Functions: Part 3 of 3

CMSC 104, Spring 2014

Christopher S. Marron

(thanks to John Park for slides)

Functions, Part 3 of 3

Topics

- In-Class Coding Practice
  - Project 1: The Box
  - Project 2: Drawing a Rectangle

Reading

- none

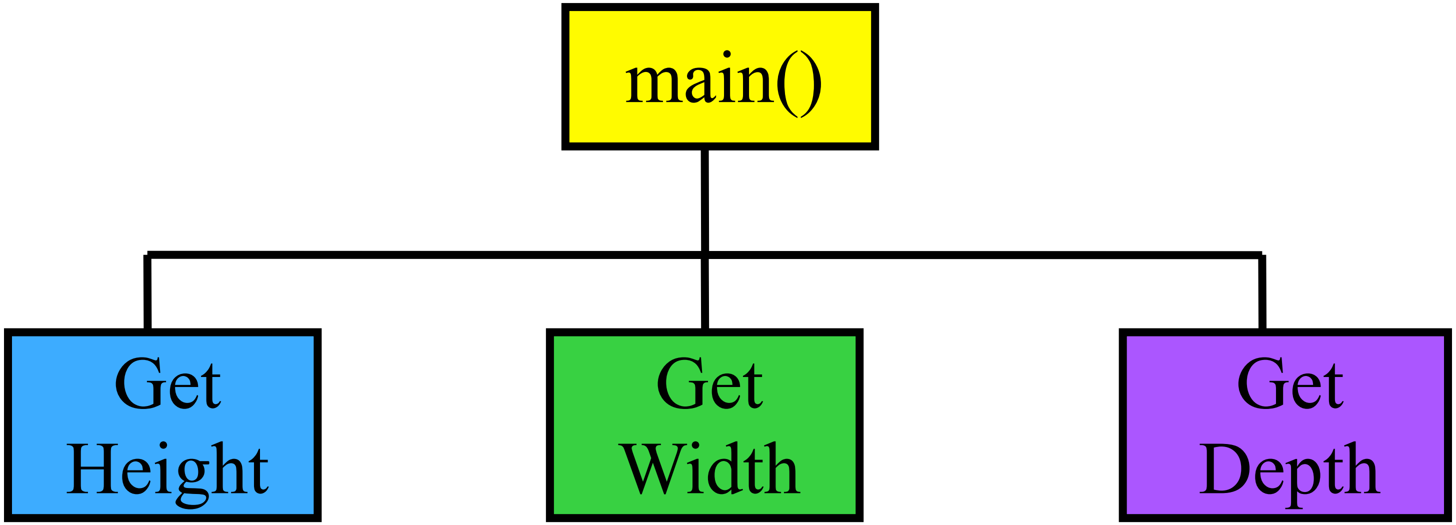
Coding Practice

- Starting with some simple problems, we will:
  1. Design appropriate algorithms
  2. Modularize them
  3. Create pseudocode
  4. Write actual C code

The Box

Problem: Write an interactive program to compute and display the volume and surface area of a box. The program must also display the box dimensions. Error checking should be done to be sure that all box dimensions are greater than zero.

Hierarchy Chart

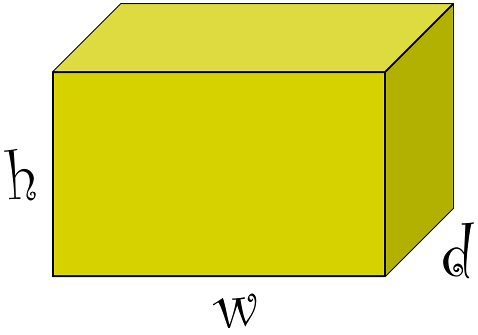


main()

Get Height

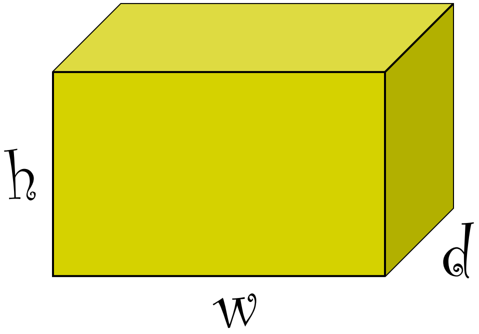
Get Width

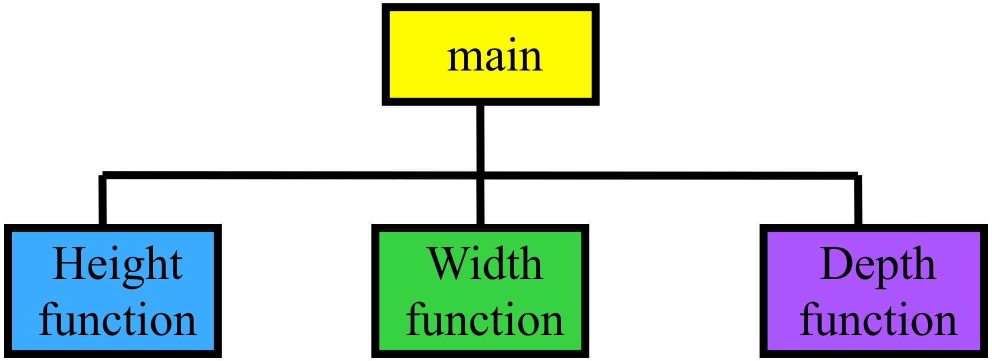
Get Depth



The Box – Pseudocode for get height function

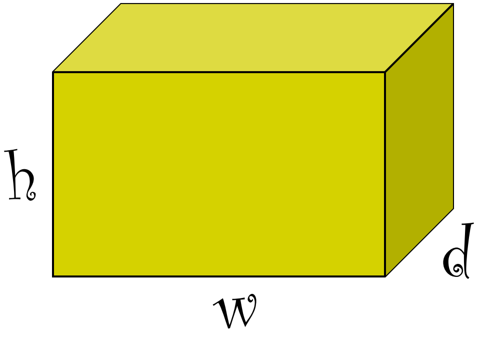
```
Display "Enter the height:  "
Read <height>
While (<height>  <=  0 )
    Display "The height must be > 0"
    Display "Enter the height:  "
    Read <height>
End_while
Return height
```

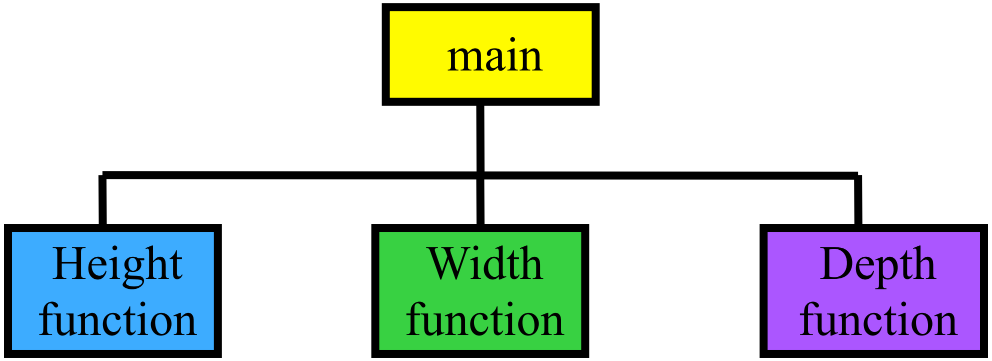




The Box - Pseudocode for get width function

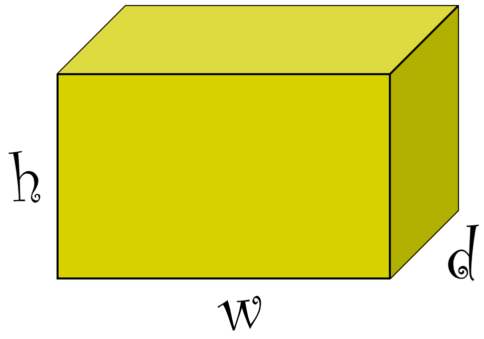
```
Display "Enter the width:  "
Read <width>
While (<width>  <=  0 )
    Display "The width must be > 0"
    Display "Enter the width:  "
    Read <width>
End_while
Return width
```

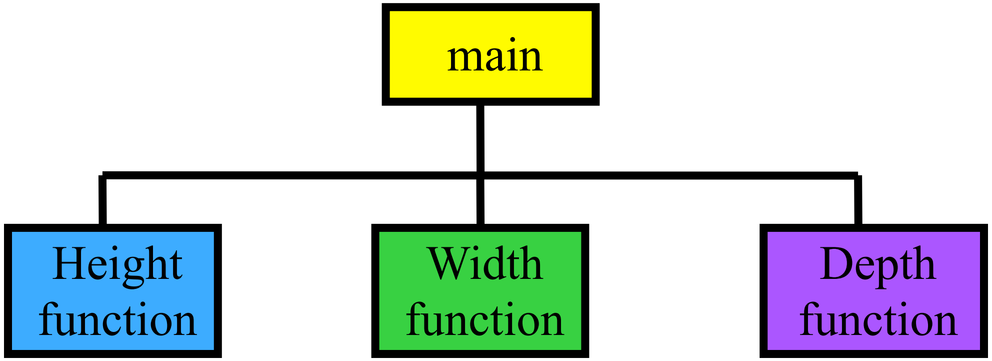




The Box – Pseudocode for get depth function

```
Display “Enter the depth:  “
Read <depth>
While (<depth>  <=  0 )
    Display “The depth must be > 0”
    Display “Enter the depth:  “
    Read <depth>
End_while
Return depth
```





main

Height function

Width function

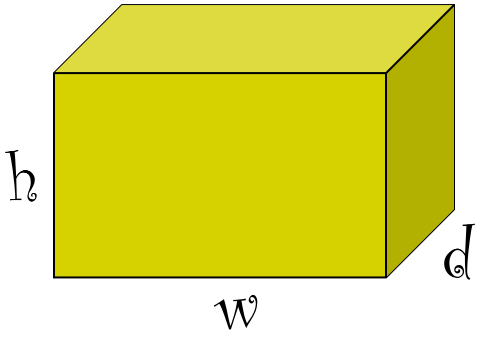
Depth function

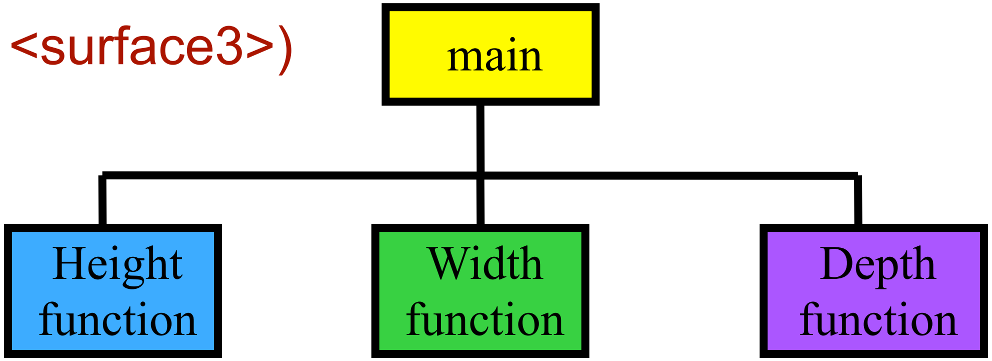
The Box - Pseudocode (cont.)

```
Call get_height saving answer in <height>
Call get_width saving answer in <width>
Call get_depth saving answer in <depth>

<volume> = <height>  X  <width>  X  <depth>

<surface1> = <height>  X  <width>
<surface2> = <width>  X  <depth>
<surface3> = <height>  X  <depth>
<surface area> =  2  X  (<surface1>  +  <surface2>
                          +  <surface3>)
```





The Box - Pseudocode (cont.)

```
Display “Height = “, <height>
Display “Width = “, <width>
Display “Depth = “, <depth>
Display “Volume = “, <volume>
Display “Surface Area = “, <surface area>
```

Code the Design

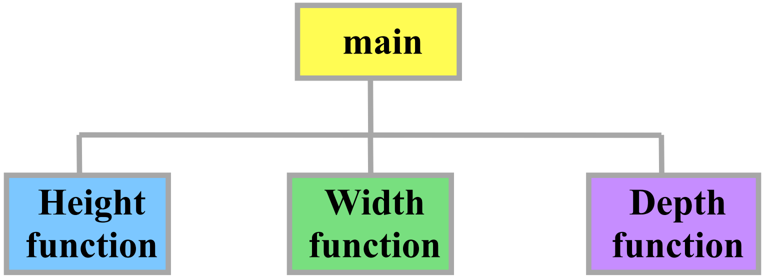
```
#include <stdio.h>
    int get_height( void );
    int get_width( void );
    int get_depth( void );
int main( void )
{
    int height, width, depth, volume;
    int surface1, surface2, surface3, surface_area;

    height = get_height( );
    width =  get_width( );
    depth = get_depth( );

    volume = height * width * depth;

    surface1 = height * width;
    surface2 = width * depth;
    surface3 = height * depth;
    surface_area = 2 * (surface1 + surface2 + surface3);

    printf( "Height = %d\n", height );
    printf( "Width = %d\n", width );
    printf( "Depth = %d\n", depth );
    printf( "Volume = %d\n", volume );
```

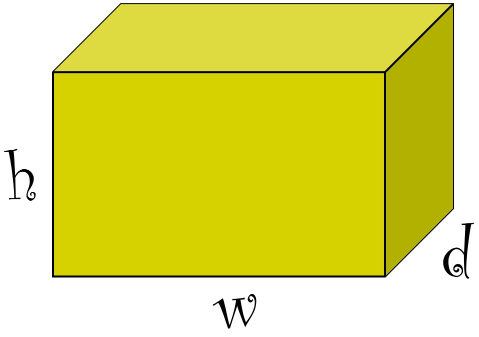


main

Height function

Width function

Depth function

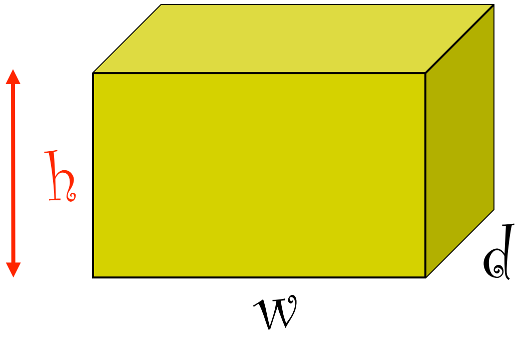


get_height( )

```
int get_height( void )
{
    int height;

    printf( "Enter the height:  " );
    scanf( "%d", &height);

    while( height <= 0 )
    {
        printf( "The height must be > 0\n" );
        printf( "Enter the height:  " );
        scanf( "%d", &height);
    }
    return height;
}
```



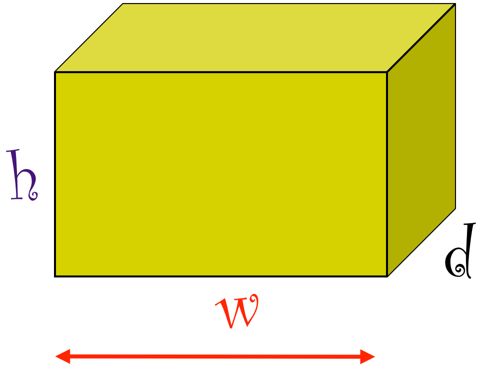
get_width( )

```
int get_width( void )
{
    int width;

    printf( "Enter the width:  " );
    scanf( "%d", &width );

    while( width <= 0 )
    {
        printf( "The width must be > 0" );
        printf( "Enter the width:  " );
        scanf( "%d", &width );
    }

    return width;
}
```



get_depth( )

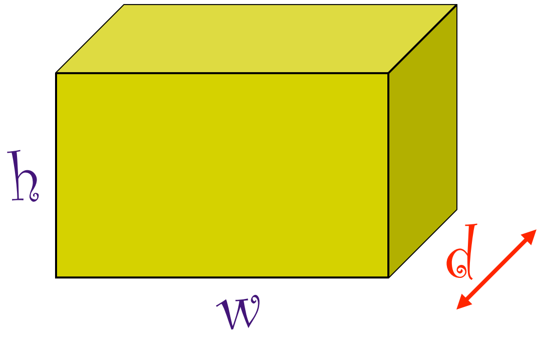
```
int get_depth( void )
{
    int depth;

    printf( "Enter the depth:  " );
    scanf( "%d", &depth );

    while( depth <= 0 )
    {
        printf( "The depth must be > 0" );
        printf( "Enter the depth:  " );
        scanf( "%d", &depth );


    }

    return depth;
}
```

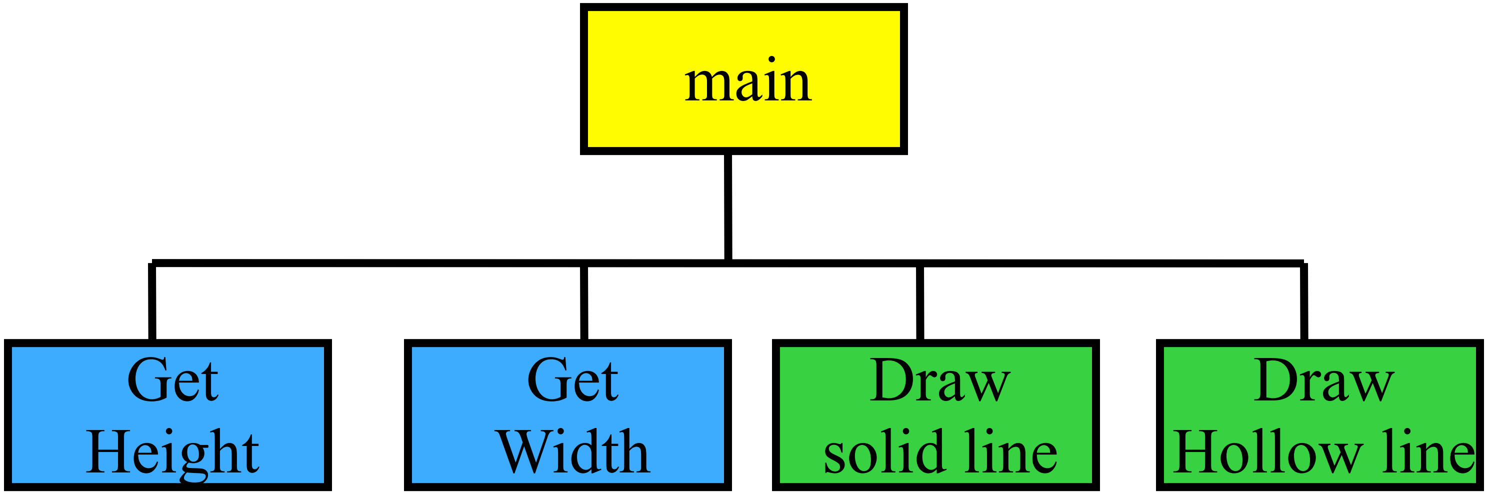


Drawing a Rectangle

Problem: Write an interactive program that will draw a solid rectangle of asterisks (*). The program must also display the dimensions of the rectangle. Error checking must be done to be sure that the dimensions are greater than zero.

```
* * * * * * * * * * * * *
*                       *
*                       *
* * * * * * * * * * * * *
```

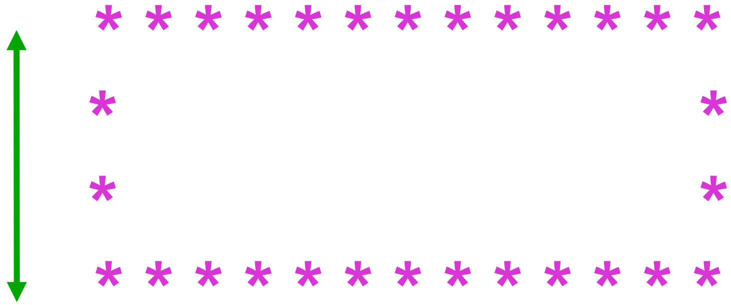
Hierarchy Chart



```
* * * * * * * * * * * * *
*                       *
*                       *
* * * * * * * * * * * * *
```

The Rectangle – Pseudocode for get_height

```
Display “Enter the height:  “
Read <height>
While (<height>  <=  0 )
    Display “The height must be > 0”
    Display “Enter the height:  “
    Read <height>
End_while
Return <height>
```



```
* * * * * * * * * * * * *
*                       *
*                       *
* * * * * * * * * * * * *
```

The Rectangle - Pseudocode for get_width

```
Display “Enter the width:  “
Read <width>
While (<width>  <=  0 )
    Display “The width must be > 0”
    Display “Enter the width:  “
    Read <width>
End_while
return <width>
```

The Rectangle – Pseudocode function Draw_solid_line

```
Receive width_size
Set I to 0
While ( I < width_size )
        Display "*"
        add 1 to I
Display "\n"
```

The Rectangle – Pseudocode function Draw_hollow_line

```
Receive <width_size>
Display “*”
Set I to 0
While ( I < <width_size> - 2 )
        Display “ ”
        add 1 to I
Display “*\n”
```

The Rectangle - Pseudocode main function

```
Call **get_height** saving answer in <height>
Call **get_width** saving answer in <width>
Skip a line
```

The Rectangle - Pseudocode (cont.)

```
Call Draw_solid_line sending <width>
Set height_counter to 1
While ( <height counter>  <=  <height - 2> )
     call Draw_hollow_line sending width
   <height counter> = <height counter> + 1
End_while
Call Draw_solid_line sending width
```

The Rectangle Code

```
#include <stdio.h>
int get_height( void );
int get_width( void );
void draw_solid_line( int width_size );
void draw_hollow_line( int width_size );
```

```
int main( void )
{
    int height;
    int     width;
    int height_counter;

    height = get_height(   );
    width = get_width(   );
    printf( "\n" );

    draw_solid_line( width );
    height_counter = 1;

    while ( height_counter < ( height - 2 ) )
    {
          draw_hollow_line( width );
          height_counter++;
    }
```

get_height( ) – software reuse

```
int get_height( void )
{
    int height;

    printf( "Enter the height:  " );
    scanf( "%d", &height);

    while( height <= 0 )
    {
        printf( "The height must be > 0\n" );
        printf( "Enter the height:  " );
        scanf( "%d", &height);
    }
    return height;
}
```

get_width( ) – software reuse

```
int get_width( void )
{
    int width;

    printf( "Enter the width:  " );
    scanf( "%d", &width );

    while( width <= 0 )
    {
        printf( "The width must be > 0" );
        printf( "Enter the width:  " );
        scanf( "%d", &width );
    }

    return width;
```

draw_solid_line( )

```
void draw_solid_line( int width_size )
{
    int i;

    i = 0;

    while ( i < width_size )
    {
          printf( "*" );
          i++;
    }
    printf( "\n" );

}
```

draw_hollow_line( )

```
void draw_hollow_line( int width_size )
{
    int i;

    printf( "*" );
    i = 0;

    while( i < ( width_size - 2 ) )
    {
        printf( " " );
        i++;
    }
    printf( "*\n" );

}
```