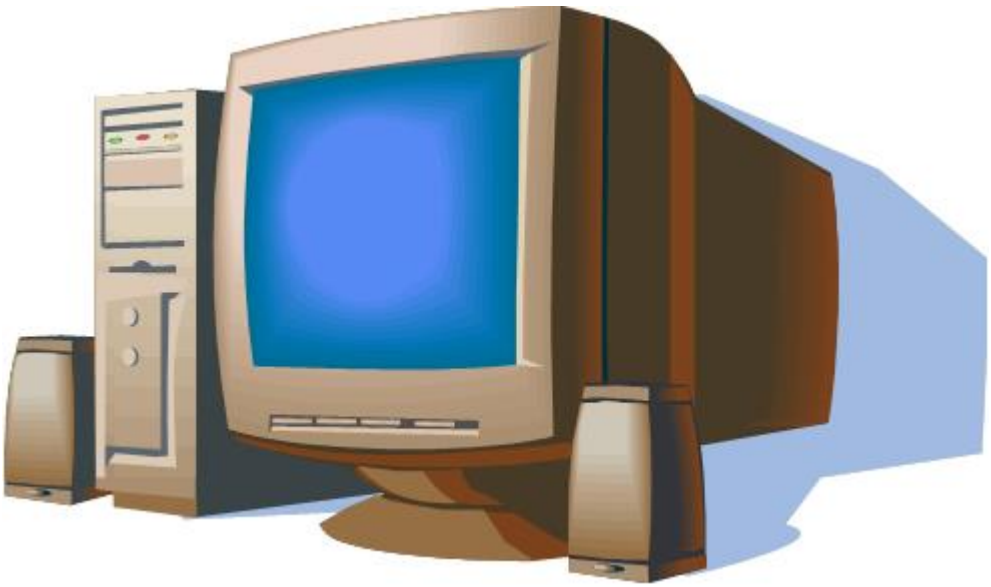
The Hindu Senior Secondary School

Indira Nagar, Adyar, Chennai-600020

COMPUTER SCIENCE PROJECT



TO DO LIST

Done by:-

R.Abhinav, Ram Srikar Putcha and Arindam Rao
XII – A

TABLE OF CONTENTS

The Hindu Senior Secondary School

Indira Nagar, Adyar, Chennai-600020

BONAFIDE CERTIFICATE

This is to certify that this project is a bonafide work of

R.Abhinav, Ram Srikar Putcha and Arindam Rao

(XII-A)

In the Computer Science Laboratory of

The Hindu Senior Secondary School

Indira Nagar, Adyar, Chennai – 600020 (2019 – 20).

Signature

(Internal Examiner)

Signature

(External Examiner)

ACKNOWLEDGEMENT

We would like to express our special thanks of gratitude to our teacher **Mrs. V. Dhanalakshmi** ma'am, our principal **Mrs. Padmini Sriraman** ma'am, as well as our Vice-principal **Mrs. Chandra Chandrashekaran** ma'am who gave us the golden opportunity to do this wonderful project based on the topic **"To-do list"**, which also helped us in doing a lot of research and we came to know about so many new things. We are really thankful to them.

Secondly I would also like to thank my group members for showing excellent cooperation in order to ensure the success of this project.

HEADER FILES USED

- **FSTREAM.H –** For file and I/O Handling
- **PROCESS.H –** For exit function
- **CONIO.H –** For getch, clrscr, cprintf, colors
- **STDIO.H –** For gets
- **STRING.H –** For string handling
- **STDLIB.H –** For atoi( )
- **TIME.H –** For time( ) and ctime( )
- **GRAPHICS.H –** For graphics handling
- **DOS.H –** For delay( )

USER DEFINED HEADER FILES

- **TODOLIST.H -** For To-do list functions
- **EVENT.H –** For declaration and definition of class Event
- **JUNK.H –** For welcome page
- **UI.H –** For menu page
- **DATE.H –** For date processing functions

FLOW OF CONTROL

1. Welcome screen
2. App menu
    2.1. Register
    2.2. Login
        2.2.1. Go to user menu
            2.2.1.1. Add new event
            2.2.1.2. Display all events
            2.2.1.3. Display pending events
            2.2.1.4. Check event
            2.2.1.5. Update event
            2.2.1.6. Display starred events
            2.2.1.7. Star event
            2.2.1.8. Display today deadlines
            2.2.1.9. Display most prioritized event
            2.2.1.10. Clear to-do list
            2.2.1.11. Log out
    2.3. Exit

CODING

DATE.H

```
int i;

char months[12][4] =
{"Jan","Feb","Mar","Apr","May","Jun",
"Jul","Aug","Sep","Oct","Nov","Dec"};

char days[7][4] =
{"Sun","Mon","Tue","Wed","Thu","Fri","Sat"};

int date_comp (char* a1, char* a2)
{
    int result;
    int d, m, y, D, M, Y;
    for(i = 0; i < 12; i++)
    {
        if(a1[4]==months[i][0]&&a1[5]==months
        [i][1]&&a1[6]==months[i][2])
            {
                m = i+1;
                break;
            }
    }
    for(i = 0; i < 12; i++)
    {
        if(a2[4]==months[i][0]&&a2[5]==months
        [i][1]&&a2[6]==months[i][2])
        {
            M = i+1;
            break;
        }
    }
    d = (int(a1[8]) - 48)*10 + (int(a1[9]) -
    48);
```

```
    D = (int(a2[8]) - 48)*10 + (int(a2[9]) -
    48);
    y = (int(a1[11]) - 48)*1000 + (int(a1[12])
    - 48)*100 + (int(a1[13]) - 48)*10 +
    (int(a1[14]) - 48);
    Y = (int(a2[11]) - 48)*1000 + (int(a2[12])
    - 48)*100 + (int(a2[13]) - 48)*10 +
    (int(a2[14]) - 48);
    if (d == D && m == M && y == Y)
        result = 0;
    else if (y > Y || y == Y && m > M || y ==
    Y && m == M && d > D)
        result = 1;
    else result = -1;
    return result;
}

int dayofweek(int d, int m, int y)
{
    static int t[] = { 0, 3, 2, 5, 0, 3,
                5, 1, 4, 6, 2, 4 };
    y -= m < 3;
    return ( y + y / 4 - y / 100 +
         y / 400 + t[m - 1] + d) % 7;
}
int isleap(int year)
{
    int leap;
        if (year % 100 == 0)
        {
            if (year % 400 == 0)
                leap = 1;
            else
                leap = 0;
        }
    else
        {
            if(year % 4 == 0)
                leap = 1;
            else
                leap = 0;
```

```
        }
    return leap;
}
```

EVENT.H

```
#include<fstream.h>
#include<stdio.h>
#include<dos.h>
#include<date.h>
#include<conio.h>
#include<string.h>
#include<stdlib.h>
#include<time.h>
time_t temp;
char ch;
int ptr = 0, ptr1;
char dd[3]={'0','1','\0'};
char mm[3]={'0','1','\0'};
char yy[5]={'2','0','1','9','\0'};
char yy1[5]={'2','0','1','9','\0'};
int up, down;
int num = 0;
int x, y;
char today[30];

class Event
{
    char Type[20], Info[50];
    int check_status, no, pinstatus;
    char start[30], end[30], comp[30];
    void date();
    void userdate();
    public:
    void getdata();
    void putdata();
    char* retend()
```

```
    {
     return end;
    }
    int retno()
    {
     return no;
    }
    int retcheck()
    {
     return check_status;
    }
    int returnpin()
    {
     return pinstatus;
    }
    void update(char choice);
}E;

// MEMBER FUNCTIONS

void Event::date()
{
    strcpy(dd,"28");
    strcpy(mm,"12");
    //YEAR
    yy[0] = start[11];
    yy[1] = start[12];
    yy[2] = start[13];
    yy[3] = start[14];
    up = atoi(yy);
    down = 2099;
    char ch;
    cout<<"\nYear\n";
    cout<<"Put 'w' for scroll up,";
    cout<<" and 's' for scroll down.";
    cout<<"\n\'Enter\' key for selecting
    year\n";
    do
    {
        cout<<yy;
        ch = getch();
```

```
        if(ch == 's')
            if(atoi(yy)<down)
            {
                if(yy[3] != '9')
                    yy[3]++;
                else
                {
                    yy[3] = '0';
                    yy[2]++;
                }
            }
        if(ch == 'w')
            if(atoi(yy)>up)
            {
                if(yy[3] != '0')
                    yy[3]--;
                else
                {
                    yy[3] = '9';
                    yy[2]--;
                }
            }
        if(ch == 13)
            break;
        cout<<'\b';
        cout<<'\b';
        cout<<'\b';
        cout<<'\b';
    }while(1);
//MONTH
    if(atoi(yy) == up)
    {
        i = 0;
        while(i<12)
        {
            if(start[4]==months[i][0]&&start[5
            ]==months[i][1]&&start[6]==months
            [i][2])
            {
                up = i+1;
                break;
```

```
            }
            i++;
        }
    }
    else
        up = 1;
    down = 12;
    cout<<"\nMonth\n";
    cout<<"Put 'w' for scroll up,";
    cout<<" and 's' for scroll down.";
    cout<<"\n\'Enter\' key for selecting
    month\n";
    do
    {
        cout<<mm;
        ch = getch();
        if(ch == 's')
            if(atoi(mm)<down)
            {
                if(mm[1] != '9')
                    mm[1]++;
                else
                {
                    mm[1] = '0';
                    mm[0]++;
                }
            }
        if(ch == 'w')
            if(atoi(mm)>up)
            {
                if(mm[1] != '0')
                    mm[1]--;
                else
                {
                    mm[1] = '9';
                    mm[0]--;
                }
            }
        if(ch == 13)
            break;
        cout<<'\b';
```

```
        cout<<'\b';
    }while(1);
//DATE
    yy1[0] = start[11];
    yy1[1] = start[12];
    yy1[2] = start[13];
    yy1[3] = start[14];
    if(atoi(yy1) == atoi(yy))
    {
        if(atoi(mm) == up)
        {
            dd[0] = start[8];
            dd[1] = start[9];
            up = atoi(dd);
        }
        else up = 1;
    }
    else up = 1;
    x = isleap(atoi(yy));
    y = atoi(mm);
    switch(x)
    {
        case 1: switch(y)
        {
            case 2: down = 29; break;
            case 1: case 3: case 5: case 7:
            down = 31; break;
            case 8: case 10: case 12: down =
            31; break;
            default: down = 30;
        }break;
        case 0: switch(y)
        {
            case 2: down = 28; break;
            case 1: case 3: case 5: case 7:
            down = 31; break;
            case 8: case 10: case 12: down =
            31; break;
            default: down = 30;
        }break;
    }
```

```
    cout<<"\nDate\n";
    cout<<"Put 'w' for scroll up,";
    cout<<" and 's' for scroll down.";
    cout<<"\n\'Enter\' key for selecting
    date\n";
    do
    {
        cout<<dd;
        ch = getch();
        if(ch == 's')
        {
            if(atoi(dd)<down)
            {
                if(dd[1] != '9')
                    dd[1]++;
                else
                {
                    dd[1] = '0';
                    dd[0]++;
                }
            }
        }
        if(ch == 'w')
            if(atoi(dd)>up)
            {
                if(dd[1] != '0')
                    dd[1]--;
                else
                {
                    dd[1] = '9';
                    dd[0]--;
                }
            }
        if(ch == 13)
            break;
        cout<<'\b';
        cout<<'\b';
    }while(1);
}

void Event::userdate()
```

```
{
     temp = time(NULL);
     strcpy(end,ctime(&temp));
     end[8] = dd[0]; end[9] = dd[1];
     x = dayofweek(atoi(dd), atoi(mm),
     atoi(yy));
     end[0] = days[x][0];
     end[1] = days[x][1];
     end[2] = days[x][2];
     y = (atoi(mm) - 1);
     end[4] = months[y][0];
     end[5] = months[y][1];
     end[6] = months[y][2];
     end[11] = yy[0];
     end[12] = yy[1];
     end[13] = yy[2];
     end[14] = yy[3];
     end[15] = '\0';
}
void Event::update(char choice)
{
      if(choice == '1')
      {
         cprintf("Enter new description:");
         gets(Info);
      }
      else if(choice == '2')
      {
         cprintf("Enter new title:");
         gets(Type);
      }
      else if(choice == '3')
      {
         cprintf("New deadline date (dd mm
         yyyy)\n");
         date();
         userdate();
      }
      else if(choice == 'e')
      {
         temp = time(NULL);
```

```
            strcpy(comp,ctime(&temp));
            comp[11] = comp[20];
            comp[12] = comp[21];
            comp[13] = comp[22];
            comp[14] = comp[23];
            comp[15] = '\0';
            check_status = 1;
         }
         else if(choice == 'h')
            pinstatus = 1;
}
void Event::getdata()
{
    cout<<"Event "<<num<<"\n";
    cout<<"------------------------\n";
    no = num;
    cout<<"Enter event Title:";
    gets(Type);
    cout<<"Enter event Description:";
    gets(Info);
    temp = time(NULL);
    strcpy(start,ctime(&temp));
    start[11] = start[20];
    start[12] = start[21];
    start[13] = start[22];
    start[14] = start[23];
    start[15] = '\0';
    cout<<"\nDeadline date\n";
    date();
    userdate();
    check_status = 0;
    pinstatus = 0;
    clrscr();
    cout<<"\n\nEvent Entered.";
}
void Event::putdata()
{
    cout<<"Event "<<no<<"."<<Type;
    if(returnpin()==1)
        cout<<" (STARRED)";
    cout<<"\n";
```

```
    cout<<"-------------------------\n";
    cout<<Info;
    cout<<"\nRegistration date/time:\n\t";
    cout<<start;
    cout<<'\n';
    cout<<"\nDeadline date/time:\n\t";
    cout<<end;
    cout<<"\n\n";
    temp = time(NULL);
    strcpy(today,ctime(&temp));
    today[11] = today[20];
    today[12] = today[21];
    today[13] = today[22];
    today[14] = today[23];
    today[15] = '\0';
    if(check_status == 0)
    {
        cout<<"Incomplete\n";
        if(date_comp(today,end)>0)
            cout<<"Deadline already over\n";
    }
    else
    {
        cout<<"Completed on ";
        cout<<comp;
        if(date_comp(end,comp)<0)
            cout<<"\nLate completion\n";
    }
}
```

TODOLIST.H

```
#include<event.h>
#include<string.h>
#include<ctype.h>
#include<process.h>
#include<junk.h>
fstream fin;
char b;
char str[20] = {'\0'};
```

```
int n;
int gm, gd = DETECT;

//TODOLIST_FUNCTIONS

void update1()
{   char i = 'N';
    fin.open(str,ios::in|ios::out|ios::binary)
    ;
    fin.seekg(0);
    int pos = 0;
    while(fin.read((char*)&E,sizeof(E)))
    {
        if(n == E.retno())
        {
            fin.seekp(pos);
            i='Y';
            if(E.retcheck() == 1 && ch == 'e')
            {   i = 'C';
                break;
            }
            if(E.returnpin() == 1 && ch ==
            'h')
            {
                i = 'P';
                break;
            }
            E.update(ch);
            fin.write((char*)&E,sizeof(E));
            break;
        }
        pos=fin.tellg();
    }
    fin.close();
    if(i=='N')
        cout<<"\nEvent number not found.";
    else if(i=='C')
        cout<<"\nAlready checked";
    else if(i=='P')
        cout<<"\nAlready pinned";
    else
```

```
        cout<<"\nDone.";
    cout<<"\n\nB. Back to menu";
}
void pinevent()
{
    cout<<"Enter no to pin event:";
    cin>>n;
    ch='h';
    update1();
}
void dispallevent()
{
    closegraph();
   fin.open(str,ios::in|ios::out|ios::binary);
    fin.seekg(0);
    while(fin.read((char*)&E,sizeof(E)))
    {  if(E.returnpin())
       {
            cout<<"E v e n t   d e t a i l
            s\n";
            cout<<"-------------------------
            \n\n";
            E.putdata();
            getch();
            clrscr();
        }
    }
    fin.clear();
    fin.seekg(0);
    while(fin.read((char*)&E,sizeof(E)))
    {
        if(!E.returnpin())
        {
            cout<<"E v e n t   d e t a i l
            s\n";
            cout<<"-------------------------
            \n\n";
            E.putdata();
            getch();
            clrscr();
        }
```

```
    }

    cprintf("End of events\n\n");
    cprintf("B. Back to menu");
    fin.close();
}
void dispallevent(char c)
{
   fin.open(str,ios::in|ios::out|ios::binary);
   fin.seekg(0);
   if(c=='c')
   {
        while(fin.read((char*)&E,sizeof(E)))
        {
            cout<<"C o m p l e t e d   E v e n
            t s\n";
            cout<<"---------------------------
            \n\n";
            if(E.retcheck() == 1)
            {
                E.putdata();
                getch();
            }
        clrscr();
        }
        fin.close();
  }
  else if(c=='g')
  {
        while(fin.read((char*)&E,sizeof(E)))
        {
            cout<<"S t a r r e d   e v e n t
            s\n";
            cout<<"---------------------------
            \n\n";
            if(E.returnpin() == 1)
            {
                E.putdata();
                getch();
            }
            clrscr();
```

```
        }
        fin.close();
    }
    else if(c == 'i')
    {
        temp = time(NULL);
        strcpy(today,ctime(&temp));
        today[11] = today[20];
        today[12] = today[21];
        today[13] = today[22];
        today[14] = today[23];
        today[15] = '\0';
        while(fin.read((char*)&E,sizeof(E)))
        {
            cout<<"T o d a y    d e a d l i n
            e s\n";
            cout<<"-------------------------
            ---\n\n";
            if(date_comp(E.retend(), today) ==
            0)
            {
                E.putdata();
                getch();
            }
            clrscr();
        }
        fin.close();
}
else if(c=='j')
{
   Event E1;
   fin.read((char*)&E1,sizeof(E1));
   fin.seekg(0);
   while(fin.read((char*)&E,sizeof(E)))
   {
     if(date_comp(E.retend(),E1.retend())>0)
       E = E1;
   }
   clrscr();

   fin.clear();
```

```
    fin.seekg(0);
    while(fin.read((char*)&E1,sizeof(E1)))
    {
        cout<<"M o s t   P r i o r i t i z e
        d\n";
        cout<<"------------------------------
        \n\n";
        if(!date_comp(E.retend(),E1.retend()))
        {
            E1.putdata();
            getch();
        }
        clrscr();
    }
    fin.close();
 }
 else
 {
    while(fin.read((char*)&E,sizeof(E)))
    {
        cout<<"P e n d i n g   E v e n t s\n";
        cout<<"--------------------------
        \n\n";
        if(E.retcheck() == 0)
        {
            E.putdata();
            getch();
        }
        clrscr();
    }
    fin.close();
 }
cprintf("End of events.\n\n");
cprintf("B. Back to menu");
}
void addnewevent()
{
    fin.open(str,ios::in|ios::out|ios::app|ios
    ::binary);
    if(num == 0)
        num++;
```

```
    else if(num != 0)
        fin.seekg(sizeof(E)*(num-1));
    while(fin.read((char*)&E,sizeof(E)))
        num++;
    fin.close();

    fin.open(str,ios::in|ios::out|ios::app|ios
    ::binary);
    clrscr();
    cout<<"E v e n t   d e t a i l s\n";
    cout<<"-------------------------\n\n";
    E.getdata();
    fin.write((char*)&E,sizeof(E));
    cout<<"\n\nB. Back to menu";
    fin.close();
}
void clearallevent()
{
    num = 1;
    cprintf("To-do list cleared.\n\n");
    cprintf("B. Back to menu");
    remove(str);
    fin.open(str,ios::in|ios::out|ios::binary)
    ;
    fin.close();
}
void updateevent()//to complete
{
    menu:
    clrscr();
    cout<<"Do you want to update/Edit by\n";
    cout<<"1. Description\n";
    cout<<"2. Title\n";
    cout<<"3. Deadline date\n";
    ch = getch();
    if(ch == '1' || ch == '2'||ch=='3')
    {
        clrscr();
        cout<<"Enter no to update event:";
        cin>>n;
        update1();
```

```
    }
    else
    {
        cout<<"\nInvalid key pressed";
        delay(500);
        goto menu;
    }
}
void checkevent()
{
    clrscr();
    cout<<"Enter no to check event:";
    cin>>n;
    ch = 'e';
    update1();
}
```

UI.H

```
#include<todolist.h>
fstream acc;
int fl;
void menu();
void disp_menu();
char name[20], n1[20];
//ACCOUNT FUNCTIONS
void createnewaccount()
{
    clrscr();
    acc.open("Accounts.txt",ios::in|ios::app|i
    os::out);
    gotoxy(25,10);
    cout<<"Enter your name:";
    cin>>n1;
    acc.seekg(0);
    fl = 0;
    while(acc>>name)
    {
        if(strcmp(n1,name) == 0)
        {
```

```
                strcpy(str,strcat(n1,".dat"));
                fl = 1;
            }
        }
        acc.close();
    }
    void login()
    {
        str[0] = '\0';
        acc.open("Accounts.txt",
        ios::in|ios::app|ios::out);
        clrscr();
        gotoxy(25,10);
        cout<<"Enter registered name:";
        cin>>n1;
        acc.seekg(0);
        fl = 0;
        while(acc>>name)
        {
            if(strcmp(n1,name) == 0)
            {
                strcpy(str,n1);
                strcat(str,".dat");
                fl = 1;
            }
        }
        acc.close();
    }
    //MENU
    void disp_menu()
    {
        clrscr();
        initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
        setbkcolor(WHITE);
        setcolor(BLUE);
        outtextxy(20,70,"Welcome,");
        outtextxy(20,90,n1);
        outtextxy(20,150,"---------------");
        outtextxy(20,160,"U s e r   M e n u");
        outtextxy(20,170,"---------------");
        outtextxy(20,200,"a.  Add new event");
```

```
outtextxy(20,220,"b.  Display all
events");
outtextxy(20,240,"c.  Display completed
events");
outtextxy(20,260,"d.  Display pending
events");
outtextxy(20,280,"e.  Check event");
outtextxy(20,300,"f.  Update event");
outtextxy(20,320,"g.  Display starred
events");
outtextxy(20,340,"h.  Star event");
outtextxy(350,200,"i.  Display today
deadlines");
outtextxy(350,220,"j.  Display most
prioritized event");
outtextxy(350,240,"k.  Clear to-do list");
outtextxy(350,260,"l.  Log out");
ch = getch();
closegraph();
ch = tolower(ch);
if(ch == 'k')
{
    clrscr();
    clearallevent();
    b = getch();
    if(b == 'b'||b == 'B')
        disp_menu();
}
else if(ch == 'a')
{
    clrscr();
    addnewevent();
    b = getch();
    if(b == 'b'||b == 'B')
    disp_menu();
}
else if(ch == 'b')
{
    clrscr();
    dispallevent();
    ch = getch();
```

```
    if(ch == 'b' ||ch == 'B')
        disp_menu();
}
else
if(ch=='c'||ch=='d'||ch=='g'||ch=='i'||ch=
=='j')
{
    clrscr();
    dispallevent(ch);
    ch = getch();
    if(ch == 'b' ||ch == 'B')
        disp_menu();
}
else if(ch == 'e')
{
    clrscr();
    checkevent();
    b = getch();
    if(b == 'b'||b == 'B')
        disp_menu();
}
else if(ch == 'f')
{
    clrscr();
    updateevent();
    b = getch();
    if(b == 'b'||b == 'B')
        disp_menu();
}
else if(ch == 'h')
{
    clrscr();
    pinevent();
    b = getch();
    if(b == 'b'||b == 'B')
        disp_menu();
}
else if(ch == 'l')
    menu();
else
    disp_menu();
```

```
}
void menu()
{
     clrscr();
     initgraph(&gd, &gm, "C:\\TURBOC3\\BGI");
     setbkcolor(WHITE);
     setcolor(BLUE);
     outtextxy(20,40,"Welcome to our to-do list
     cum event manager.");
     outtextxy(20,60,"Here is our best attempt
     to design an efficient to-do list ");
     outtextxy(20,80,"for managing your daily
     life activities.");
     outtextxy(20,100,"Check this out1!");
     outtextxy(20,150,"---------------");
     outtextxy(20,160,"A p p   M e n u");
     outtextxy(20,170,"---------------");
     outtextxy(20,200,"a.  Register");
     outtextxy(20,220,"b.  Login");
     outtextxy(20,240,"c.  Exit");
     ch = getch();
     closegraph();
     if(ch == 'a')
     {
          createnewaccount();
          if(fl == 1)
          {
               gotoxy(25,11);
               cout<<"Account already r
               egistered.";
               getch();
               menu();
               fl = 0;
          }
          else
          {
               gotoxy(25,11);
               cout<<"Done.";
               acc.open("Accounts.txt",ios::in|io
               s::out|ios::app);
               acc<<n1<<' ';
```

```
            acc.close();
            getch();
            menu();
        }
    }
    else if(ch == 'b')
    {
        login();
        if(fl == 0)
        {
            gotoxy(28,12);
            cout<<"Account does not exist";
            getch();
            menu();
        }
        else
        {
            load();
            disp_menu();
            fl = 0;
        }
    }
    else if(ch == 'c')
    {
        initgraph(&gd, &gm,
        "C:\\TURBOC3\\BGI");
        setbkcolor(WHITE);
        setcolor(BLUE);
        settextstyle(8,0,20);
        outtextxy(20,20,"BYE");
        settextstyle(0,0,1);
        outtextxy(400,400,"Project done by:-
        ");
        outtextxy(400,420,"R. Abhinav");
        outtextxy(400,440,"Ram Srikar");
        outtextxy(400,460,"Arindam Rao");
        delay(2000);
        closegraph();
        exit(0);
    }
    else
```

```
        menu();

}
```

JUNK.H

```
// Source: https://planet-source-code.com/vb/scripts/ShowCode.asp?txtCodeId=453&lngWId=3
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<dos.h>
#include<graphics.h>
int DEL = 10;
     void draw_bar(void);
     void draw_bar3d(void);
     void draw_line(void);
     void draw_pieslice(void);
void draw_bar(void)
{
    int i;
    setfillstyle(CLOSE_DOT_FILL,BLUE);
    for(i=0;i<=640;i+=10)
    {
        bar(0,0,i,50);
        delay(DEL/2);
        sound(200+i);
        delay(DEL);
        nosound();
    }
    for(i=50;i<=480;i+=10)
    {
        bar(590,50,640,i);
        delay(DEL/2);
        sound(400+i);
        delay(DEL);
        nosound();
```

```
    }
    for(i=590;i>=0;i-=10)
    {
         bar(590,430,i,480);
         delay(DEL/2);
         sound(600+i);
         delay(DEL);
         nosound();
    }
    for(i=430;i>=50;i-=10)
    {
        bar(0,430,50,i);
        delay(DEL/2);
        sound(800+i);
        delay(DEL);
        nosound();
    }
    for(i=70;i<=140;i+=10)
    {
        setfillstyle(SOLID_FILL,BLUE);
        bar(430,70,440,140);
        delay(DEL/2);
        sound(1000+i);
        delay(DEL);
        nosound();
    }
    for(i=70;i<=140;i+=10)
    {
            bar(460,70,470,140);
            delay(DEL/2);
            sound(1000+i);
            delay(DEL);
            nosound();
    }
}
void draw_line(void)
{
    setfillstyle(CLOSE_DOT_FILL,BLUE);
    int i;
    for(i=70;i<=140;i+=10)
    {
```

```
        line(70,70,80,i);
        delay(DEL/2);
        sound(1000+i);
        delay(DEL);
        nosound();
    }
    for(i=140;i>=90;i-=10)
    {
        line(80,140,100,i);
        delay(DEL/2);
        sound(1000+i);
        delay(DEL);
        nosound();
    }
    for(i=90;i<=140;i+=10)
    {
        line(100,90,120,i);
        delay(DEL/2);
        sound(1000+i);
        delay(DEL);
        nosound();
    }
    for(i=140;i>=70;i-=10)
    {
        line(120,140,130,i);
        delay(DEL/2);
        sound(1000+i);
        delay(DEL);
        nosound();
    }
    for(i=70;i<=105;i+=5)
    {
       line(440,70,450,i);
       delay(DEL/2);
       sound(1000+i);
       delay(DEL);
       nosound();
    }
    for(i=105;i>=70;i-=5)
    {
           line(450,105,460,i);
```

```
                delay(DEL/2);
                sound(1000+i);
                delay(DEL);
                nosound();
        }
     }
void draw_bar3d(void)
{
        int i;
        setfillstyle(CLOSE_DOT_FILL, BLUE);
        for(i=0;i<=5;i++)
        {
            bar3d(150,70,170,140,5,3);
            delay(DEL);
            bar3d(170,70,200,90,5,3);
            delay(DEL);
            bar3d(170,95,190,115,5,3);
            delay(DEL);
            bar3d(170,120,200,140,5,3);
            delay(DEL);
            sound(1000+i);
            delay(DEL);
            nosound();
            bar3d(220,70,240,140,i,3);
            delay(DEL);
            bar3d(240,120,280,140,i,3);
            delay(DEL);
            bar3d(300,70,320,140,i,2);
            delay(DEL);
            bar3d(320,70,350,90,i,3);
            delay(DEL);
            bar3d(320,120,350,140,i,2);
            delay(DEL);
            bar3d(480,70,500,140,i,3);
            delay(DEL);
            bar3d(500,70,530,90,i,2);
            delay(DEL);
            bar3d(500,95,520,115,i,2);
            delay(DEL);
            bar3d(500,120,530,140,i,2);
            delay(DEL);
```

```
      }
}
void draw_pieslice(void)
{
     int i;
     setfillstyle(CLOSE_DOT_FILL,BLUE);
     for(i=5;i<=30;i++)
     {
         pieslice(390,105,0,360,i);
         delay(DEL);
         setfillstyle(EMPTY_FILL,YELLOW);
         pieslice(390,105,0,360,15);
         sound(1000+i);
         delay(DEL);
         nosound();
     }
     for(i=40;i<=480;i+=40)
     {
         setfillstyle(CLOSE_DOT_FILL,CYAN);
         pieslice(60+i,240,0,360,40);
         delay(DEL);
       setfillstyle(CLOSE_DOT_FILL,LIGHTGREEN);
         pieslice(60+i,320,0,360,40);
         delay(DEL);
         sound(1000+i);
         delay(DEL);
         nosound();
     }
}
void call()
{
     int k,i;
     int gdriver = DETECT, gmode, errorcode;
     initgraph(&gdriver, &gmode,
     "C:\\TURBOC3\\BGI");
     setbkcolor(BROWN);
     setcolor(BLUE);
     draw_bar();
     draw_line();
     draw_bar3d();
     draw_pieslice();
```

```
    getch();
    closegraph();
}
void load()
{
    clrscr();
    int x=170,i,gdriver=DETECT,gmode;
    initgraph(&gdriver,&gmode,"C:\\TURBOC3\\BG
    I");
    setbkcolor(BROWN);
    setcolor(BLUE);
    settextstyle(DEFAULT_FONT,HORIZ_DIR,2);
    outtextxy(170,180,"LOADING ACCOUNT");
    rectangle(170,200,470,220);
    for(i=0;i<300;++i)
    {
        delay(DEL/2);
        line(x,200,x,220);
        x++;
    }
    closegraph();
}
```

MAIN PROGRAM

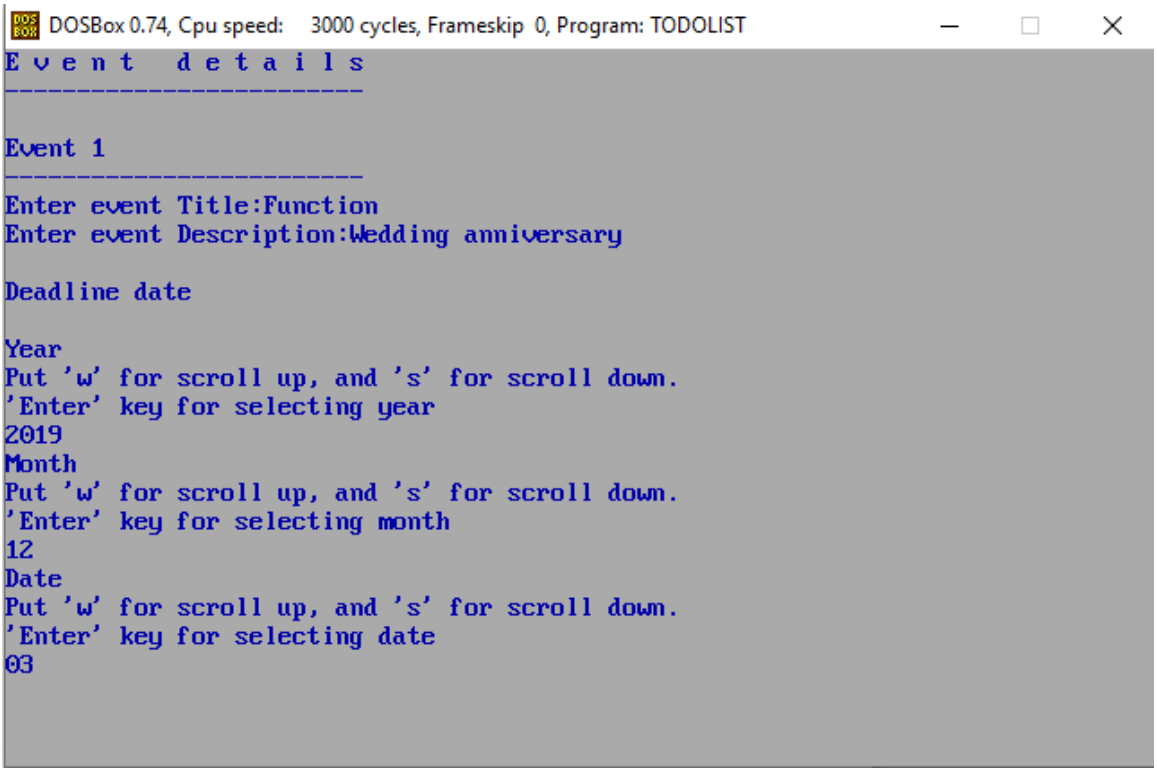
```
#include<ui.h>
void main()
{
    textcolor(1);
    textbackground(15);
    clrscr();
    call();
    menu();
}
```

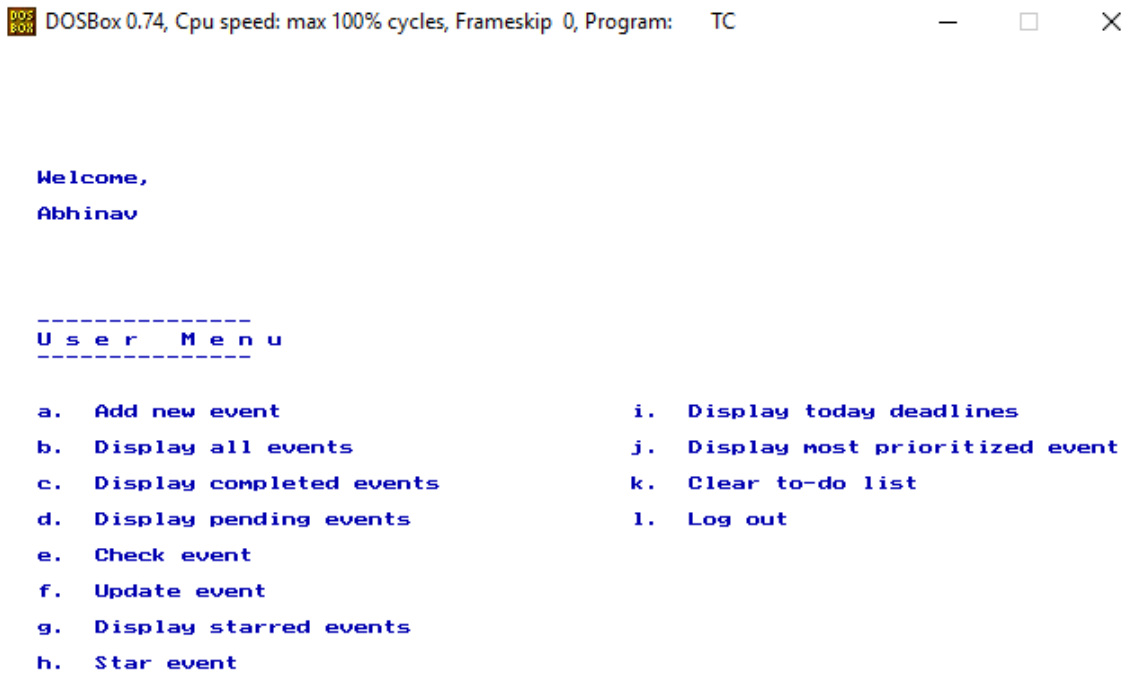
SCREEN SHOTS

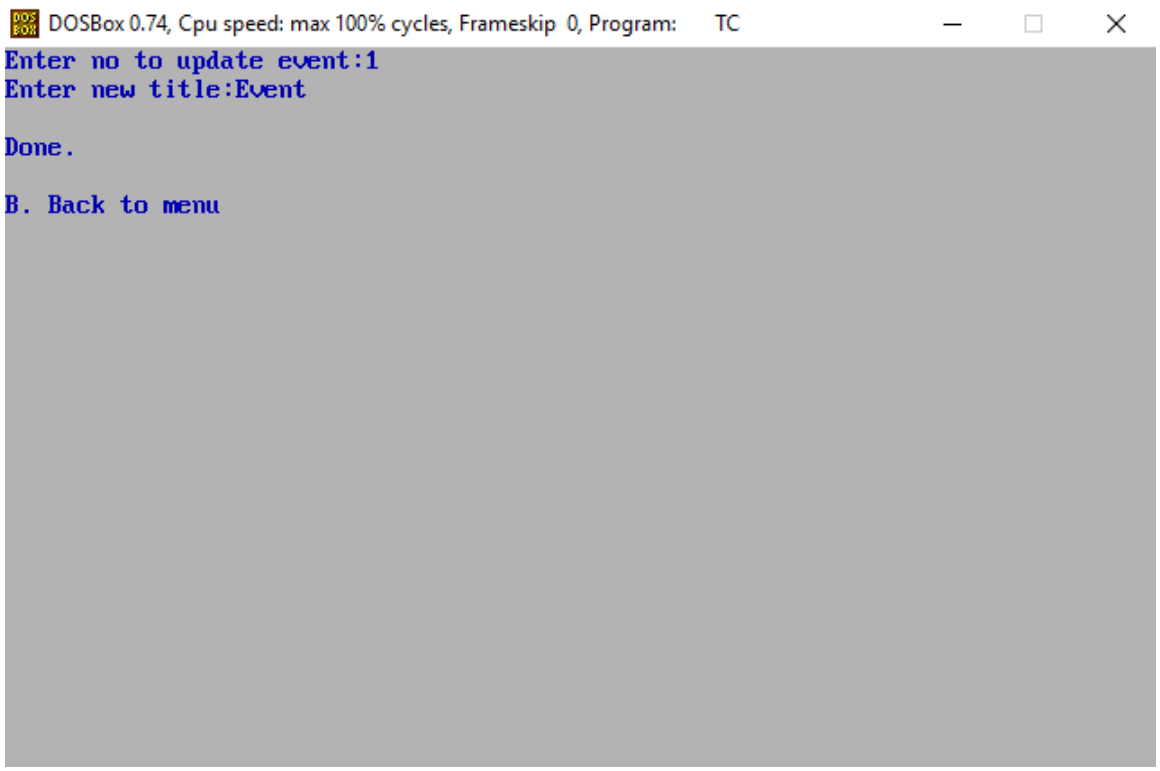
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: TODOLIST

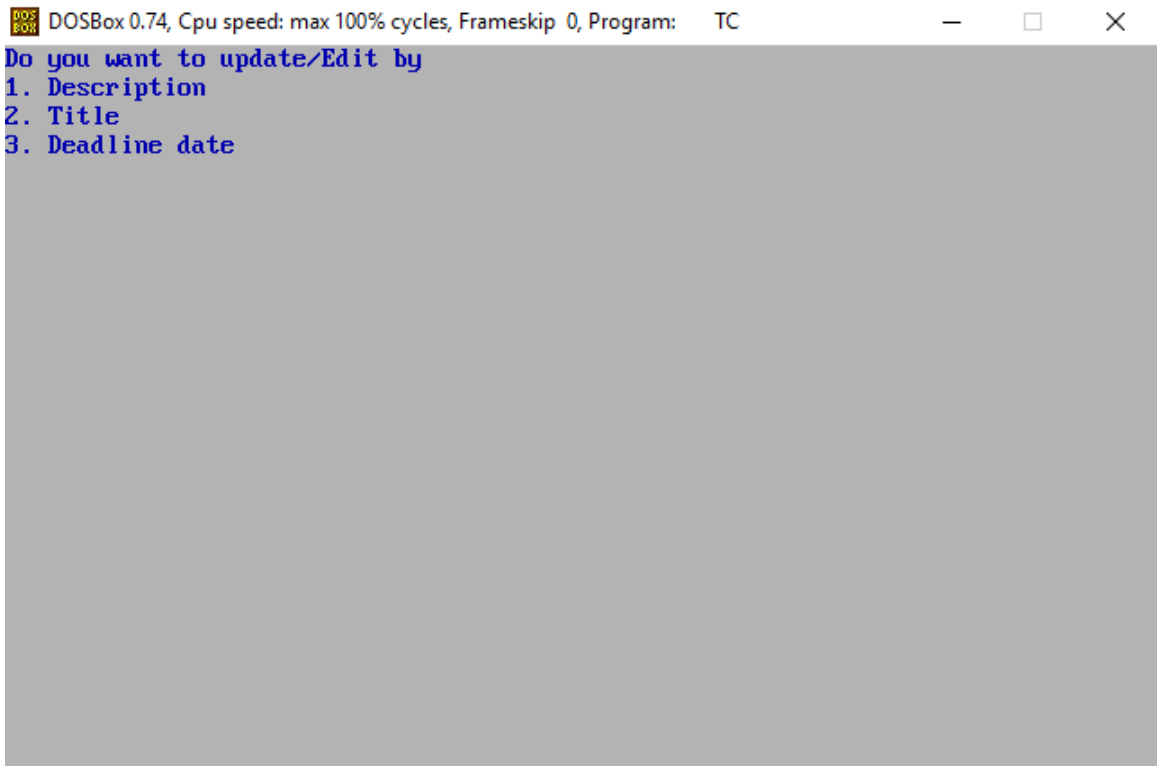
DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: TODOLIST

```
Welcome to our to-do list cum event manager.
Here is our best attempt to design an efficient to-do list
for managing your daily life activities.
Check this out!


---------------
A p p   M e n u
---------------

a.  Register
b.  Login
c.  Exit
```

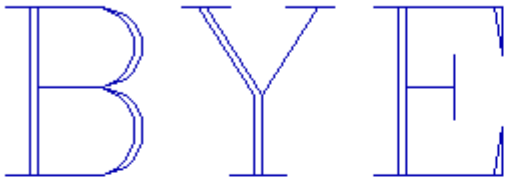

DOSBox 0.74, Cpu speed: 3000 cycles, Frameskip 0, Program: TODOLIST
E v e n t d e t a i l s
Event 1
Enter event Title:Function
Enter event Description:Wedding anniversary
Deadline date
Year
Put 'w' for scroll up, and 's' for scroll down.
'Enter' key for selecting year
2019
Month
Put 'w' for scroll up, and 's' for scroll down.
'Enter' key for selecting month
12
Date
Put 'w' for scroll up, and 's' for scroll down.
'Enter' key for selecting date
03


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
Welcome,
Abhinav
U s e r M e n u
a. Add new event
b. Display all events
c. Display completed events
d. Display pending events
e. Check event
f. Update event
g. Display starred events
h. Star event
i. Display today deadlines
j. Display most prioritized event
k. Clear to-do list
l. Log out


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
Enter no to update event:1
Enter new title:Event
Done.
B. Back to menu


DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC
Do you want to update/Edit by
1. Description
2. Title
3. Deadline date

DOSBox 0.74, Cpu speed: max 100% cycles, Frameskip 0, Program: TC



Project done by:-
R. Abhinav
Ram Srikar
Arindam Rao

LIMITATIONS

- Improper implementation of time
- Can be more user friendly
-

REQUIREMENTS

- Turbo C++ application to edit and compile the program
- Microsoft word to copy the compiled code
-

BIBLIOGRAPHY

COMPUTER SCIENCE IN C++ BY: – SUMITA ARORA