

**ANNEXURE I: SOLUTION FOR QUESTION 3: DELPHI**

```

unit Question3_U;

interface
uses
  Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls,
  Forms, Dialogs, StdCtrls, Buttons, ExtCtrls, ComCtrls;

type
  TfrmQuestion3 = class(TForm)
    Panell: TPanel;
    GroupBox1: TGroupBox;
    btnTerminal1: TBitBtn;
    btnTerminal2: TBitBtn;
    btnTerminal3: TBitBtn;
    GroupBox2: TGroupBox;
    btnNorth: TBitBtn;
    btnSouth: TBitBtn;
    btnEast: TBitBtn;
    btnWest: TBitBtn;
    GroupBox3: TGroupBox;
    btnQues31: TButton;
    btnQues32: TButton;
    btnQues33: TButton;
    btnQues34: TButton;
    redQ3: TRichEdit;
    cmbQues3: TComboBox;
    GroupBox4: TGroupBox;
    procedure btnQues31Click(Sender: TObject);
    procedure FormActivate(Sender: TObject);
    procedure btnQues32Click(Sender: TObject);
    procedure btnTerminal1Click(Sender: TObject);
    procedure btnTerminal2Click(Sender: TObject);
    procedure btnTerminal3Click(Sender: TObject);
    procedure btnNorthClick(Sender: TObject);
    procedure btnSouthClick(Sender: TObject);
    procedure btnEastClick(Sender: TObject);
    procedure btnWestClick(Sender: TObject);
    procedure btnQues33Click(Sender: TObject);
    procedure btnQues34Click(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
//=====
// Provided code
//=====
var
  frmQuestion3: TfrmQuestion3;
  arrCodes: array [1 .. 10] of char = (
    'W',
    'A',
    'S',
    'R',
    'X',
    'D',
    'H',
    'P',
    'T',
    'L'
  );

```

```

arrActivities: array [1 .. 10] of String = (
    'Water park',
    'Aquarium',
    'Sea',
    'Restaurants',
    'Shopping',
    'Diving',
    'Help desk',
    'Penguin park',
    'Shark tank',
    'Dolphin shows'
);
arrActCodes: array [1 .. 3, 1 .. 4] of String = (('DXWAT', 'HRDST',
    'STWLP', 'RDT'), ('SWA', 'SRXD', 'LWXH', 'SHA'),
    ('WLSR', 'AT', 'DATX', 'HW'));

iTerminal: integer = 1;
iDirection: integer = 1;
//=====
// Global variables
//=====
arrDirections: array [1 .. 4] of string = (
    'North',
    'South',
    'East',
    'West'
);
//=====
// Provided code
//=====
implementation

{$R *.dfm}
procedure TfrmQuestion3.btnTerminal1Click(Sender: TObject);
begin
    iTerminal := 1;
end;

procedure TfrmQuestion3.btnTerminal2Click(Sender: TObject);
begin
    iTerminal := 2;
end;

procedure TfrmQuestion3.btnTerminal3Click(Sender: TObject);
begin
    iTerminal := 3;
end;

procedure TfrmQuestion3.btnNorthClick(Sender: TObject);
begin
    iDirection := 1;
end;

procedure TfrmQuestion3.btnSouthClick(Sender: TObject);
begin
    iDirection := 2;
end;

procedure TfrmQuestion3.btnEastClick(Sender: TObject);
begin
    iDirection := 3;
end;

```

```

procedure TfrmQuestion3.btnWestClick(Sender: TObject);
begin
    iDirection := 4;
end;
//=====
//Question 3.1
//=====
procedure TfrmQuestion3.btnQues31Click(Sender: TObject);
var
    iRow, iCol: integer;
    sLine: string;
begin
    redQ3.Lines.Add('' + #9 + 'North' + #9 + 'South' + #9 +
        'East' + #9 + 'West');
    for iRow := 1 to 3 do
    begin
        sLine := 'Terminal ' + IntToStr(iRow) + #9;
        for iCol := 1 to 4 do
        begin
            sLine := sLine + arrActCodes[iRow, iCol] + #9;
        end;
        redQ3.Lines.Add(sLine);
    end;
end;

procedure TfrmQuestion3.FormActivate(Sender: TObject);
begin
    redQ3.Paragraph.TabCount := 4;
    redQ3.Paragraph.Tab[0] := 80;
    redQ3.Paragraph.Tab[1] := 130;
    redQ3.Paragraph.Tab[2] := 180;
    redQ3.Paragraph.Tab[3] := 230;
end;
//=====
//Question 3.2
//=====
procedure TfrmQuestion3.btnQues32Click(Sender: TObject);
var
    i, j: integer;
    sCodes: string;
begin
    redQ3.Clear;
    redQ3.Lines.Add('Terminal ' + IntToStr(iTerminal) + ', ' + arrDirections
        [iDirection]);
    sCodes := arrActCodes[iTerminal, iDirection];
    for i := 1 to length(sCodes) do
    begin
        for j := 1 to length(arrCodes) do
        begin
            if sCodes[i] = arrCodes[j] then
                redQ3.Lines.Add(arrActivities[j]);
        end;
    end;
end;
end;

```

```
//=====
//Question 3.3
//=====
procedure TfrmQuestion3.btnQues33Click(Sender: TObject);
var
    iRow, iCol, iCount, iIndex: integer;
begin
    redQ3.Clear;
    iCount := 0;
    iIndex := cmbQues3.ItemIndex;
    redQ3.Lines.Add('Access routes to ' + cmbQues3.Items[cmbQues3.ItemIndex]
    );
    for iRow := 1 to 3 do
        for iCol := 1 to 4 do
            begin
                if pos(arrCodes[iIndex + 1], arrActCodes[iRow, iCol]) > 0 then
                    begin
                        redQ3.Lines.Add('Terminal ' + IntToStr(iRow) + ', ' + arrDirections
                        [iCol]);
                        Inc(iCount);
                    end;
                end;
            end;
        redQ3.Lines.Add(#13 + 'Number of access routes: ' + IntToStr(iCount));
    end;
//=====
//Question 3.4
//=====
procedure TfrmQuestion3.btnQues34Click(Sender: TObject);
var
    iIndex, iRow, iCol: integer;

begin
    redQ3.Clear;
    redQ3.Lines.Add('Updated information:');
    redQ3.Lines.Add('');
    iIndex := cmbQues3.ItemIndex;
    for iRow := 1 to 3 do
        for iCol := 1 to 4 do
            if pos(arrCodes[iIndex + 1], arrActCodes[iRow, iCol]) > 0 then
                begin
                    Delete(arrActCodes[iRow, iCol], pos(arrCodes[iIndex + 1],
                    arrActCodes[iRow, iCol]), 1);
                end;

                ShowMessage('The access routes to ' + arrActivities[iIndex + 1]
                + ' are closed.');
```

btnQues31.Click;

```
end;

end.
```