PROGRAMMING III PROJECT: SOCCER CLUB TECHNICAL TEAM

Surname & initials: Mabasa NM

Student Number: 209088398

```
//---------------------------------------------------------------------------

#ifndef TTeam_uH
#define TTeam_uH
#include <vcl.h>
//---------------------------------------------------------------------------
class TTeam
{
        private:
                AnsiString cName;
                AnsiString tFormation;
        public:
                TTeam();
                TTeam(AnsiString cN);
                void setFormation(AnsiString tF);
                void selcLineUp();
                int defPerformance();
                int attPerformance();
                void DisplayRadio(TRadioGroup *rdg);
                ~TTeam();
};
#endif
//---------------------------------------------------------------------------

#pragma hdrstop

#include "TTeam_u.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)

TTeam ::TTeam()
{
        cName = "UnKnown";
        tFormation = "x-x-x";
}

TTeam ::TTeam(AnsiString cN)
{
        cName = cN;
}

void TTeam ::setFormation(AnsiString tF)
{
   tFormation = tF;
}

void TTeam ::DisplayRadio(TRadioGroup *rdg)
{
        rdg->Items->Add(cName);
}
```

```
int TTeam ::defPerformance()
{
        int def = random(100)+ 30;

        return def;
}

int TTeam ::attPerformance()
{
        int att = random(100)+ 30;

        return att;
}

TTeam::~TTeam()
{

}
//---------------------------------------------------------------------------
#ifndef TPlayer_uH
#define TPlayer_uH
#include <vcl.h>
//---------------------------------------------------------------------------
class TPlayer
{
        private:
                AnsiString name;
                AnsiString birthDate;
                AnsiString position;
                int jersyNo;

        public:
          TPlayer();
          TPlayer(AnsiString nm,AnsiString bD,AnsiString pos,int jN);
          AnsiString getName(){return name;};
          AnsiString getBirthDate(){return birthDate;};
          AnsiString getPosition(){return position;};
          int getJersyNo(){return jersyNo;};
          ~TPlayer();
};
#endif
//---------------------------------------------------------------------------

#pragma hdrstop

#include "TPlayer_u.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
TPlayer::TPlayer()
{
         name = "Unknown" ;
```

```
            birthDate = "1800/01/01";
            position = "xx";
            jersyNo = 0;
}
TPlayer::TPlayer(AnsiString nm,AnsiString bD,AnsiString pos,int jN)
{
            name = nm;
            birthDate = bD;
            position = pos;
            jersyNo = jN;
}
TPlayer::~TPlayer()
{

}
//---------------------------------------------------------------------------

#ifndef PGG_Project_uH
#define PGG_Project_uH
//---------------------------------------------------------------------------
#include <System.Classes.hpp>
#include <Vcl.Controls.hpp>
#include <Vcl.StdCtrls.hpp>
#include <Vcl.Forms.hpp>
#include "TPlayer_u.h"
#include "TTeam_u.h"
#include <Vcl.ComCtrls.hpp>
#include <Vcl.ExtCtrls.hpp>
#include "SquadForm_u.h"
#include <Vcl.Grids.hpp>
//---------------------------------------------------------------------------
class TfrmTournament : public TForm
{
__published:     // IDE-managed Components
            TPageControl *PageControl1;
            TTabSheet *tbsCountry;
            TImage *imgTournament;
            TButton *btnLoad;
            TRadioGroup *rdgCountry;
            TTabSheet *tbsResult;
            TLabel *lblCompTeam;
            TLabel *lblFormation;
            TPanel *Panel1;
            TImage *imgGround;
            TButton *btnPerFormacce;
            TTabSheet *tbsPerFormance;
            TStringGrid *stgOverallPerformance;
            TImage *imgBackGround;
            TImage *imgPerFormance;
            TLabel *lblStadium;
            TButton *btnDisplayFormation;
```

```
        void __fastcall btnLoadClick(TObject *Sender);
        void __fastcall rdgCountryClick(TObject *Sender);
        void __fastcall FormShow(TObject *Sender);
        void __fastcall tbsResultShow(TObject *Sender);
        void __fastcall btnPerFormacceClick(TObject *Sender);
        void __fastcall btnDisplayFormationClick(TObject *Sender);

private:// User declarations
                int countGK,countDF,countMF,countFW,countPlayers;

public:         // User declarations
        __fastcall TfrmTournament(TComponent* Owner);
        void computerLineup();
        TTeam *teams[18];
        TPlayer *teamSquad[25];
        void extractLineUp();
        AnsiString comLineUp[11];
        void userLineUp();
};
//---------------------------------------------------------------------------
extern PACKAGE TfrmTournament *frmTournament;
//---------------------------------------------------------------------------
#endif

//---------------------------------------------------------------------------

#include <vcl.h>
#pragma hdrstop

#include "PGG_Project_u.h"
#include <jpeg.hpp>
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
TfrmTournament *frmTournament;
AnsiString cName;
TPanel *pnlPlayer;
TLabel *lblPlayer;
AnsiString compTeam,players[11];
//---------------------------------------------------------------------------
__fastcall TfrmTournament::TfrmTournament(TComponent* Owner)
        : TForm(Owner)
{
        countGK = 0,countDF = 0,countMF = 0,countFW = 0,countPlayers = 0;
        randomize();
        Randomize();
        PageControl1->ActivePageIndex =  0;
}
//---------------------------------------------------------------------------
void __fastcall TfrmTournament::btnLoadClick(TObject *Sender)
{
        rdgCountry->Items->Clear();
        TStringList *myList = new TStringList();

        myList->LoadFromFile("afcon2013_teams.txt");
        AnsiString line, country;

        for(int c = 0; c < 2; c++)
        {
                myList->Delete(0);
        }
        for (int x = 0 ; x < myList->Count;x++)
        {
                line = myList->Strings[x];

                int pos = line.AnsiPos("\t");

                for(int i = 0; i < 2; i++)
                {
                        line =  line.Delete(1,pos);
                        line = line.Trim();
                        pos = line.AnsiPos("\t");
                }
                teams[x] = new TTeam(line.SubString(1,pos-1));
                teams[x]->DisplayRadio(rdgCountry);
        }
        btnLoad->Enabled = false;
}
//---------------------------------------------------------------------------
void __fastcall TfrmTournament::rdgCountryClick(TObject *Sender)
{
        randomize();
        frmSquad->chkSquad->Enabled = true;
        frmSquad->lsbxStart->Clear();
        frmSquad->lsbxSubs->Clear();
        computerLineup();
        if(compTeam != rdgCountry->Items->Strings[rdgCountry->ItemIndex])
        {
                tbsCountry->Enabled = false;
  }

}
//---------------------------------------------------------------------------
void __fastcall TfrmTournament::FormShow(TObject *Sender)
{
        imgTournament->Picture->LoadFromFile("PICS//opening1.bmp");
}
//---------------------------------------------------------------------------
void TfrmTournament::computerLineup()
{
        int RandTeam;
        RandTeam = random(rdgCountry->Items->Count);
```

```
compTeam = rdgCountry->Items->Strings[RandTeam];
frmSquad->chkSquad->Clear();
cName = compTeam;

AnsiString rdgCountryText,LineNumber,Position,Names,line,line2,ItemX[4];
int posTab;
bool found = false;

TStringList *myList = new TStringList();
myList->LoadFromFile("afcon2013_players.txt");

if(RandTeam != rdgCountry->ItemIndex)
{

        int index = 0;
        for (int i = 0; i < myList->Count;i++)
        {
                line = myList->Strings[i];
                index++;
                if(line.Trim().UpperCase() == rdgCountry->Items->Strings[rdgCountry
                        ->ItemIndex].UpperCase())
                {
                        frmSquad->lblTeamSquad->Caption = line.Trim().UpperCase();
                        found = true;
                        break;
                }
        }
        if (!found)
        {
         return;

        }
        int t = 0;
        for(int x = index+1 ; x < myList->Count; x++)
        {
                line2 = myList->Strings[x];

                if(line2.IsEmpty())
                {
                        break;
                }
                posTab = line2.AnsiPos("\t");
                ItemX[0] = line2.SubString(1,posTab-1);
                line2 = line2.Delete(1,posTab).Trim();
                posTab = line2.AnsiPos("\t");
                ItemX[1] = line2.SubString(1,posTab-1);
                line2 = line2.Delete(1,posTab).Trim();
                posTab = line2.AnsiPos("\t");
                ItemX[2] = line2.SubString(1,posTab-1);
                line2 = line2.Delete(1,posTab).Trim();
                ItemX[3] = line2.Trim();
                teamSquad[t] = new TPlayer(ItemX[2],ItemX[3],ItemX[1],
                ItemX[0].ToInt());

                frmSquad->chkSquad->Items->Add(IntToStr(teamSquad[t]->getJersyNo())+
                " "+teamSquad[t]->getPosition()+" "+teamSquad[t]->getName());
                t++;
        }

  frmSquad->Show();
}
else
{
        ShowMessage("You have selected the same team as the computer");
}
int indx = 0;
AnsiString comLine,comLine2,comItem[3],comSquad[25];
found = false;

for(int x = 0; x < myList->Count; x++)
{
        comLine = myList->Strings[x];
        indx++;
        if(comLine.Trim().UpperCase() == compTeam.Trim().UpperCase())
        {
                found = true;
                break;
        }
}

if(!found)
{
  return;
}
int n = 0;
for(int m = indx+1; m < myList->Count; m++)
{
        comLine2 = myList->Strings[m];

        if(comLine2.IsEmpty())
        {
                break;
        }  //ShowMessage(comLine2);
        posTab = comLine2.AnsiPos("\t");
        comItem[0] = comLine2.SubString(1,posTab-1);
        comLine2 = comLine2.Delete(1,posTab).Trim();
        posTab = comLine2.AnsiPos("\t");
        comItem[1] = comLine2.SubString(1,posTab-1);
        comLine2 = comLine2.Delete(1,posTab).Trim();
        posTab = comLine2.AnsiPos("\t");
        comItem[2] = comLine2.SubString(1,posTab-1);
```

```
    comLine2 = comLine2.Delete(1,posTab).Trim();

    comSquad[n] = comItem[0]+" "+comItem[1]+" "+comItem[2];
    n++;

}
AnsiString role,itemP;

AnsiString GK,DF[4],MF[4],FW[2];

for(int v = 0; v < 11;v++)
{
    if (countPlayers == 11)
    {
        break;
        countPlayers = 0;
    }

    for (int s = 0; s < 25; s++)
    {
        int randPlayers = Random(23);

        itemP = comSquad[randPlayers];
        role = itemP.SubString(itemP.AnsiPos(" "),3).Trim();

        if (role == "GK")
        {
            GK = comSquad[randPlayers];
        }
        if (role == "DF")
        {
            switch (countDF)
            {
                case 0: DF[0] = comSquad[randPlayers]; break;
                case 1: DF[1] = comSquad[randPlayers]; break;
                case 2: DF[2] = comSquad[randPlayers]; break;
                case 3: DF[3] = comSquad[randPlayers]; break;
            }
            countDF++;
        }
        if (role == "MF")
        {
            switch (countMF)
            {
                case 0: MF[0] = comSquad[randPlayers]; break;
                case 1: MF[1] = comSquad[randPlayers]; break;
                case 2: MF[2] = comSquad[randPlayers]; break;
                case 3: MF[3] = comSquad[randPlayers]; break;
            }
            countMF++;
        }
```

```
            if (role == "FW")
            {
                switch (countFW)
                {
                    case 0: FW[0] = comSquad[randPlayers]; break;
                    case 1: FW[1] = comSquad[randPlayers]; break;
                }
                countFW++;
            }
        }

        countPlayers++;
     }
        countGK = 0,countDF = 0,countMF = 0,countFW = 0,countPlayers = 0;

        comLineUp[0] = GK;
        comLineUp[1] = DF[0];
        comLineUp[2] = DF[1];
        comLineUp[3] = DF[2];
        comLineUp[4] = DF[3];
        comLineUp[5] = MF[0];
        comLineUp[6] = MF[1];
        comLineUp[7] = MF[2];
        comLineUp[8] = MF[3];
        comLineUp[9] = FW[0];
        comLineUp[10] = FW[1];

}
//---------------------------------------------------------------------------
void __fastcall TfrmTournament::tbsResultShow(TObject *Sender)
{
    AnsiString pRole,jNo;int pos;
    AnsiString tempLine;

    for(int x = 0; x < 11; x++)
    {
        pnlPlayer = new TPanel(this);
        lblPlayer = new TLabel(this);
        lblPlayer->Parent = Panel1;
        pnlPlayer->Parent = Panel1;
        pnlPlayer->ParentBackground = false;
        pnlPlayer->Width = 20;
        pnlPlayer->Height = 20;
        pnlPlayer->Color = clBlue;
        pnlPlayer->Left = 30 + (30*x);

        tempLine = comLineUp[x];

        pos = tempLine.AnsiPos(" ");
        jNo = tempLine.SubString(1,pos-1);
        tempLine = tempLine.Delete(1,pos).Trim();
        pos = tempLine.AnsiPos(" ");
        pRole = tempLine.SubString(1,pos-1).Trim();



        if(pRole == "GK")
        {
            pnlPlayer->Left = 30 + (30*x);
            pnlPlayer->Top = Panel1->Height-(Panel1->Height/1.9);
            pnlPlayer->Caption = jNo;
            lblPlayer->Top = pnlPlayer->Top-20;
            lblPlayer->Caption = tempLine.Trim();
        }
        if(pRole == "DF")
        {
            pnlPlayer->Left = 30 + (30*4);
            pnlPlayer->Top = 10 + (70*(x));
            lblPlayer->Top = pnlPlayer->Top-20;
            lblPlayer->Left = 30 + (30*4);
            pnlPlayer->Caption = jNo;
            lblFormation->Tag = 3;
            lblPlayer->Caption = tempLine.Trim();
        }
        if(pRole == "MF")
        {
            pnlPlayer->Left = 30 + (30*8);
            pnlPlayer->Top = 10 + (70*(x-4));
            lblPlayer->Top = pnlPlayer->Top-20;
            pnlPlayer->Caption = jNo;
            lblPlayer->Left = 30 + (30*8);
            lblPlayer->Caption = tempLine.Trim();
        }
        if(pRole == "FW")
        {
            pnlPlayer->Left = 30 + (30*12);
            pnlPlayer->Top = 60 + (70*(x-8));
            lblPlayer->Top = pnlPlayer->Top-20;
            pnlPlayer->Caption = jNo;
            lblPlayer->Left = 30 + (30*12);
            lblPlayer->Caption = tempLine.Trim();
        }
    }
    imgGround->Picture->LoadFromFile("Ground.bmp");
    lblCompTeam->Caption ="COMPUTER TEAM : " +cName;
    lblFormation->Caption = "FORMATION : 4-4-2";
}
//---------------------------------------------------------------------------
void TfrmTournament::userLineUp()
{
   AnsiString Splayers,formation,pRole,jNo,name;
```

```
formation = frmSquad->cmbFormation->Text;
//dynamically create panels representing the formation of 4-4-2 else of 4-3-3
if (formation == "4-4-2")
{
        for (int x = 0; x < 11; x++)
        {
                Splayers =  frmSquad->lsbxStart->Items->Strings[x];

                jNo = Splayers.SubString(1,Splayers.AnsiPos(" ")-1);
                pRole = Splayers.SubString(Splayers.AnsiPos(" "),3).Trim();
                name = Splayers.SubString(Splayers.AnsiPos(" "),Splayers.Length());

                TPanel *pnlPlayer = new TPanel(this);
                TLabel *lblPlayer = new TLabel(this);
                lblPlayer->Parent = Panel1;
                pnlPlayer->Parent = Panel1;
                pnlPlayer->ParentBackground = false;
                pnlPlayer->Width = 40;
                pnlPlayer->Height = 20;
                pnlPlayer->Color = clRed;
                pnlPlayer->Left = 30 + (30*x);

                if(pRole == "GK")
                {
                        pnlPlayer->Left = 30 + (30*x);
                        pnlPlayer->Top = Panel1->Height-(Panel1->Height/1.8);
                        pnlPlayer->Caption = jNo;
                        lblPlayer->Top = pnlPlayer->Top-20;
                        lblPlayer->Caption = name;
                }
                if(pRole == "DF")
                {
                        pnlPlayer->Left = 30 + (30*4);
                        pnlPlayer->Top = 10 + (70*(x));
                        lblPlayer->Top = pnlPlayer->Top-20;
                        lblPlayer->Left = 30 + (30*4);
                        pnlPlayer->Caption = jNo;
                        lblPlayer->Caption = name;
                }
                if(pRole == "MF")
                {
                        pnlPlayer->Left = 30 + (30*8);
                        pnlPlayer->Top = 10 + (70*(x-4));
                        lblPlayer->Top = pnlPlayer->Top-20;
                        pnlPlayer->Caption = jNo;
                        lblPlayer->Left = 30 + (30*8);
                        lblPlayer->Caption = name;
                }
                if(pRole == "FW")
                {
                        pnlPlayer->Left = 30 + (30*12);
```

```
                    pnlPlayer->Top = 60 + (70*(x-8));
                    lblPlayer->Top = pnlPlayer->Top-20;
                    pnlPlayer->Caption = jNo;
                    lblPlayer->Left = 30 + (30*12);
                    lblPlayer->Caption = name;
                }

            }

        }
        else
        {
            for (int x = 0; x < 11; x++)
            {
                Splayers =  frmSquad->lsbxStart->Items->Strings[x];

                jNo = Splayers.SubString(1,Splayers.AnsiPos(" ")-1);
                pRole = Splayers.SubString(Splayers.AnsiPos(" "),3).Trim();
                name = Splayers.SubString(Splayers.AnsiPos(" "),Splayers.Length());

                TPanel *pnlPlayer = new TPanel(this);
                TLabel *lblPlayer = new TLabel(this);
                lblPlayer->Parent = Panel1;
                pnlPlayer->Parent = Panel1;
                pnlPlayer->ParentBackground = false;
                pnlPlayer->Width = 40;
                pnlPlayer->Height = 20;
                pnlPlayer->Color = clRed;
                pnlPlayer->Left = 30 + (30*x);

                if(pRole == "GK")
                {
                    pnlPlayer->Left = 30 + (30*x);
                    pnlPlayer->Top = Panel1->Height-(Panel1->Height/1.8);
                    pnlPlayer->Caption = jNo;
                    lblPlayer->Top = pnlPlayer->Top-20;
                    lblPlayer->Caption = name;
                }
                if(pRole == "DF")
                {
                    pnlPlayer->Left = 30 + (30*4);
                    pnlPlayer->Top = 10 + (70*(x));
                    lblPlayer->Top = pnlPlayer->Top-20;
                    lblPlayer->Left = 30 + (30*4);
                    pnlPlayer->Caption = jNo;
                    lblPlayer->Caption = name;
                }
                if(pRole == "MF")
                {
                    pnlPlayer->Left = 30 + (30*8);
                    pnlPlayer->Top = 40 + (70*(x-4));
                    lblPlayer->Top = pnlPlayer->Top-20;
                    pnlPlayer->Caption = jNo;
                    lblPlayer->Left = 30 + (30*8);
                    lblPlayer->Caption = name;
                }
                if(pRole == "FW")
                {
                    pnlPlayer->Left = 30 + (30*12);
                    pnlPlayer->Top = 40+ (70*(x-7));
                    lblPlayer->Top = pnlPlayer->Top-20;
                    pnlPlayer->Caption = jNo;
                    lblPlayer->Left = 30 + (30*12);
                    lblPlayer->Caption = name;
                }

            }

        }


    }
    //---------------------------------------------------------------------------
    void __fastcall TfrmTournament::btnPerFormacceClick(TObject *Sender)
    {


        Sleep(1000);
        tbsPerFormance->Show();
        tbsCountry->Enabled = true;
        btnLoad->Enabled = true;

        int compAttPerf = random(100)+ 30;
        int compDefPerf = random(100)+ 30;
        int scoreUser,scoreComp;
        int userDefPerf = teams[rdgCountry->ItemIndex]->defPerformance();
        int userAttPerf = teams[rdgCountry->ItemIndex]->attPerformance();

        if(compAttPerf < userAttPerf)
        {
            scoreUser = random(4)+2;
            scoreComp = random(2);
        }
        else
        {
            scoreComp = random(4)+2;
            scoreUser = random(2);
        }
        if(compDefPerf < userDefPerf)
        {
            scoreComp = scoreComp * 0.5;
            scoreUser = scoreUser * 1;
```

```
          }
          else
          {
                    scoreComp = scoreComp * 1;
                    scoreUser = scoreUser * 0.5;
   }

          imgPerFormance->Picture->LoadFromFile("PICS//opening2.bmp");
          imgBackGround->Picture->LoadFromFile("PICS//OPENING.JPG");

           AnsiString Headers[] = {"Stats",compTeam,rdgCountry->Items->
           Strings[rdgCountry->ItemIndex]};
           AnsiString fColumn[] = {"Deffence","Attack","Score"};

           for(int j = 1; j < 4;j++)
           {
                    stgOverallPerformance->Cells[0][j] = fColumn[j-1];
           }
           for(int c = 0; c < 3; c++)
           {
                    stgOverallPerformance->Cells[c][0] = Headers[c];
           }
           int userStats[] = {compDefPerf,userAttPerf,scoreUser};
           for(int d = 1; d < 4; d++)
           {
                    stgOverallPerformance->Cells[2][d] =  userStats[d-1];
           }
           int compStats[] = {userDefPerf,compAttPerf,scoreComp};
           for(int e = 1; e < 4; e++)
           {
                    stgOverallPerformance->Cells[1][e] =  compStats[e-1];
           }
           AnsiString venue[5] = {"Mbombela","RustenBurg","Moses Mabida","Soccer City"
           ,"Polokwane"};

           lblStadium->Caption = "Match was played at "+venue[random(5)]+ " On "+
           FormatDateTime("dddd,'the' dd 'of' mmmm,yyyy '@' hh:nn",Now());
}
//---------------------------------------------------------------------------
void __fastcall TfrmTournament::btnDisplayFormationClick(TObject *Sender)
{
          lblCompTeam->Caption ="USER TEAM : " +rdgCountry->Items->
           Strings[rdgCountry->ItemIndex];
          lblFormation->Caption = "FORMATION : "+ frmSquad->cmbFormation->Text;
          userLineUp();
          btnDisplayFormation->Enabled = false;

          for (int i = 0; i < 22;i++)
            {
                              Sleep(50);
                              Update();
```

```
                        delete Components[17];

         }


}
//---------------------------------------------------------------------------

//---------------------------------------------------------------------------

#ifndef SquadForm_uH
#define SquadForm_uH
//---------------------------------------------------------------------------
#include <System.Classes.hpp>
#include <Vcl.Controls.hpp>
#include <Vcl.StdCtrls.hpp>
#include <Vcl.Forms.hpp>
#include <Vcl.CheckLst.hpp>
#include "PGG_Project_u.h"
#include "TPlayer_u.h"
#include "TTeam_u.h"
#include <Vcl.ExtCtrls.hpp>
//---------------------------------------------------------------------------
class TfrmSquad : public TForm
{
__published:	// IDE-managed Components
        TCheckListBox *chkSquad;
        TListBox *lsbxStart;
        TListBox *lsbxSubs;
        TLabel *lblSubst;
        TLabel *lblLineup;
        TLabel *lblTeamSquad;
        TTimer *Timer1;
        TComboBox *cmbFormation;
        TLabel *lblFormation;
        void __fastcall chkSquadClickCheck(TObject *Sender);
        void __fastcall Timer1Timer(TObject *Sender);
        void __fastcall cmbFormationChange(TObject *Sender);
private:// User declarations
        int maxGK,maxDF,maxMF,maxFW;
public:		// User declarations
        __fastcall TfrmSquad(TComponent* Owner);
                void __fastcall myButton1Click(TObject *Sender);
                void __fastcall mybuttonClick(TObject *Sender);
                void MaxPlayerRole();

};
//---------------------------------------------------------------------------
extern PACKAGE TfrmSquad *frmSquad;
//---------------------------------------------------------------------------
#endif
```

```
//---------------------------------------------------------------------------

#include <vcl.h>
#pragma hdrstop

#include "SquadForm_u.h"
//---------------------------------------------------------------------------
#pragma package(smart_init)
#pragma resource "*.dfm"
TfrmSquad *frmSquad;
//---------------------------------------------------------------------------
__fastcall TfrmSquad::TfrmSquad(TComponent* Owner)
	: TForm(Owner)
{
	Timer1->Enabled = false;
	cmbFormation->Items->Add("4-4-2");
	cmbFormation->Items->Add("4-3-3");
	maxGK = 1,maxDF = 1,maxMF = 1,maxFW = 1;
	chkSquad->Visible = false;
	lsbxStart->Visible = false;
	lsbxSubs->Visible = false;
	lblTeamSquad->Visible = false;
	lblLineup->Visible = false;
	lblSubst->Visible = false;
}
//---------------------------------------------------------------------------
void __fastcall TfrmSquad::myButton1Click(TObject *Sender)
{
 for (int i = 0; i < chkSquad->Count; i++)
 {
	if(chkSquad->Checked[i])
	{
	 lsbxStart->Items->Add(chkSquad->Items->Strings[i]);
	}
	else
	{
	 lsbxSubs->Items->Add(chkSquad->Items->Strings[i]);
	}
 }
 for(int c = 0; c < frmSquad->ComponentCount;)
 {
	TButton *btn = dynamic_cast<TButton*>(Components[c]);

	if(btn != NULL)
	{
	 delete btn;
	}
	else
		c++;
 }
 Timer1->Enabled = true;
```

```
}
//---------------------------------------------------------------------------


void __fastcall TfrmSquad::chkSquadClickCheck(TObject *Sender)
{
chkSquad->ItemEnabled[chkSquad->ItemIndex] = false;
        MaxPlayerRole();
        int count = 0;
        for (int i = 0; i < chkSquad->Count; i++)
        {
                if(chkSquad->Checked[i])
                {
                        count++;

                }
        }
        if (count == 11)
        {
                MessageDlg("You have already selected Eleven players\n",mtInformation,
                TMsgDlgButtons()<<mbOK,0);
                chkSquad->Enabled = false;
        }
        if (count == 11 )
        {
                TButton *mybutton[2];

                for (int i = 0; i < 2; i++)
                {
                        mybutton[i] = new TButton(this);
                        mybutton[i]->Parent = frmSquad;
                }

                mybutton[0]->Width = 150;
                mybutton[0]->Top = 50;
                mybutton[0]->Left = 519;
                mybutton[0]->OnClick = myButton1Click;
                mybutton[0]->Caption = "***POPULATE PLAYERS***";

                mybutton[1]->Width = 150;
                mybutton[1]->Top = mybutton[0]->Top*2;
                mybutton[1]->Left = 519;
                mybutton[1]->OnClick = mybuttonClick;
                mybutton[1]->Caption = "***MAKE CHANGES***";
        }
        if (count > 11)
        {
                for (int i = 0; i < chkSquad->Count; i++)
                {
                        chkSquad->Checked[i] = false;
                }
                lsbxStart->Clear();
                lsbxSubs->Clear();

                for (int i = 0; i < frmSquad->ComponentCount; )
                {
                        if (frmSquad->Components[i]->ClassNameIs("TButton"))
                        {
                                delete Components[i];
                        }
                        else
                        i++;
                }
 }

}
void __fastcall TfrmSquad::mybuttonClick(TObject *Sender)
{
  chkSquad->Enabled = true;
  maxGK = 1,maxDF = 1,maxMF = 1,maxFW = 1;
        for (int i = 0; i < chkSquad->Count; i++)
        {
                chkSquad->Checked[i] = false;
        }
  lsbxStart->Clear();
  lsbxSubs->Clear();

 for(int c = 0; c < frmSquad->ComponentCount;)
 {
        TButton *btn = dynamic_cast<TButton*>(Components[c]);

        if(btn != NULL)
        {
          delete btn;
        }
        else
                c++;
 }
}
//---------------------------------------------------------------------------
void TfrmSquad::MaxPlayerRole()
{
   int pos,indx;
        AnsiString formation,role,chekPlayers = chkSquad->Items->Strings[chkSquad->ItemIndex];
        indx = chkSquad->ItemIndex;
        pos = chekPlayers.AnsiPos(" ");
        chekPlayers = chekPlayers.Delete(1,pos).Trim();
        pos = chekPlayers.AnsiPos(" ");
        role = chekPlayers.SubString(1,pos-1).Trim();

        formation = cmbFormation->Text.Trim();
```

```
if (formation == "4-3-3")
{
        if (role == "GK")
        {
                if (maxGK > 1)
                {
                        ShowMessage("You can select only 1 Goalkeeper");
                        chkSquad->Checked[indx] = false;
                }
                maxGK++;
        }
        if (role == "DF")
        {

                if (maxDF > 4)
                {
                        ShowMessage("You can select only 4 Defenders");
                        chkSquad->Checked[indx] = false;
                }
                maxDF++;
        }
        if (role == "MF")
        {

                if (maxMF > 3)
                {
                        ShowMessage("You can select only 3 Midfielders");
                        chkSquad->Checked[indx] = false;
                }
                maxMF++;
        }
        if (role == "FW")
        {

                if (maxFW > 3)
                {
                        ShowMessage("You can select only 3 Foward");
                        chkSquad->Checked[indx] = false;
                }
                maxFW++;
        }
}
else
{
        if (role == "GK")
        {
                if (maxGK > 1)
                {
                        ShowMessage("You can select only 1 Goalkeeper");
                        chkSquad->Checked[indx] = false;
                }
```

```
                        maxGK++;
                }
                if (role == "DF")
                {

                        if (maxDF > 4)
                        {
                                ShowMessage("You can select only 4 Defenders");
                                chkSquad->Checked[indx] = false;
                        }
                        maxDF++;
                }
                if (role == "MF")
                {

                        if (maxMF > 4)
                        {
                                ShowMessage("You can select only 4 Midfielders");
                                chkSquad->Checked[indx] = false;
                        }
                        maxMF++;
                }
                if (role == "FW")
                {

                        if (maxFW > 2)
                        {
                                ShowMessage("You can select only 2 Foward");
                                chkSquad->Checked[indx] = false;
                        }
                        maxFW++;
                }
        }
}
//---------------------------------------------------------------------------

void __fastcall TfrmSquad::Timer1Timer(TObject *Sender)
{
        frmSquad->Close();
        maxGK = 1,maxDF = 1,maxMF = 1,maxFW = 1;
        frmTournament->tbsResult->Show();
        Timer1->Enabled = false;
}
//---------------------------------------------------------------------------

void __fastcall TfrmSquad::cmbFormationChange(TObject *Sender)
{
        chkSquad->Visible = true;
        lsbxStart->Visible = true;
        lsbxSubs->Visible = true;
        lblFormation->Caption = "Your Team Formation : "+cmbFormation->Text;
```

```
	cmbFormation->Visible = false;
	lblTeamSquad->Visible = true;
	lblLineup->Visible = true;
	lblSubst->Visible = true;
}
//---------------------------------------------------------------------------
```