

Midterm 2 – “Open Book Portion” Key
BE-205, Winter '06-'07, Dr. C. S. Tritt

This portion of this test is an open book, open notes, open web. You may only have Microsoft Word, Matlab and a web browser open on your desktop while working it. When you have completed this portion of the test, e-mail you .m and .fig files and program documentation (which is required) to me. Filenames should be in the form, *lastnameMT2p2p1*. where *lastname* is your last name. There is only 1 question on this portion of the text. Partial credit will be given and in part based on the quality of your documentation.

Write a Matlab program with a GUI that calculates the strength of a metal bar to be used in a medical device. The program must be designed to allow the user to specify the dimensions of the bar (height, width and depth in cm) and select from 4 materials (brass, aluminum, stainless steel and cobalt-chrome) from which the bar can be made. Use the following (made-up) equation for calculation of the strength:

$$\text{strength (in MPa)} = k * \text{width} * \text{depth} / \text{height}$$

where the “constant” *k* has units of MPa/cm and depends on the material as shown in the following table:

Material	k (MPa/cm)
Brass	5.0
Aluminum	10.0
Stainless Steel	20.0
Cobalt-Chrome	35.0

Use the following results to test your program:

Material	Height (cm)	Width (cm)	Depth (cm)	Strength (MPa)
Brass	1.0	1.0	1.0	5.0
Aluminum	2.0	2.0	1.0	10.0
Stainless Steel	0.5	1.5	2.0	120.0
Cobalt-Chrome	5.0	1.0	0.5	3.5

Format errors -2.5

Material Dependent Bar Strength Calculator

Background and Equations

See problem statement above.

Operations Description

See problem statement above.

User Interface Description

strengthText – A static text box that displays the calculated strength.

heightEdit, widthEdit & depthEdit – Three edit boxes used for entry of the respective value. Callbacks validate and echo the inputs.

calcButton – A pushbutton that cause the strength to be calculated and displayed (in strengthText) it also disables the input edit boxes (heightEdit, widthEdit & depthEdit).

clearButton – A pushbutton that clears strength display (in strengthText) and re-enables the input edit boxes (heightEdit, widthEdit & depthEdit).

materialGroup – A button group that contains and controls the material radio buttons.

brRadio, alRadio, ssRadio, ccRadio – Four radio buttons contained in materialGroup that specify the selected material.

strengthLabel, etc. – Statics text boxes that label corresponding GUI items.

Algorithms

Opening Function (*_OpeningFcn*)

Not modified.

Input Edit Box Callbacks (*_Callback*)

Get the input string and convert it to a double.

if value is not within valid range

 Assign 'Error' to curString

else

 Assign value to curString.

Use set to echo the current string (curString).

Calculate Function (*calcButton_Callback*)

Set error flag to false.

Get, validate and echo height (called height); set error flag if necessary.

Get, validate and echo width; set error flag if necessary.

Get, validate and echo depth; set error flag if necessary.

Determine k value based on which radio button is selected.

if error

 Display error message

else

 Disable input edit boxes

 Calculate strength

 Display strength

Clear Function (*clearButton_Callback*)

Clear strength display.

Enable input edit boxes.

Source Code

```
function varargout = barStrength(varargin)
% BARSTRENGTH M-file for barStrength.fig
%     BARSTRENGTH - Calculates the strength of a bar.
% Note height is consistantly misspelled throughout this program.
%
% Created by Dr. C. S. Tritt
% Last revised 2/2/07 (version 1.0)

% Copyright 2002-2003 The MathWorks, Inc.

% Edit the above text to modify the response to help barStrength

% Last Modified by GUIDE v2.5 02-Feb-2007 09:59:12

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @barStrength_OpeningFcn, ...
                  'gui_OutputFcn',  @barStrength_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
% End initialization code - DO NOT EDIT

% --- Executes just before barStrength is made visible.
function barStrength_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to barStrength (see VARARGIN)

% Choose default command line output for barStrength
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes barStrength wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = barStrength_OutputFcn(hObject, eventdata, handles)
```

```

% varargout    cell array for returning output args (see VARARGOUT);
% hObject     handle to figure
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

function heigthEdit_Callback(hObject, eventdata, handles)
% hObject     handle to heigthEdit (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of heigthEdit as text
%         str2double(get(hObject,'String')) returns contents of heigthEdit as
a double

curValue = str2double(get(hObject,'String'));
if ~(curValue > 0.0)
    curString = 'Error';
else
    curString = sprintf('%f', curValue);
end
set(hObject, 'String', curString);

% --- Executes during object creation, after setting all properties.
function heigthEdit_CreateFcn(hObject, eventdata, handles)
% hObject     handle to heigthEdit (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function widthEdit_Callback(hObject, eventdata, handles)
% hObject     handle to widthEdit (see GCBO)
% eventdata   reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of widthEdit as text
%         str2double(get(hObject,'String')) returns contents of widthEdit as a
double

curValue = str2double(get(hObject,'String'));
if ~(curValue >= 0.0)
    curString = 'Error';
else
    curString = sprintf('%f', curValue);
end

```

```

end
set(hObject, 'String', curString);

% --- Executes during object creation, after setting all properties.
function widthEdit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to widthEdit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

function depthEdit_Callback(hObject, eventdata, handles)
% hObject    handle to depthEdit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of depthEdit as text
%         str2double(get(hObject,'String')) returns contents of depthEdit as a
double

curValue = str2double(get(hObject,'String'));
if ~(curValue >= 0.0)
    curString = 'Error';
else
    curString = sprintf('%f', curValue);
end
set(hObject, 'String', curString);

% --- Executes during object creation, after setting all properties.
function depthEdit_CreateFcn(hObject, eventdata, handles)
% hObject    handle to depthEdit (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc
    set(hObject,'BackgroundColor','white');
else
    set(hObject,'BackgroundColor',get(0,'defaultUicontrolBackgroundColor'));
end

% --- Executes on button press in calcButton.
function calcButton_Callback(hObject, eventdata, handles)
% hObject    handle to calcButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB

```

```

% handles      structure with handles and user data (see GUIDATA)

% Set up error flag
error = false;

% Retrieve and revalidate each input.
height = str2double(get(handles.heightEdit,'String'));
if ~(height > 0.0)
    curString = 'Error';
    error = true;
else
    curString = sprintf('%f', height);
end
set(handles.heightEdit, 'String', curString);

width = str2double(get(handles.widthEdit,'String'));
if ~(width >= 0.0)
    curString = 'Error';
    error = true;
else
    curString = sprintf('%f', width);
end
set(handles.widthEdit, 'String', curString);

depth = str2double(get(handles.depthEdit,'String'));
if ~(depth >= 0.0)
    curString = 'Error';
    error = true;
else
    curString = sprintf('%f', depth);
end
set(handles.depthEdit, 'String', curString);

if get(handles.brRadio, 'Value') == 1
    k = 5.;
elseif get(handles.alRadio, 'Value') == 1
    k = 10.;
elseif get(handles.ssRadio, 'Value') == 1
    k = 20.;
elseif get(handles.ccRadio, 'Value') == 1
    k = 35.;
else
    k = 0.; % Should never happen.
end

if error
    set(handles.strengthText, 'String', 'Error');
else
    % Disable input controls only if strength can be calculated
    set(handles.heightEdit, 'Enable', 'off');
    set(handles.widthEdit, 'Enable', 'off');
    set(handles.depthEdit, 'Enable', 'off');

    strength = k*width*depth/height;
    curString = sprintf('%f', strength);
    set(handles.strengthText, 'String', curString);
end

```

```
% --- Executes on button press in clearButton.
function clearButton_Callback(hObject, eventdata, handles)
% hObject    handle to clearButton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Clear output and re-enable inputs.

set(handles.strengthText, 'String', '');
set(handles.heigthEdit, 'Enable', 'on');
set(handles.widthEdit, 'Enable', 'on');
set(handles.depthEdit, 'Enable', 'on');
```