Enc_Panel.m:

```
function varargout = Enc_Panel(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Enc_Panel_OpeningFcn,
...
                   'gui_OutputFcn',  @Enc_Panel_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State,
varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
function Enc_Panel_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = Enc_Panel_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
[filename pathname]=uigetfile('File Selector');
fullpathname=strcat(pathname,filename);
global c;
c=imread(fullpathname);
axes(handles.axes1);
imshow(c);
fid = fopen('in.txt');
msg= fgetl(fid);
enc(msg,c);

function edit1_Callback(hObject, eventdata, handles)
    global message;
    message=get(hObject,'String');
    fileID = fopen('in.txt','w');
    fwrite(fileID,message);
    fclose(fileID);

function edit1_CreateFcn(hObject, eventdata, handles)
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');end
```

enc.m:

```
function enc(message,c)
  message = strtrim(message);
    m = length(message) * 8;
    AsciiCode = uint8(message);
    binaryString = transpose(dec2bin(AsciiCode,8));
    binaryString = binaryString(:);
    N = length(binaryString);
    b = zeros(N,1); %b is a vector of bits
    for k = 1:N
        if(binaryString(k) == '1')
            b(k) = 1;
        else
            b(k) = 0;
        end
    end
    s = c;
    height = size(c,1);
    width = size(c,2);
    k = 1;v=1;
    for i = 1 : height
        for j = 1 : width
             if(k<=m && (mod(j,2)==0))
             s(i,j,v)=s(i,j,v)+b(k);
             k=k+1;
             end
        end
    end
     imwrite(s, 'encrypted.bmp');
end
```

Dec Panel.m:

```
function varargout = Dec_Panel(varargin)
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @Dec_Panel_OpeningFcn,
...
                   'gui_OutputFcn',  @Dec_Panel_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end
if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State,
varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end

function Dec_Panel_OpeningFcn(hObject, eventdata, handles,
varargin)
handles.output = hObject;
guidata(hObject, handles);

function varargout = Dec_Panel_OutputFcn(hObject, eventdata,
handles)
varargout{1} = handles.output;

function pushbutton1_Callback(hObject, eventdata, handles)
[filename pathname]=uigetfile('encrypted.bmp');
fullpathname=strcat(pathname,filename);
global s;
s=imread(fullpathname);
axes(handles.axes1);
imshow(s);
dec(s);
fid = fopen('out.txt');
msg= fgetl(fid);
msg=regexprep(msg,'ÿ','');
set(handles.final_msg,'String',msg);
```

dec.m:

```
function dec(s)
    height = size(s,1);
    width = size(s,2);
    m = 24000;
    k = 1;v=1;
    for i = 1 : height
        for j = 1 : width
            if (k<=m && (mod(j,2)==0))
                b(k) = mod(double(s(i,j,v)),2);
                k = k + 1;
            end
        end
    end
    binaryVector = b;
    binValues = [ 128 64 32 16 8 4 2 1 ];
    binaryVector = binaryVector(:);
    if mod(length(binaryVector),8) ~= 0
        error('Length of binary vector must be a multiple of
8.');
    end
    binMatrix = reshape(binaryVector,8,3000);
   % display(binMatrix);
    textString = char(binValues*binMatrix);
    fileID = fopen('out.txt','w');
    fwrite(fileID,textString);
    fclose(fileID);
end
```