                setfillstyle(SOLID_FILL,WHITE);
                bar(0,400,700,600);
                setcolor(100);
                settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
                outtextxy(60,400,"PASS RESET PAGE");
                printf("Hey it seems you entered wrong name....\nPlease take care while entering
it, all the best...!!\n\nLets try again... :)");
                getch();
        }
        else if(sucess==1)
        {
                sucess=0;
                temp=top[j];
                while(temp!=NULL)
                {
                        if(!strcmp(name,temp->name))
                        {
                                sucess=1;
                                if(!strcmp(answer,temp->answer))
                                {
                                        clrscr();
                                        setfillstyle(SOLID_FILL,BLACK);
                                        bar(0,0,700,480);
                                        setfillstyle(SOLID_FILL,WHITE);
                                        bar(0,400,700,600);
                                        setcolor(100);
                                        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
                                        outtextxy(60,400,"PASS RESET PAGE");
                                        printf("You provided us the correct answer.\nWell
Done...!!\n\nPlease save this information at some secure place for future.\nYour password is :
%s\nYour account id : %d",temp->password,temp->id);
                                        getch();
                                }
                        }
                temp=temp->next;
                }
        }
        if(sucess==0)
        {
                clrscr();
                setfillstyle(SOLID_FILL,BLACK);
                bar(0,0,700,480);
                setfillstyle(SOLID_FILL,WHITE);
```

```
                bar(0,400,700,600);
                setcolor(100);
                settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
                outtextxy(60,400,"PASS RESET PAGE");
                printf("Sorry, we were unable to find your a/c.\nPlease contact the admin at
skbly7@gmail.com for further help...");
                getch();
        }
}

void dashboard()
{
        int choice;

        do
        {
                getch();
                clrscr();
                setfillstyle(SOLID_FILL,BLACK);
                bar(0,0,700,480);
                setfillstyle(SOLID_FILL,GREEN);
                bar(0,400,700,600);
                setcolor(100);
                settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
                outtextxy(100,400,"DASHBOARD");
                printf("                WELCOME %s, TO YOUR
DASHBOARD\n_____
_____\n\n",logined->name);
                printf("1. Add status\n");
                printf("2. See all your past status\n");
                printf("3. News Feed\n");
                printf("4. View friend`s profile\n");
                printf("5. View your friend list\n");
                printf("6. Add friend\n");
                printf("7. See pending friend request\n");
                printf("8. See your statistics\n");
                printf("9. Logout\n");
                printf("Enter your choice : ");
                scanf("%d",&choice);
                switch(choice)
                {
                case 1: add_status(logined);
                break;
                case 2: show_status(logined);
                break;
                case 3: newsfeed(logined);
```

```
                break;
                case 4: friend_profile(logined);
                break;
                case 5: show_friends(logined);
                break;
                case 6: send_req(logined);
                break;
                case 7: show_req(logined);
                break;
                case 8: viewprofile(logined);
                        //member_stats(logined);
                break;
                case 9:
                break;
                default : printf("Wrong choice... :P\nPlease try again...\nThank you.. ");
                break;
                }
        }while(choice!=9);
        main();
}


void member_stats(struct member *temp)
{
        clrscr();
        flushall();
        printf("Name                    : %s\n",temp->name);
        printf("DOB                     : %s\n",temp->dob);
        printf("Gender          : %c\n",temp->gender);
        printf("Total Friends   : %d\n",temp->tf);
        printf("Pending                 : %d\n",temp->tr);
        printf("Total Status            : %d\n",temp->ts);
        //getch();
}

void add_status(struct member *temp)
{
        struct status *newr,*add;
        //getch();
        clrscr();
        setfillstyle(SOLID_FILL,BLACK);
        bar(0,0,700,480);
        setfillstyle(SOLID_FILL,GREEN);
        bar(0,400,700,600);
        setcolor(100);
        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
```

```
	outtextxy(100,400,"ADD STATUS");

	flushall();
	newr=(status *)malloc(sizeof(status));
	printf("Enter your status : ");
	gets(newr->show);
	newr->next=NULL;
	if(temp->status==NULL)
	{
		temp->status=newr;
	}
	else
	{
		newr->next=temp->status;
		temp->status=newr;
		//add=temp->status;
		//	while(add->next!=NULL)
	  //	{
	  //		add=add->next;
	  ///	}
	  //	add->next=newr;
	}
	temp->ts++;
	printf("Status added successfully.....");
}

void newsfeed(struct member *temp)
{
	struct member *from;
	id_struct *f;
	//getch();
	clrscr();
	setfillstyle(SOLID_FILL,BLACK);
	bar(0,0,700,480);
	setfillstyle(SOLID_FILL,GREEN);
	bar(0,400,700,600);
	setcolor(100);
	settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
	outtextxy(100,400,"NEWS FEED");
	flushall();
	f=temp->friend;
	while(f!=NULL)
	{
		from=search_by_id(f->id);
		if(from->status!=NULL)
		printf("%s says :\n%s\n\n",from->name,from->status->show);
```

```
                f=f->next;
        }
}


void show_status(struct member *temp)
{
        struct status *skbly7;
        skbly7=temp->status;
        while(skbly7!=NULL)
        {
                printf("%s\n",skbly7->show);
                skbly7=skbly7->next;
        }
        //getch();
}

void send_req(struct member *temp)
{
        struct member *send_to;
        int choice,id;
        char name[10],b;
        clrscr();
        setfillstyle(SOLID_FILL,BLACK);
        bar(0,0,700,480);
        setfillstyle(SOLID_FILL,GREEN);
        bar(0,400,700,600);
        setcolor(100);
        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
        outtextxy(100,400,"SEARCH FRIEND");
        printf("1. Search by Name\n");
        printf("2. Search by ID \n");
        printf("Enter your choice : ");
        scanf("%d",&choice);
        if(choice==1)
        {
                flushall();
                printf("Enter friend`s name : ");
                gets(name);
                send_to=search_by_name(name);
        }
        else if(choice==2)
        {       flushall();
                printf("Enter friend`s id : ");
                scanf("%d",&id);
                send_to=search_by_id(id);
```

```
                //getch();
        }
    //  else {
    //  send_to=
    //  }
        if(choice==1||choice==2)
        {
                flushall();
                if(send_to!=NULL)
                {
                        flushall();
                        printf("We have found %s (ID : %d )....",send_to->name,send_to->id);
                        printf("\nPress 'Y' to confirm sending request.....");
                        b=getch();
                        if(b=='Y'||b=='y')
                        {
                        if(send_to->id==temp->id)
                        printf("\nSorry but you cant send friend request to yourself.. ");
                        else
                        add_req(send_to,temp->id);
                        }
                }
        }
}

struct member * search_by_name(char name[])
{
        int i,j,sucess;
        struct member *temp;
        for(i=65;i<91;i++)
        {
                if((name[0]==i)||(name[0]==i+32))
                {
                        j=i-65;
                        sucess=1;
                }
        }
        if(sucess==0)
        {
                clrscr();
                printf("Hey it seems you entered wrong name....\nPlease take care while entering
it, all the best...!!\n\nLets try again... :)");
                return NULL;
                //getch();
        }
        else if(sucess==1)
```

```
        { //    printf("%d %c",j,j);
                sucess=0;
                temp=top[j];
                while(temp!=NULL)
                {
                        if(!strcmp(name,temp->name))
                        {
                                flushall();
                                //printf("We found %s with userid : %d",temp->name,temp->id);
                                //getch();
                                return temp;
                        }
                        temp=temp->next;
                }
        }
        printf("Ohh NO !!!\nNo user with this name found... :(\nTry searching with ID...");
        return NULL;
}


struct member * search_by_id(int id_n)
{
        int j;
        struct member *temp;
        if(id_n>id||id_n<1001)
        {
                printf("Wrong id..!! No user exist with this id...");
                getch();
                return NULL;
        }
        for(j=0;j<26;j++)
        {
                temp=top[j];
                while(temp!=NULL)
                {
                        if(id_n==temp->id)
                        {
                                //printf("We found %s with userid : %d",temp->name,temp->id);
                                //getch();
                                return temp;
                        }
                        temp=temp->next;
                }
        }
        printf("Internal script error.. \nPlease report problem to
skbly7@gmail.com");
```

```
	return NULL;
}

void add_req(struct member *temp,int id)
{
	id_struct *newr;
	//getch();
	flushall();
	newr=(id_struct *)malloc(sizeof(id_struct));
	newr->id=id;
	newr->next=NULL;
	if(temp->req==NULL)
	{
		temp->req=newr;
	}
	else
	{
		newr->next=temp->req;
		temp->req=newr;
	}
	temp->tr++;
	printf("\nRequest sent successfully.....");
}

void show_req(struct member *temp)
{
	id_struct *reqq;
	member *from;
	int accept;
	clrscr();
	setfillstyle(SOLID_FILL,BLACK);
	bar(0,0,700,480);
	setfillstyle(SOLID_FILL,GREEN);
	bar(0,400,700,600);
	setcolor(100);
	settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
	outtextxy(60,400,"PENDING REQUEST");
	reqq=temp->req;
	while(reqq!=NULL)
	{
		from=search_by_id(reqq->id);
		printf("You have request from %s (ID : %d)\n",from->name,from-
>id);
		reqq=reqq->next;
	}
	flushall();
```

```
        printf("\nEnter id of user whose friend request you want to accept :",&accept);
        scanf("%d",&accept);
        reqq=temp->req;
        while(reqq!=NULL)
        {
                if(reqq->id==accept)
                {
                        accept_req(temp,accept);
                        break;
                }
                reqq=reqq->next;
        }
//      accept_req(temp,accept);
}

void accept_req(struct member *temp,int accept)
{
        id_struct *reqq,*reqqq,*new_friend;
        member *from;
        int sucess=0;
        reqq=temp->req;
        reqqq=NULL;
        while(reqq!=NULL)
        {
                if(accept==reqq->id)
                {
                        sucess=1;
                        temp->tf++;
                        temp->tr--;
                        if(reqqq==NULL)
                        {
                                temp->req=reqq->next;
                                if(temp->friend==NULL)
                                {
                                        temp->friend=reqq;
                                        reqq->next=NULL;
                                }
                                else
                                {
                                        reqq->next=temp->friend;
                                        temp->friend=reqq;
                                }
                        }
                        else
                        {
                                reqqq->next=reqq->next;
```

```
                        reqq->next=temp->friend;
                        temp->friend=reqq;
                    }
                    printf("Friend Successfully added... :)");
                }
                reqqq=reqq;
                reqq=reqq->next;
        }
        from=search_by_id(accept);
        if(sucess==1)
        {       new_friend=(id_struct *)malloc(sizeof(id_struct));
                new_friend->id=temp->id;
                new_friend->next=from->friend;
                from->friend=new_friend;
                from->tf++;
        }
}

void show_friends(struct member *temp)
{
        id_struct *skbly7;
        struct member *new;
        skbly7=temp->friend;
        while(skbly7!=NULL)
        {
                new=search_by_id(skbly7->id);
                printf("Friend name : %s\tFriend ID : %d\n",new->name,new->id);
                skbly7=skbly7->next;
        }
        getch();
}

void friend_profile(struct member *temp)
{
        id_struct *skbly7;
        struct member *new;
        int id,sucess=0;
        clrscr();
        setfillstyle(SOLID_FILL,BLACK);
        bar(0,0,700,480);
        setfillstyle(SOLID_FILL,GREEN);
        bar(0,400,700,600);
        setcolor(100);
        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
        outtextxy(60,400,"VIEW PROFILE");
```

```
	skbly7=temp->friend;
	while(skbly7!=NULL)
	{
		new=search_by_id(skbly7->id);
		printf("Friend name : %s\tFriend ID : %d\n",new->name,new->id);
		skbly7=skbly7->next;
	}
	printf("Enter ID of friend whom profile you want to see : ");
	scanf("%d",&id);

	skbly7=temp->friend;
	while(skbly7!=NULL)
	{
		if(id==skbly7->id)
		sucess=1;
		skbly7=skbly7->next;
	}
	if(sucess==1)
	{
		new=search_by_id(id);
		viewprofile(new);
	}
	else
		printf("Wrong ID entered, this user isn`t your friend... ");
}


void encrypt(char *a)
{
	int i;
	for(i=0;a[i]!='\0';i++)
	{
		(*(a+i))=(char)((int)(*(a+i))+30);
	}
}

void save_db()
{
	FILE *fmain,*fmember,*fstatus,*freq,*ffriend;
	int i,j;
	member *temp;
	status *statuss;
	id_struct *req,*friend;
	//fmain here...
	fmember=fopen("MEMBER.DAT","wb");
	fmain=fopen("count.dat","wb");
```

```
	fprintf(fmain,"%d",id);
	fclose(fmain);
	fstatus=fopen("status.dat","wb");
	freq=fopen("req.dat","wb");
	ffriend=fopen("friend.dat","wb");
	//printf("%d\n",id);
	//getch();
	for(i=1001;i<id;i++)
	{
		temp=search_by_id(i);
		fwrite(temp,sizeof(member),1,fmember);
		req=temp->req;
		for(j=0;j<temp->tr;j++)
		{
			fwrite(req,sizeof(id_struct),1,freq);
			req=req->next;
		}
		friend=temp->friend;
		for(j=0;j<temp->tf;j++)
		{
			fwrite(friend,sizeof(id_struct),1,ffriend);
			friend=friend->next;
		}
		statuss=temp->status;
		for(j=0;j<temp->ts;j++)
		{
			fwrite(statuss,sizeof(status),1,fstatus);
			statuss=statuss->next;
		}
	}
	fclose(fmember);
	fclose(fstatus);
	fclose(freq);
	fclose(ffriend);
}

void load_db()
{
	FILE *fmain,*fmember,*fstatus,*freq,*ffriend;
	int i,j;
	member *temp;
	status *statuss;
	id_struct *req,*friend;
	//fmain here..
	fmain=fopen("count.dat","rb");
	fscanf(fmain,"%d",&id);
```

```
//printf("%d",id);
//getch();
fclose(fmain);
fmember=fopen("MEMBER.DAT","rb");
fstatus=fopen("status.dat","rb");
freq=fopen("req.dat","rb");
ffriend=fopen("friend.dat","rb");
//printf("%d\n",id);
//getch();
for(i=1001;i<id;i++)
{
	temp=(member *)malloc(sizeof(member));
	fread(temp,sizeof(member),1,fmember);
	temp->req=NULL;
	temp->friend=NULL;
	temp->status=NULL;

	for(j=0;j<temp->tr;j++)
	{
		req=(id_struct *)malloc(sizeof(id_struct));
		fread(req,sizeof(id_struct),1,freq);
		req->next=NULL;
		if(temp->req==NULL)
		{
			temp->req=req;
		}
		else
		{
			req->next=temp->req;
			temp->req=req;
		}
	}

	for(j=0;j<temp->tf;j++)
	{
		friend=(id_struct *)malloc(sizeof(id_struct));
		fread(friend,sizeof(id_struct),1,ffriend);
		friend->next=NULL;
		if(temp->friend==NULL)
		{
			temp->friend=friend;
		}
		else
		{
			friend->next=temp->friend;
			temp->friend=friend;
```

```
                }
        }

        for(j=0;j<temp->ts;j++)
        {
                statuss=(status *)malloc(sizeof(status));
                fread(statuss,sizeof(status),1,fstatus);
                statuss->next=NULL;
                if(temp->status==NULL)
                {
                        temp->status=statuss;
                }
                else
                {
                        statuss->next=temp->status;
                        temp->status=statuss;
                }
        }

        for(j=65;j<91;j++)
        {
                if(((int)temp->name[0]==j)||((int)temp->name[0]==j+32))
                {
                        //printf("Inserting %s into : %c",temp->name,j);
                        insert_register(temp,j-65);
                        //getch();
                }
        }


    }
    fclose(fmember);
    fclose(fstatus);
    fclose(freq);
    fclose(ffriend);
}

void wheel()
{
    int i;
    outtextxy(260,240,"Loading");
    setfillstyle(SOLID_FILL,BLUE);
    bar(0,0,700,480);
    sound(4000);
    for(i=0;i<20;i++)
    {
```

```
                setfillstyle(SOLID_FILL,CYAN);
                fillellipse(285,240,10*i,10*i);
                outtextxy(260,240,"Loading");
                sound(40*i);
                delay(40);
                setfillstyle(SOLID_FILL,BLUE);
                fillellipse(285,240,5*i,5*i);
                outtextxy(260,240,"Loading");
                delay(40);
        }
        nosound();

}

void newbar(int x,int y,int a, int b)
{
        setfillstyle(SOLID_FILL,BLACK);
        bar(x+5,y+5,a+5,b+5);
        setfillstyle(SOLID_FILL,9);
        bar(x,b+5,a,b+25);
        setfillstyle(SOLID_FILL,WHITE);
        bar(x,y,a,b);
}

void selected(int x,int y,int a, int b)
{
        setfillstyle(SLASH_FILL,WHITE);
        bar(x,y,a,b);
        setfillstyle(SOLID_FILL,YELLOW);
        bar(x,b+5,a,b+22);
        setfillstyle(SOLID_FILL,WHITE);
}

int home_page()
{
        int i=0;
        char a=' ';
        setfillstyle(SOLID_FILL,9);
        bar(0,0,700,480);
        newbar(170,30,450,70);
        setfillstyle(WIDE_DOT_FILL,WHITE);
        bar(170,30,450,70);
        settextstyle(SANS_SERIF_FONT,HORIZ_DIR,2);
        setcolor(255);
        outtextxy(180,35,"WELCOME TO NEW WORLD");
```

```
setcolor(20);
settextstyle(SMALL_FONT,HORIZ_DIR,5);
//ellipse

do
{
        if(a=='a')
        {
                i--;
                if(i<0)
                        i=0;
        }
        if(a=='d')
        {
                i++;
                if(i>3)
                        i=3;
        }
        if(i==0)
        {
                selected(30,200,150,350);
                newbar(180,200,300,350);
                newbar(330,200,450,350);
                newbar(480,200,600,350);
        }
        else
        if(i==1)
        {
                newbar(30,200,150,350);
                selected(180,200,300,350);
                newbar(330,200,450,350);
                newbar(480,200,600,350);
        }
        else
        if(i==2)
        {
                newbar(30,200,150,350);
                newbar(180,200,300,350);
                selected(330,200,450,350);
                newbar(480,200,600,350);
        }
        else
        if(i==3)
        {
                newbar(30,200,150,350);
                newbar(180,200,300,350);
```

```
                        newbar(330,200,450,350);
                        selected(480,200,600,350);
                }
                outtextxy(35,355,"REGISTER");
                outtextxy(185,355,"LOGIN");
                outtextxy(335,355,"FORGET PASS");
                outtextxy(485,355,"EXIT");
                a=getch();
        } while(a=='a'||a=='d');

        return i+1;
}

/*
void skbly7(int x,int y,char * a)
{
int i=0,j=1;
char t[100],c;
while((c=getch())&&j==1)
{
if(((c>64)&&(c<65+27))||((c>95)&&(c<122)))
{
t[i]=c;
t[i+1]='\0';
t++;
//sprintf(,"%s",t);
outtextxy(x,y,t);
i++;
j=1;
}
else
j=0;

}
//printf("Shivam khandelwal");
strcpy(a,t);
}

*/

void viewprofile(struct member * temp)
{
        int i=1;
        char a[3];
        status *s;
        setfillstyle(SOLID_FILL,BLACK);
```

```
        bar(0,0,700,480);
        setfillstyle(SOLID_FILL,RED);
        bar(0,400,700,600);
        setcolor(255);
        settextstyle(TRIPLEX_FONT,HORIZ_DIR,7);
        outtextxy(60,400,"VIEWING PROFILE");
        setfillstyle(SOLID_FILL,220);
        bar(460,60,590,240);
        setfillstyle(SOLID_FILL,0);
        bar(470,70,580,230);
        setfillstyle(SOLID_FILL,220);
        bar(480,80,570,220);
        settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
        outtextxy(490,90,"NO");
        outtextxy(490,110,"IMAGE");
        settextstyle(SANS_SERIF_FONT,HORIZ_DIR,3);
        outtextxy(30,40,temp->name);
        settextstyle(DEFAULT_FONT,HORIZ_DIR,1);
        outtextxy(30,70,"Total Friend");
        itoa(temp->tf,a,10);
        outtextxy(30,90,a);
        outtextxy(30,110,"Total Status");
        itoa(temp->ts,a,10);
        outtextxy(30,130,a);
        s=temp->status;
        outtextxy(170,70,"ALL STATUS");
        while(s!=NULL)
        {
                outtextxy(170,70+(i*20),s->show);
                bar(170,80+(i*20),400,82+(i*20));
                s=s->next;
                i++;
        }
}
```