CODING ASSIGNMENT - FUNCTIONS & PARAMETER PASSING WITH POINTERS

To view the assignment submission instructions for this assignment, see the Assignment 8 web page at:

http://www.gibsonr.com/classes/cop2000/fall/a9.html

ANALYSIS DOCUMENTATION

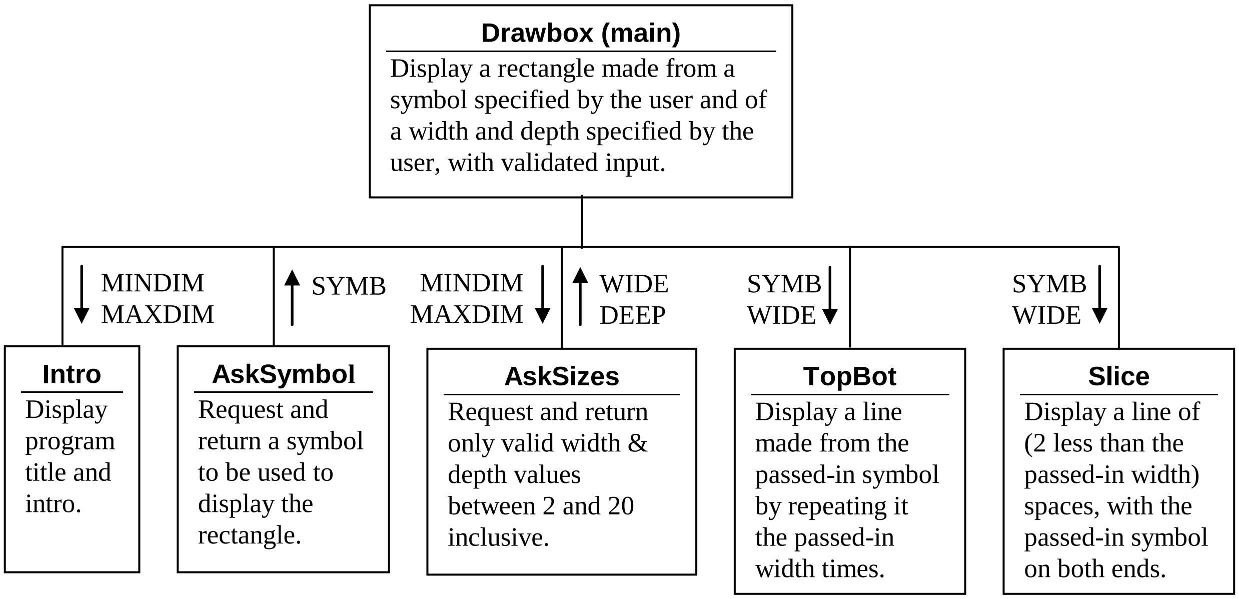
SITUATION

You want to display a rectangle made from a symbol specified by the user (such as a star) and of a size (width and depth in symbols) specified by the user, with validated input within a limited range.

SYMBOLIC CONSTANT LIST (GLOBAL)

LABEL	DESCRIPTION	DATA TYPE	VALUE	USAGE	DESTINATION
MINDIM	Minimum dimension of rectangle	Integer	2	---	Intro & AskSizes
MAXDIM	Maximum dimension of rectangle	Integer	20	---	Intro & AskSizes

STRUCTURE CHART (GLOBAL)



Drawbox (main)

Display a rectangle made from a symbol specified by the user and of a width and depth specified by the user, with validated input.

Intro

Display program title and intro.

AskSymbol

Request and return a symbol to be used to display the rectangle.

AskSizes

Request and return only valid width & depth values between 2 and 20 inclusive.

TopBot

Display a line made from the passed-in symbol by repeating it the passed-in width times.

Slice

Display a line of (2 less than the passed-in width) spaces, with the passed-in symbol on both ends.

See the notes in the documentation on following pages for function AskSizes that discuss the technique that will be used to return multiple parameters. Because of our choice of the C++ programming language and its ability to pass pointers, we will be passing pointers (address data) *into* the function AskSizes and then use them to return two items of data to main's variables WIDE and DEEP. However, because this passage of pointers *into* the function is based on a coding decision rather than a logical requirement of the program, that passage of data *into* the function is not shown on the diagram.

FUNCTION "main"

PROBLEM STATEMENT

Display a rectangle made from a symbol specified by the user and of a width and depth specified by the user, with validated input.

SAMPLE SOFTCOPY

In the sample output below:

- The numbers to the left of the sample screen are there only for your reference in other documentation
- Bracketed text indicates examples of user responses to prompts
- The words "INVALID ENTRY" (see example) must be accompanied by a beep (alert tone).

```
 1  BOX DISPLAYING PROGRAM
 2
 3  This program will display a rectangle made from a symbol
 4  specified by the user and of a width and depth specified
 5  by the user. The program will accept user input of width
 6  and depth values between 2 and 20 (inclusive) only.
 7
 8  Symbol to display? [*]
 9  Width (2-20)? [25]
10  INVALID ENTRY: Enter a value between 2 and 20 inclusive -
11  Width (2-20)? [5]
12  Depth (2-20)? [3]
13
14  *****
15  *   *
16  *****
```

VARIABLE LIST

LABEL	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
C	Counter of number of times to repeat the Slice function	Integer	Set to 1	Loop index	-
SYMB	Symbol to draw the rectangle	Character	Function AskSymbol	-	Function TopBot&Slice
WIDE	Width of rectangle	Integer	Function AskSizes	-	Function TopBot&Slice
DEEP	Depth of rectangle	Integer	Function AskSizes	Loop Limit	-

(continued on next page)

FUNCTION "main" continued

ALGORITHM

```
A.  Start.
B.  Perform Intro (passing in the values MINDIM and MAXDIM).
C.  Request and store valid input data.
    C.1. Perform AskSymbol and assign its returned value into SYMB.
    C.2. Perform AskSizes (passing in the values MINDIM and MAXDIM, and receiving back WIDE and DEEP). Note: the accomplishment of the reception of WIDE and DEEP in the C++ Programming Language will require that the main function also pass the addresses of those receiving variables into the function AskSizes.
D.  Display a blank line.
E.  Perform TopBot (sending it SYMB and WIDE).
F.  Use a counting loop to perform the Slice function (sending it send SYMB and WIDE) DEEP-2 times.
    F.1. Set C to 1
    F.2. While C is less than or equal to DEEP-2, do the following:
        F.2.a. Perform the Slice function (sending it send SYMB and WIDE).
        F.2.b. Increment C by 1
G.  Perform TopBot (sending it SYMB and WIDE).
H.  End.
```

DESK CHECK

Note: The desk check for the main algorithm could not be performed until all of the child functions had been developed. See the Desk Check for the main function on page 10.

FUNCTION "Intro"

PROBLEM STATEMENT:

Display the program title and introduction as shown in the first seven lines of the Sample Softcopy using the passed dimension parameter values on line 6 instead of the literal characters "'2" and "20".

SAMPLE SOFTCOPY

In the sample output below, the numbers to the left of the sample screen are there only for your reference in other documentation

```
1  BOX DISPLAYING PROGRAM
2
3  This program will display a rectangle made from a symbol
4  specified by the user and of a width and depth specified
5  by the user. The program will accept user input of width
6  and depth values between 2 and 20 (inclusive) only.
7
```

PARAMETER LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
MIN	Minimum dimension of rectangle	Integer	Passed in	---	Screen
MAX	Maximum dimension of rectangle	Integer	Passed in	---	Screen

ALGORITHM

A. Start.
B. Display the program title (lines 1-2 on Sample Softcopy).
C. Display the program introduction (lines 3-7 on Sample Softcopy) using the passed dimension parameter values on line 6 instead of the literal characters "'2" and "20".
D. End.

TEST SOFTCOPY

```
1  BOX DISPLAYING PROGRAM
2
3  This program will display a rectangle made from a symbol
4  specified by the user and of a width and depth specified
5  by the user. The program will accept user input of width
6  and depth values between 2 and 20 (inclusive) only.
7
```

FUNCTION "AskSymbol"

PROBLEM STATEMENT:

Request and return a symbol to be used to display the rectangle.

SAMPLE SOFTCOPY

In the sample output below:

- The numbers to the left of the sample screen are there only for your reference in other documentation
- Bracketed text indicates examples of user responses to prompts

```
1   Symbol to display? [*]
```

LOCAL VARIABLE LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
S	Symbol to draw the rectangle	Character	Requested	-	Returned

ALGORITHM

A. Start.
B. Prompt for the symbol to draw the rectangle as shown on the Sample Softcopy.
C. Store the response in S.
D. Return value S to the parent process.
E. End

DATA TRACING CHART

Step	S
C	*

TEST SOFTCOPY

```
1   Symbol to display? [*]
```

FUNCTION "AskSizes"

PROBLEM STATEMENT:

Request and return only valid width & depth values between inclusive dimension limits that are passed into the function.

SAMPLE SOFTCOPY

In the sample output below:

- The numbers to the left of the sample screen are there only for your reference in other documentation
- Bracked text indicates examples of user responses to prompts
- The words "INVALID ENTRY" (see example) must be accompanied by a beep (alert tone).

```
1  Width (2-20)? [25]
2  INVALID ENTRY: Enter a value between 2 and 20 inclusive -
3  Width (2-20)? [5]
4  Depth (2-20)? [3]
```

LOCAL VARIABLE LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
D	Depth of rectangle	Integer	Requested	-	Returned
W	Width of rectangle	Integer	Requested	-	Returned

PARAMETER LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DEST.
MIN	Minimum dimension of rectangle	Integer	Passed in	Compared	Screen
MAX	Maximum dimension of rectangle	Integer	Passed in	Compared	Screen
DPTR	Pointer to parent's depth parameter	Integer Pointer	Passed in	Indirection	---
WPTR	Pointer to parent's width parameter	Integer Pointer	Passed in	Indirection	---

Note:
The parameters MIN and MAX above are *input parameters*, received by the function from its parent.
The parameters DPTR and WPTR are *output parameters*, needed to receive and store pointers to variables outside of this function that will be used to return the values from local variable D and W back to variables declared by the parent function (main). The need to use these pointers to return the two values arises because of our choice of programming language (in our case, C++ – which offers excellent manipulation of pointers). So technically speaking, this Parameter List relates to a *coding* issue and would not be included in the documentation of the analysis. It has been included here to assist students in coding this function because (ultimately) a formal parameter list will be used in the function header. If you refer back to the Structure Diagram, you will notice that these output parameters do not appear as values being passed into this function. This is because the need for the passage of the output parameters is not a *logical requirement* (derived from the purpose) of this program; rather it is a physical one (based on choice of language to implement the analysis). The inclusion of the Parameter List in the overall documentation of the project is reasonable, but it is not actually a part of the analysis.

FUNCTION "AskSizes" continued

ALGORITHM *(with hints in italics)*

```
A.  Start.
B.  Request and store a valid width.  Do: (Trailing Decision Loop with Selection/Guard)
    B.1. Prompt for the width within a valid range as shown on line 1 of the Sample Softcopy.
    B.2. Store the response in W.
    B.3 If W is less than MIN or W is greater than MAX, then display an error message as shown on line
        2 of the Sample Softcopy and make the computer beep.
    B.4. Go back to step B1 while W is less than MIN or W is greater than MAX.
C.  Request and store a valid depth. (Leading Decision Loop with a Prime Read)
    C.1. Prompt for depth within a valid range as shown on line 4 of the Sample Softcopy.
    C.2. Store the response in D. (Prime Read)
    C.3. While D is less than MIN or D is greater than MAX:
        C.3.a. Display an error message as shown on line 2 of the Sample Softcopy and output a beep.
        C.3.b. Prompt for the depth within a valid range as shown on line 4 of the Sample Softcopy.
        C.3.c. Store the response in D. (Next Read)
D.  Return values W and D to the parent process (using indirection with pointers).
E.  End
```

DATA TRACING CHART

Step	**MIN**	**MAX**	***DPTR***	***WPTR***	**W**	**D**	**(Step B3) W<MIN or W>MAX**	**(Step B4) W<MIN or W>MAX**	**(Step C3) D<MIN or D>MAX**
A	2	20	*Adr.**	*Adr.**					
B2					25				
B3							True		
B4								True	
B2					5				
B3							False		
B4								False	
C2						3			
C3									False

** - The two notations "Adr." in the table above indicate memory address values for the variables in the parent function that will receive the values in D and W by "indirection" at the end of this process. Because the use of these "pointers" is a requirement of the programming language C++ (to support the return of multiple values), they normally would not appear in the Tracing Chart. They are included here to illustrate to the student reader where they would be appear in the process.*

TEST SOFTCOPY

In the test output below the numbers to the left of the sample screen are there only for your reference in other documentation and the words "INVALID ENTRY" were accompanied by a beep (alert tone)

```
1   Width (2-20)? [25]
2   INVALID ENTRY: Enter a value between 2 and 20 inclusive -
3   Width (2-20)? [5]
4   Depth (2-20)? [3]
```

FUNCTION "TopBot"

PROBLEM STATEMENT:

Display a line made from the passed-in symbol by repeating it the passed-in width times.

SAMPLE SOFTCOPY

In the sample output below the numbers above and to the left of the sample screen are there only for your reference in other documentation.

```
   123456789
1  *****
```

PARAMETER LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
W	Width of rectangle	Integer	Passed in	Loop Limit	-
S	Symbol to draw the rectangle	Character	Passed in	-	Displayed

VARIABLE LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
I	Counting Loop Index	Integer	Set to 1	Loop Index	-

ALGORITHM

```
A.  Start.
B.  Use a counting loop to display the passed-in symbol, W times.
        B.1. Set I to 1
        B.2. While I is less than or equal to W, do the following:
                B.2.a. Display S
                B.2.b. Increment I by 1
C. Display a new line (in C++ a '\n' character or an endl stream object).
D. End.
```

DATA TRACING CHART

Step	S	W	I	I<=W
A	*	5		
B1			1	
B2				True
B2b			2	
B2				True
B2b			3	
B2				True
B2b			4	
B2				True
B2b			5	
B2				True
B2b			6	
B2				False

TEST SOFTCOPY

```
   123456789
1  *****
```

FUNCTION "Slice"

PROBLEM STATEMENT:

Display a line of (2 less than the passed-in width) spaces, with the passed-in symbol on both ends.

SAMPLE SOFTCOPY

In the sample output below the numbers above and to the left of the sample screen are there only for your reference in other documentation.

```
     123456789
  1  *   *
```

PARAMETER LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
W	Width of rectangle	Integer	Passed in	Loop Limit	-
S	Symbol to draw the rectangle	Character	Passed in	-	Displayed

VARIABLE LIST

IDENTIFIER	DESCRIPTION	DATA TYPE	SOURCE	USAGE	DESTINATION
I	Counting Loop Index	Integer	Set to 1	Loop Index	-

ALGORITHM

```
A.  Start.
B.  Display the passed symbol S, staying on the same line.
C.  Use a counting loop to display W-2 blank spaces, staying on the same line..
        C.1. Set I to 1
        C.2. While I is less than or equal to W-2, do the following:
                C.2.a. Display a blank space.
                C.2.b. Increment I by 1
D.  Display the passed symbol S and then start a new line ( '\n' in C++ ).
E.  End.
```

DATA TRACING CHART

Step	S	W	I	I<=W-2
A	*	5		
C1			1	
C2				True
C2b			2	
C2				True
C2b			3	
C2				True
C2b			4	
C2				False

TEST SOFTCOPY

The numbers to the left of the sample screen are there only for your reference in other documentation:

```
     123456789
  1  *   *
```

DESK CHECK FOR FUNCTION "main"

DATA TRACING CHART

Step	SYMB	WIDE	DEEP	C	C <= DEEP-2
C1	*				
C2		5	3		
F1				1	
F2					True
F2b				2	
F2					False

TEST SOFTCOPY

In the test output below:

- The numbers to the left of the sample screen are there only for your reference in other documentation
- Bracketed text indicates examples of user responses to prompts
- The words "INVALID ENTRY" (see example) must be accompanied by a beep (alert tone).

```
 1  BOX DISPLAYING PROGRAM
 2
 3  This program will display a rectangle made from a symbol
 4  specified by the user and of a width and depth specified
 5  by the user. The program will accept user input of width
 6  and depth values between 2 and 20 (inclusive) only.
 7
 8  Symbol to display? [*]
 9  Width (2-20)? [25]
10  INVALID ENTRY: Enter a value between 2 and 20 inclusive -
11  Width (2-20)? [5]
12  Depth (2-20)? [3]
13
14  *****
15  *   *
16  *****
```