TIME TABLE MANAGEMENT

By –

Aishwarya Athani

Arundhati Rajan

Palasha Rana

Shravya Sharan

SYNOPSIS

This project has been implemented using the programming language, C++. This program employs object-oriented programming, file handling techniques and functions.

The objective of this project is to generate a revised time table for school teachers. This user-friendly menu-driven program contains previously stored data of the time table for the teachers as well as the classes. When the user enters the required day, it first displays the original time tables for both the teachers and the classes.

Next, it is designed to modify the time tables of teachers and allot substitution periods in case another teacher is absent. This allotment is done on a priority basis, with the teacher of the same subject as the absentee teacher getting favoured. Also, it is ensured that no teacher gets more than six teaching periods in a day. The program also allows the user to change the time periods of different classes. These changes then get reflected in the teachers' schedule and are marked by an asterisk. Teachers can then, at ease, view their updated time tables for that particular day. The changes are transitory and the schedule is reset to the original after use.

The purpose of this program is to optimise the process of time table allocation. It reduces human error, ensures efficient management of records, and is easy to use.

CODE

```
#include <iostream.h>

#include <fstream.h>

#include <stdlib.h>

#include <string.h>

#include <iomanip.h>

#include <conio.h>


#define MAX_CLASSES 6

#define MAX_DAYS 5

#define MAX_PERIODS 8

#define MAX_TEACHERS 10


char today[5][5] = { "Mon", "Tue", "Wed", "Thu", "Fri"};


class TimeTable

{

        public:

        char  subject[5], subst;

        int classCd, teacherCd;
```

```
    void initTimeTable(int clCd, int tchCd, char* sub, char substChr)

    {

        classCd=clCd;

        teacherCd=tchCd;

        strcpy(subject, sub);

        subst = substChr;

    }


    intisFreeSlot()

      {

    if (classCd==99) // class code 99 indicates free period for tch

        return 1;

    else

    return 0;

    }

};

class Teacher

{

    public:

    int teacherCd;

    char teacherName[10];
```

```
        int isAbsent[MAX_DAYS], load[MAX_DAYS],       myClass,
othClass[3];
        char mySubject[5];
        TimeTable timeTable[MAX_DAYS][MAX_PERIODS];

        void markAbsent(intdow) { isAbsent[dow] = 1; }
        int isMySubject(char* sub)
           {
                if(strcmp(mySubject, sub) == 0)
                return 1;
                else
                return 0;
            }
} teacher[MAX_TEACHERS];
classClassRoom
{
        public:
        int classCd;
        char className[5];
        TimeTable timeTable[MAX_DAYS][MAX_PERIODS];
} classRoom[MAX_CLASSES];
```

```
TimeTable freeSlot, absentSlot;

void initClassRooms()
{
        ifstream classFile ("c:\\ttbl\\classes.dat");
        for(int i=0; i<MAX_CLASSES; i++)
           {
                classFile>>classRoom[i].classCd;
                classFile>>classRoom[i].className;
        }
classFile.close();
}

//Pass Class Code when Class Name is supplied, 99 if none
int findClass(char clName[5] )
{
        for(int i=0; i<MAX_CLASSES; i++)
                if(strcmp(classRoom[i].className, clName)==0)
                      {
                        return classRoom[i].classCd;
}
```

```
    return 99;
}

void readClassRooms()
{
    char sub[5];
    int tchCd=-1;

    ifstream classFile ("c:\\ttbl\\classrm.dat");

    //Read Time Table from File

    for(int dow=0; dow<MAX_DAYS; dow++)

    for(int pd=0; pd<MAX_PERIODS;pd++)

        for(int classCd=0; classCd<MAX_CLASSES; classCd++)
           {
        classFile>> sub;

    classRoom[classCd].timeTable[dow][pd].initTimeTable(classCd,tchCd,sub,’ ‘);
```

```
        }

    classFile.close();
}


void readTeachers()
{
    int i, j;
    int ctCnt, clCnt;
    char clName[5], sub[5];

    ifstream teacherFile("c:\\ttbl\\teachers.dat");

    //Read Teachers from File for each teacher in file

    for(i=0; i<MAX_TEACHERS; i++)
      {
    teacherFile>> teacher[i].teacherCd>>
    teacher[i].teacherName>>
    teacher[i].mySubject>>
```

```
ctCnt>>clCnt;

for(j=0; j<ctCnt; j++)// Class Teacher

     {

         teacherFile>>clName;

         teacher[i].myClass=findClass(clName);

  }

int ctr=0;

for(j=0; j<clCnt; j++)  // Classes taught

    {

         teacherFile>>clName;

         teacher[i].othClass[ctr++]=findClass(clName);

  }


for(int k=0; k<MAX_DAYS; k++) // Init Matrix

    {

         teacher[i].load[k]=0;

         teacher[i].isAbsent[k] = 0; //Present

         for(int j=0; j<MAX_PERIODS; j++)

         teacher[i].timeTable[k][j]=freeSlot;

  }

}
```

```
        teacherFile.close();

}


int findTeacher(int clCd, char sub[5])

{

        for(int i=0; i<MAX_TEACHERS;i++)

          {

                if(strcmp(sub, "EXAM")==0 &&
(teacher[i].myClass==clCd))

                        return teacher[i].teacherCd;

                else

                if(strcmp(sub, teacher[i].mySubject)==0 &&
(teacher[i].othClass[0]==clCd ||

                        teacher[i].othClass[1]==clCd ||

                        teacher[i].othClass[2]==clCd))

                        return teacher[i].teacherCd;

        }

        return -1;

}


//Given Class Time Table and Teacher assign teacher in ClassMatrix
```

```
void assignTeacher()

{

    Int tchCd;

    for(int dow=0; dow<MAX_DAYS; dow++)

    for(int pd=0; pd<MAX_PERIODS;pd++)

        for(int classCd=0; classCd<MAX_CLASSES; classCd++)

              {

        tchCd = findTeacher(classRoom[classCd].classCd,

            classRoom[classCd].timeTable[dow][pd].subject);

        classRoom[classCd].timeTable[dow][pd].teacherCd =
tchCd;

        teacher[tchCd].timeTable[dow][pd]=

            classRoom[classCd].timeTable[dow][pd];

        teacher[tchCd].load[dow]++;

        }

}


int exactSubs(TimeTablett, intdow, int pd)

{

    for(int i=0; i<MAX_TEACHERS; i++)

    if(!teacher[i].isAbsent[dow] &&
        teacher[i].isMySubject(tt.subject)  &&
        teacher[i].timeTable[dow][pd].isFreeSlot() &&
        teacher[i].load[dow]<6)
         {
            tt.teacherCd = teacher[i].teacherCd;
            tt.subst='*';
            teacher[i].timeTable[dow][pd]=tt;
            teacher[i].load[dow]++;
            return 1;
         }
    return 0;
}

void genSubs(TimeTablett, intdow, int pd)
{
    for(int i=0; i<MAX_TEACHERS; i++)
    if(tt.subst != '*' && !teacher[i].isAbsent[dow] &&
        teacher[i].timeTable[dow][pd].isFreeSlot() &&
        teacher[i].load[dow]<6)
         {
            tt.teacherCd = teacher[i].teacherCd;
            tt.subst='*';

            teacher[i].timeTable[dow][pd]=tt;

            teacher[i].load[dow]++;

        }

}


void reassignTeachers(intdow)

{

    for(int i=0; i<MAX_TEACHERS; i++)

        if(teacher[i].isAbsent[dow])

        for(int pd=0; pd<MAX_PERIODS; pd++)

            if(!teacher[i].timeTable[dow][pd].isFreeSlot())

                if(!exactSubs(teacher[i].timeTable[dow][pd],
dow, pd))

                genSubs(teacher[i].timeTable[dow][pd], dow,
pd);

}


void drawLine(char c, intsz)

{

    for(int i=0; i<sz; i++)

        cout<< c;
```

```
}

void displayClassTable(intdow)
{
    int i;
    drawLine(205, 79);
    cout<< endl << "\t";
    for(i=0; i<MAX_PERIODS; i++)
        cout<< "\t"<< i+1;
    cout<< endl;
    drawLine(205, 79);
    for(i=0; i<MAX_CLASSES; i++)
      {
        cout<<setw(10) <<endl<<classRoom[i].className;
        for(int pd=0; pd<MAX_PERIODS;pd++)
            cout<< "\t"
<<classRoom[i].timeTable[dow][pd].subject;
    }
    cout<< endl;
    drawLine(205, 79);
}
```

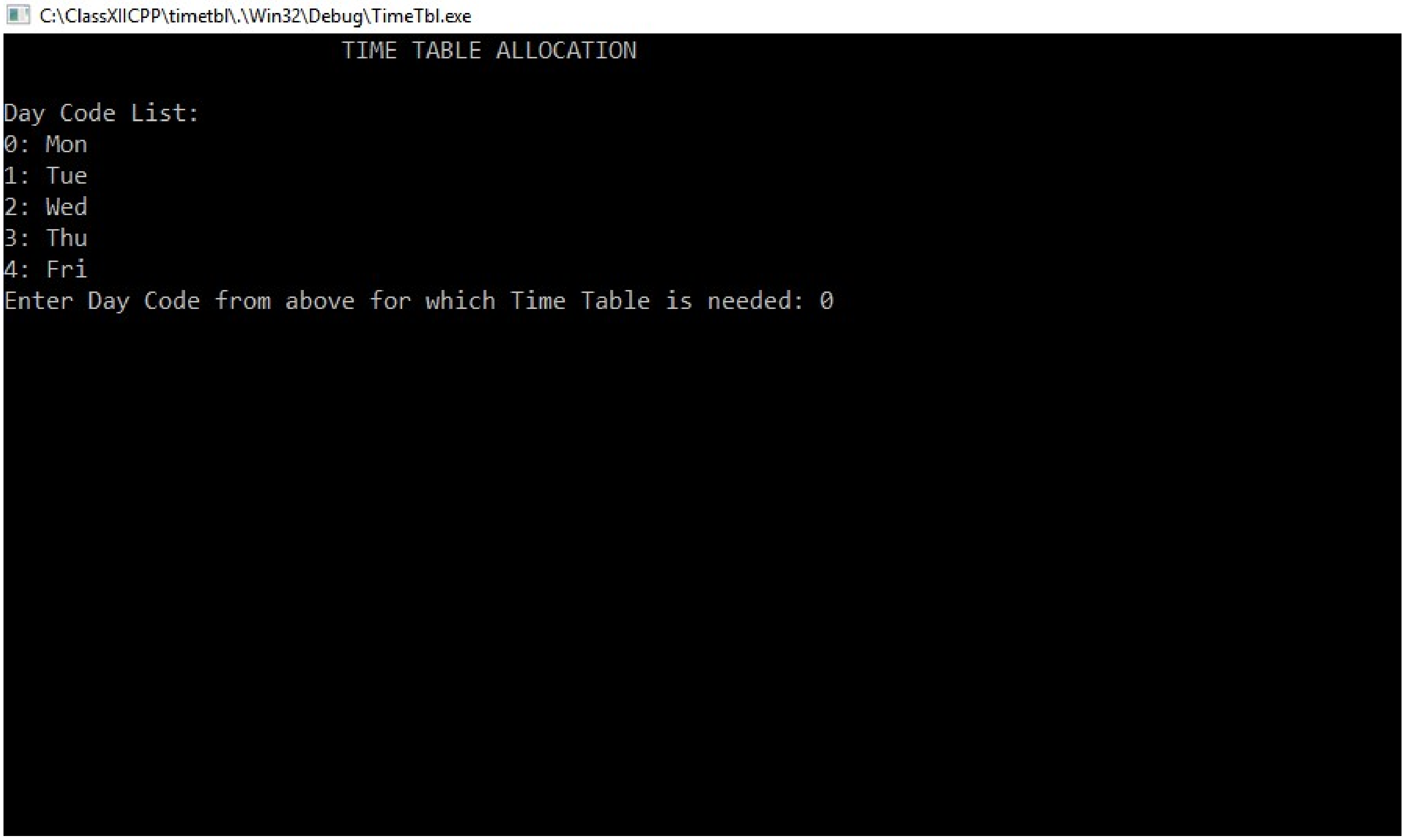
```
void displayTeacherTable(intdow)
{
    int i;
    drawLine(205, 79);
    cout<< endl << "\t";

    for(i=0; i<MAX_PERIODS; i++)
        cout<< "\t"<< i+1;
    cout<< endl;
    drawLine(205, 79);

    for(i=0; i<MAX_TEACHERS; i++)
      {
        cout<<endl<< teacher[i].teacherCd<< ": "
<<teacher[i].teacherName;
        for(int pd=0; pd<MAX_PERIODS;pd++)
        if(teacher[i].isAbsent[dow])
            cout<< "\t" << "ABSENT";
        else
if(teacher[i].timeTable[dow][pd].classCd<MAX_CLASSES)
                cout<< "\t" <<
    classRoom[teacher[i].timeTable[dow][pd].classCd].className<
<
                teacher[i].timeTable[dow][pd].subst;
        else
            cout<< "\t" << "" ; //"Free";
    }
    cout<< endl;
    drawLine(205, 79);
}

#include <c:\ttbl\datastr.cpp>
#include <c:\ttbl\classrm.cpp>
#include <c:\ttbl\teachers.cpp>
#include <c:\ttbl\assign.cpp>
#include <c:\ttbl\display.cpp>

int main()
{
    clrscr();
    cout << ”\t\t\tTIME TABLE ALLOCATION “ << endl << endl;
```

```
          freeSlot.initTimeTable(99,-1,"Free",' ');
          absentSlot.initTimeTable(99,-1,"Absent",' ');

            initClassRooms();

            readTeachers();

           readClassRooms();

           assignTeacher();

          int dow;
          cout<< "Day Code List:" << endl;
          for(inti=0;i<MAX_DAYS;i++)
                    cout<< i << ": " << today[i] << endl;

          cout<< "Enter Day Code from above for which Time Table is
needed: ";
          cin>>dow;

clrscr();
```

```
    cout<< "\t\t\tClass Time Table for " << today[dow] <<endl;
    displayClassTable(dow);

    cout<< "\n\n\n\n\t\t\tTeacher Time Table for " << today[dow]
<<endl;
    displayTeacherTable(dow);

    cout<< "\n\nEnter Absent Teacher's Codes, -1 to end: ";
    inttchCd;
    cin>>tchCd;

while (tchCd>=0)
     {
    teacher[tchCd].markAbsent(dow);
    cin>>tchCd;
    };

    cout<< "\n\t\t\tRevised Teacher Time Table for " <<
today[dow] <<endl;

    reassignTeachers(dow);
    displayTeacherTable(dow);
```

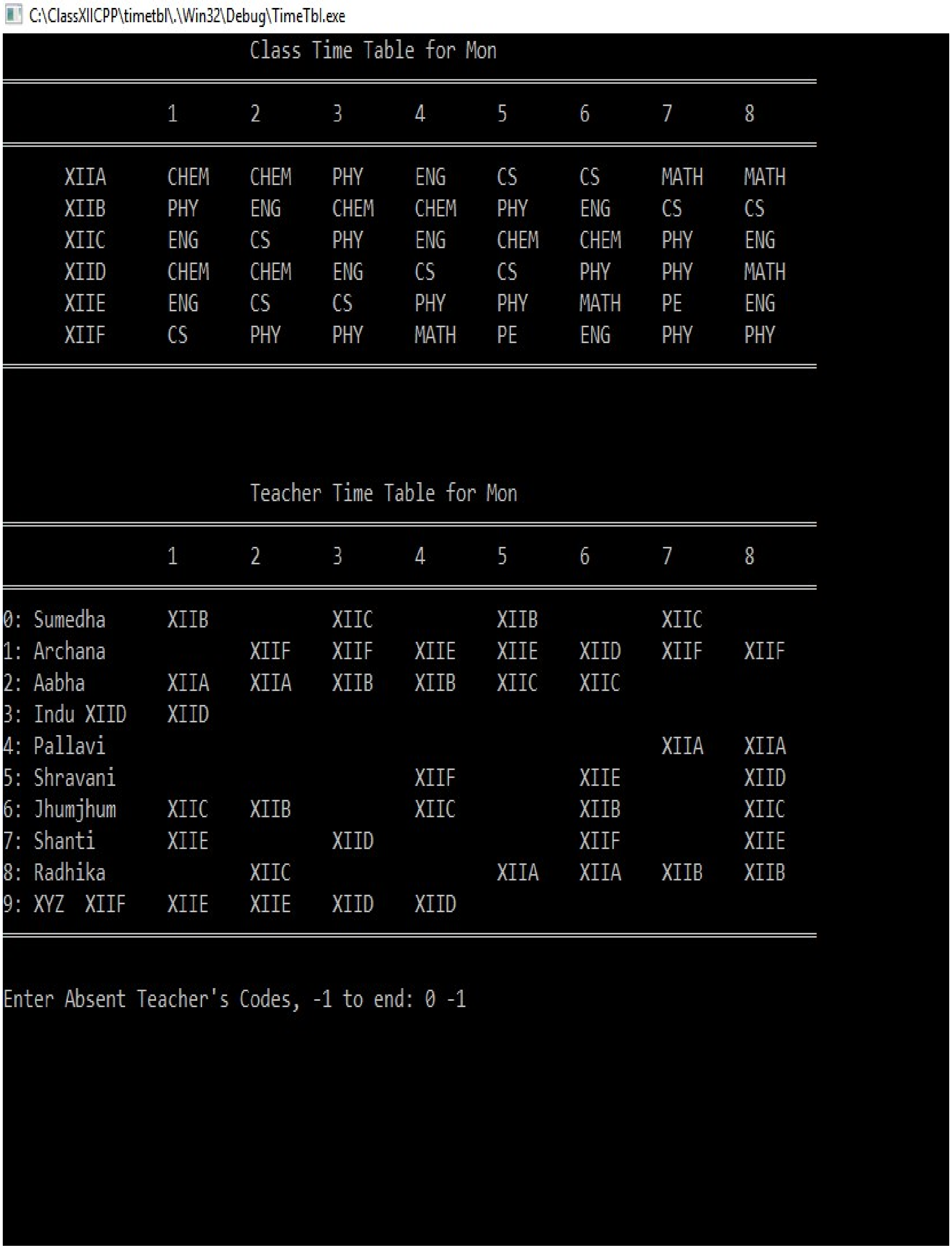
```
cout<< endl << "Press any key to end . . .";
        getch();
        return 0;
}
```

OUTPUTS

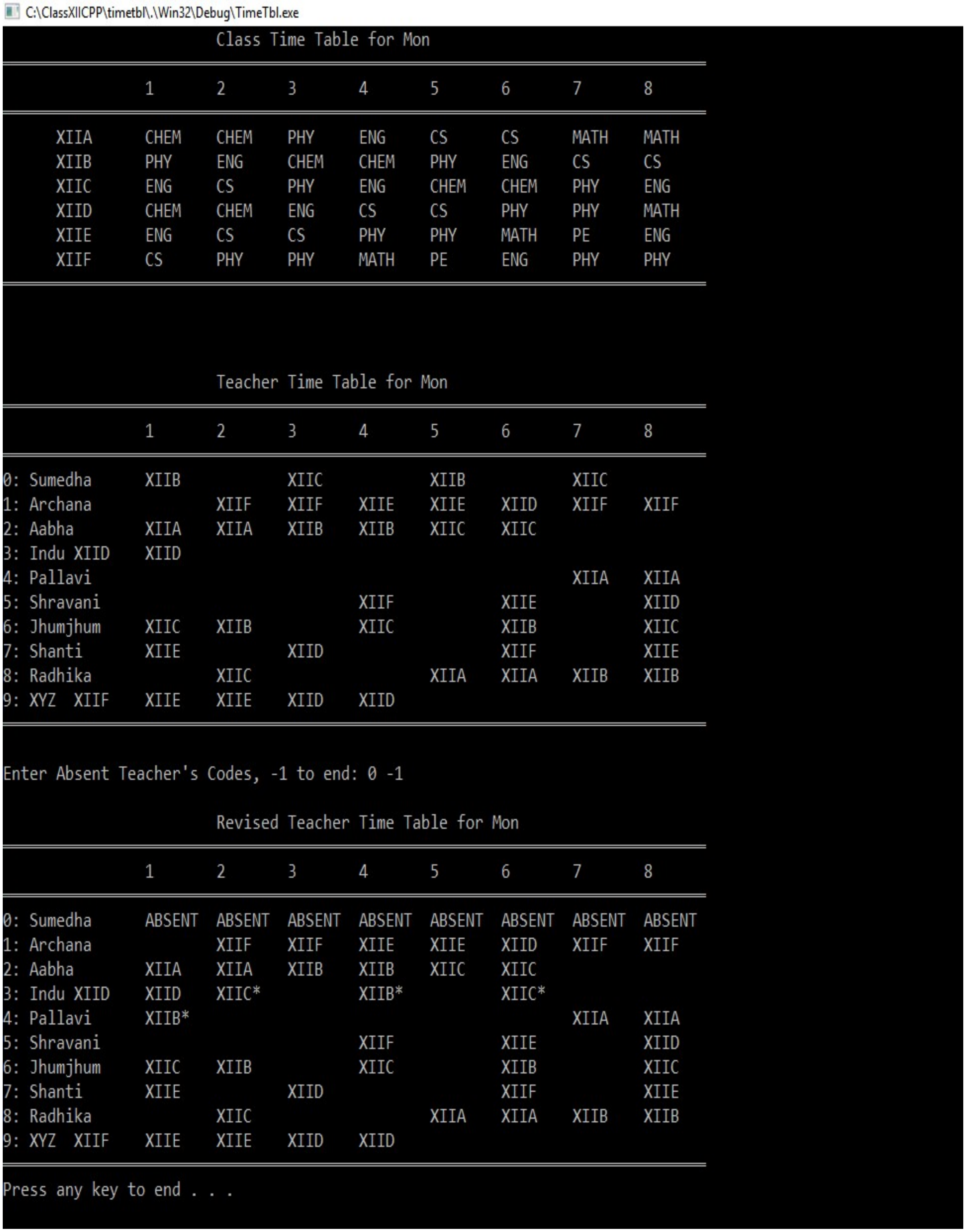


C:\ClassXIICPP\timetbl\.\Win32\Debug\TimeTbl.exe

```
                        TIME TABLE ALLOCATION

Day Code List:
0: Mon
1: Tue
2: Wed
3: Thu
4: Fri
Enter Day Code from above for which Time Table is needed: 0
```


C:\ClassXIICPP\timetbl\.\Win32\Debug\TimeTbl.exe

```
                        Class Time Table for Mon
===============================================================================
                1       2       3       4       5       6       7       8
===============================================================================
      XIIA      CHEM    CHEM    PHY     ENG     CS      CS      MATH    MATH
      XIIB      PHY     ENG     CHEM    CHEM    PHY     ENG     CS      CS
      XIIC      ENG     CS      PHY     ENG     CHEM    CHEM    PHY     ENG
      XIID      CHEM    CHEM    ENG     CS      CS      PHY     PHY     MATH
      XIIE      ENG     CS      CS      PHY     PHY     MATH    PE      ENG
      XIIF      CS      PHY     PHY     MATH    PE      ENG     PHY     PHY
===============================================================================



                        Teacher Time Table for Mon
===============================================================================
                1       2       3       4       5       6       7       8
===============================================================================
0: Sumedha      XIIB            XIIC            XIIB            XIIC
1: Archana              XIIF    XIIF    XIIE    XIIE    XIID    XIIF    XIIF
2: Aabha        XIIA    XIIA    XIIB    XIIB    XIIC    XIIC
3: Indu XIID    XIID
4: Pallavi                                                      XIIA    XIIA
5: Shravani                             XIIF            XIIE            XIID
6: Jhumjhum     XIIC    XIIB            XIIC            XIIB            XIIC
7: Shanti       XIIE            XIID                    XIIF            XIIE
8: Radhika              XIIC                    XIIA    XIIA    XIIB    XIIB
9: XYZ  XIIF    XIIE    XIIE    XIID    XIID
===============================================================================


Enter Absent Teacher's Codes, -1 to end: 0 -1
```



C:\ClassXIICPP\timetbl\.\Win32\Debug\TimeTbl.exe

```
                        Class Time Table for Mon
===============================================================================
                1       2       3       4       5       6       7       8
===============================================================================
      XIIA      CHEM    CHEM    PHY     ENG     CS      CS      MATH    MATH
      XIIB      PHY     ENG     CHEM    CHEM    PHY     ENG     CS      CS
      XIIC      ENG     CS      PHY     ENG     CHEM    CHEM    PHY     ENG
      XIID      CHEM    CHEM    ENG     CS      CS      PHY     PHY     MATH
      XIIE      ENG     CS      CS      PHY     PHY     MATH    PE      ENG
      XIIF      CS      PHY     PHY     MATH    PE      ENG     PHY     PHY
===============================================================================



                        Teacher Time Table for Mon
===============================================================================
                1       2       3       4       5       6       7       8
===============================================================================
0: Sumedha      XIIB            XIIC            XIIB            XIIC
1: Archana              XIIF    XIIF    XIIE    XIIE    XIID    XIIF    XIIF
2: Aabha        XIIA    XIIA    XIIB    XIIB    XIIC    XIIC
3: Indu XIID    XIID
4: Pallavi                                                      XIIA    XIIA
5: Shravani                             XIIF            XIIE            XIID
6: Jhumjhum     XIIC    XIIB            XIIC            XIIB            XIIC
7: Shanti       XIIE            XIID                    XIIF            XIIE
8: Radhika              XIIC                    XIIA    XIIA    XIIB    XIIB
9: XYZ  XIIF    XIIE    XIIE    XIID    XIID
===============================================================================

Enter Absent Teacher's Codes, -1 to end: 0 -1

                        Revised Teacher Time Table for Mon
===============================================================================
                1       2       3       4       5       6       7       8
===============================================================================
0: Sumedha      ABSENT  ABSENT  ABSENT  ABSENT  ABSENT  ABSENT  ABSENT  ABSENT
1: Archana              XIIF    XIIF    XIIE    XIIE    XIID    XIIF    XIIF
2: Aabha        XIIA    XIIA    XIIB    XIIB    XIIC    XIIC
3: Indu XIID    XIID    XIIC*           XIIB*           XIIC*
4: Pallavi      XIIB*                                           XIIA    XIIA
5: Shravani                             XIIF            XIIE            XIID
6: Jhumjhum     XIIC    XIIB            XIIC            XIIB            XIIC
7: Shanti       XIIE            XIID                    XIIF            XIIE
8: Radhika              XIIC                    XIIA    XIIA    XIIB    XIIB
9: XYZ  XIIF    XIIE    XIIE    XIID    XIID
===============================================================================

Press any key to end . . .
```