Accessing Image Sub-Regions

Matlab has many in-built image processing functions and sample images that we incorporate in the programming problems. In this problem, we will use the 'Ca is one of the most popular standard grayscale test images of size 256 x 256, owned by MIT. For many Computer Vision and Image Processing tasks, you shou with accessing Image sub-regions.

Your task is to extract 50x50 image sub-regions from the top-left as well as the bottom right and store them in the variables **subimg1** and **subimg2** respectively

After you successfully extract **subimg1** and **subimg2** images, compute the SSD (Sum of Squared Differences) of the intensities between them and store in the

Script

Save Reset MATLAB Documentation (https://www.mathworks.com/help/)

```
img = imread('cameraman.tif');
subimg1 = img(1:50,1:50);
subimg2 = img(  end -49 :end,  end -49 :end);
SSD = sum(sum((double(subimg1) - double(subimg2)).^2));
SSD
```

Run Script

Output

```
SSD =

     6529105
```

Previous Assessment: All Tests Passed

- **Validating subimg1**
- **Validating subimg2**
- **Validating SSD**

Color Imaging - RGB Channels

Sergei Mikhailovich Prokudin-Gorskii (1863-1944) was a photographer who, between the years 1909-1915, traveled the Russian empire and took thousands o everything he saw. He used an early color technology that involved recording three exposures of every scene onto a glass plate using a red, green, and blue fi was no way to print such photos, and they had to be displayed using a special projector.

The goal of this assignment is to learn to work with images in Matlab by taking the digitized Prokudin-Gorskii glass plate images and automatically producing a order to do this, you will need to extract the three color channel images, place them on top of each other, so that they form a single RGB color image. Your pro glass plate image (https://lcms-files.mathworks.com/content/file/6c1e8eb5-60e2-444c-a584-559ef26ce5a6/image.jpg?versionId=we1PYUROoVB..TeJmkJNw4 input and produce a single color image as output.

The top most image belongs to Blue channel. The middle image belongs to Green channel and the bottom image belongs to Red channel.

The program should divide the image into three equal parts and place the second and the third parts (G and R) on to the first (B)

You have to name your variables as below

Blue channel Image - `B`

Green channel Image - `G`

Red channel Image - `R`

Concatenated Color Image - `ColorImg`

Download Image (https://lcms-files.mathworks.com/content/file/b44625b3-25ca-4366-a4c3-2f9b1dfa0611/image.jpg?versionId=8oPXPkcVedswvscE2ZR7p3N

A sample image from the collection-

Credits: Dr. Alexei A. Efros, Dr. Svetlana Lazebnik

Script

Save Reset MATLAB Documentation (https://www.mathworks.com/help/)

```
%Read the image
img = imread('image.jpg');

%Get the size (rows and columns) of the image
[r,c] = size(img);
rr=r/3;
%Wrire code to split the image into three equal parts and store them in B, G, R channels

B=imcrop(img,[1,1,c,rr-1]);
size(B)
```

Run Script

Output

```
ans =

   341   400


ans =

   341   400


ans =

   341   400
```

Previous Assessment: All Tests Passed

- Validating Blue Channel Image - B
- Validating Red Channel Image - R
- Validating Green Channel Image - G
- Validating Color Image - ColorImg

Image Gradient Magnitude

The image gradient magnitude corresponds to the strength of edges in any given image.

In this problem, you will evaluate the Sobel Image Gradients **Gx** and **Gy** for the 'cameraman.tif' image and subsequently Gradient magnitude **Gmag** and Gradie

Reference: https://www.mathworks.com/help/images/ref/imgradient.html (https://www.mathworks.com/help/images/ref/imgradient.html)

Read the 'cameraman.tif' image into variable **img**

Compute **Gx** and **Gy** using `imgradientxy` function

Compute **Gmag** and **Gdir** using `imgradient` function

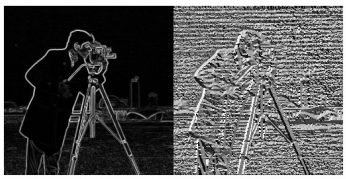
Script

Save Reset MATLAB Documentation (https://www.mathworks.com/help/)

```
1
2  img=imread('cameraman.tif');
3  [Gx,Gy]=imgradientxy(img);
4  [Gmag,Gdir]=imgradient(Gx,Gy);
5  %Uncomment the code below to visualize Gx and Gy
6  imshowpair(Gx,Gy,'montage')
7
8  %Uncomment the code below to visualize Gmag and Gdir
9  imshowpair(Gmag,Gdir,'montage')
10
11
12
```

Run Script

Output



Previous Assessment: All Tests Passed (100%)

✓ **Validating img**	**20%**	(20%)
✓ **Validating Gx**	**20%**	(20%)
✓ **Validating Gy**	**20%**	(20%)
✓ **Validating Gmag**	**20%**	(20%)
✓ **Validating Gdir**	**20%**	(20%)

Total: 100%

Simulation Study 1

Analyzed File	Air Cylinder Assembly - PE v3
Version	Autodesk Fusion 360 (2.0.8950)
Creation Date	2020-09-26, 02:14:32
Author	Sterling James Richard
Summary	In this lesson, Fusion360 was utilized to create simulations and G-code for CNC machining. These integrated tools help complete the design for manufacturing process using the design, engineering and manufacturing tools in Fusion 360.

Project Properties

Title	Studies
Author	J

Simulation Model 1:1

Study 1 - Static Stress

Study Properties

Study Type	Static Stress
Last Modification Date	2020-09-14, 00:56:49

Settings

General

Contact Tolerance	0.1 mm
Remove Rigid Body Modes	No

Damping

Mesh

Average Element Size (% of model size)	
Solids	10
Scale Mesh Size Per Part	No
Average Element Size (absolute value)	-
Element Order	Parabolic
Create Curved Mesh Elements	Yes
Max. Turn Angle on Curves (Deg.)	60
Max. Adjacent Mesh Size Ratio	1.5
Max. Aspect Ratio	10
Minimum Element Size (% of average size)	20

Adaptive Mesh Refinement

Number of Refinement Steps	0
Results Convergence Tolerance (%)	20
Portion of Elements to Refine (%)	10
Results for Baseline Accuracy	Von Mises Stress

Materials

Component	Material	Safety Factor
Link:1	ABS Plastic	Yield Strength
Gripper:1	ABS Plastic	Yield Strength

ABS Plastic

Density	1.06E-06 kg / mm^3
Young's Modulus	2240 MPa
Poisson's Ratio	0.38
Yield Strength	20 MPa
Ultimate Tensile Strength	29.6 MPa
Thermal Conductivity	1.6E-04 W / (mm C)
Thermal Expansion Coefficient	8.57E-05 / C
Specific Heat	1500 J / (kg C)

Contacts

Bonded

Name
[S] Bonded1 [Link:1||Gripper:1]
[S] Bonded2 [Link:1||Gripper:1]
[S] Bonded3 [Link:1||Gripper:1]
[S] Bonded4 [Link:1||Gripper:1]

Mesh

Type	Nodes	Elements
Solids	6943	3568

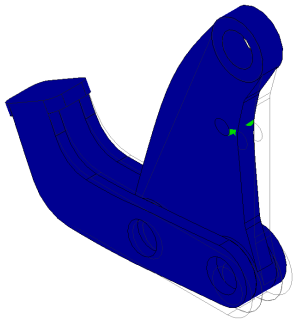
Load Case1

Results

Safety Factor

Safety Factor (Per Body)

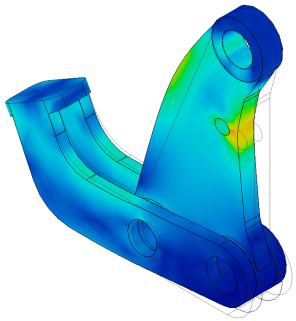
0 8



Stress

Von Mises

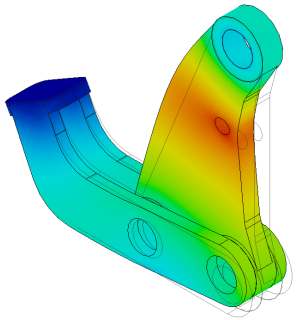
[MPa] 0.002 4.07



Displacement

Total

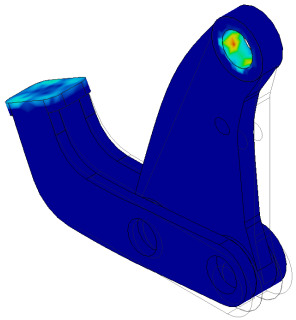
[mm] 0 0.05259



Reaction Force

Total

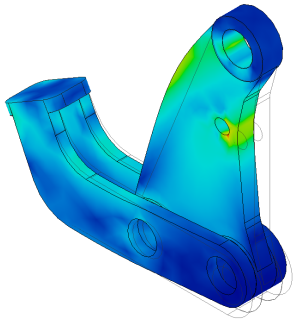
[N] 0 8.801



Strain

Equivalent

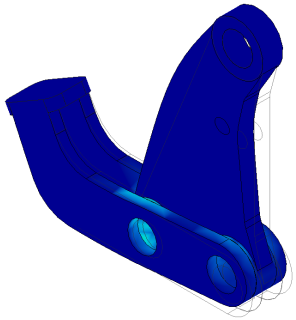
0.000001 0.002319

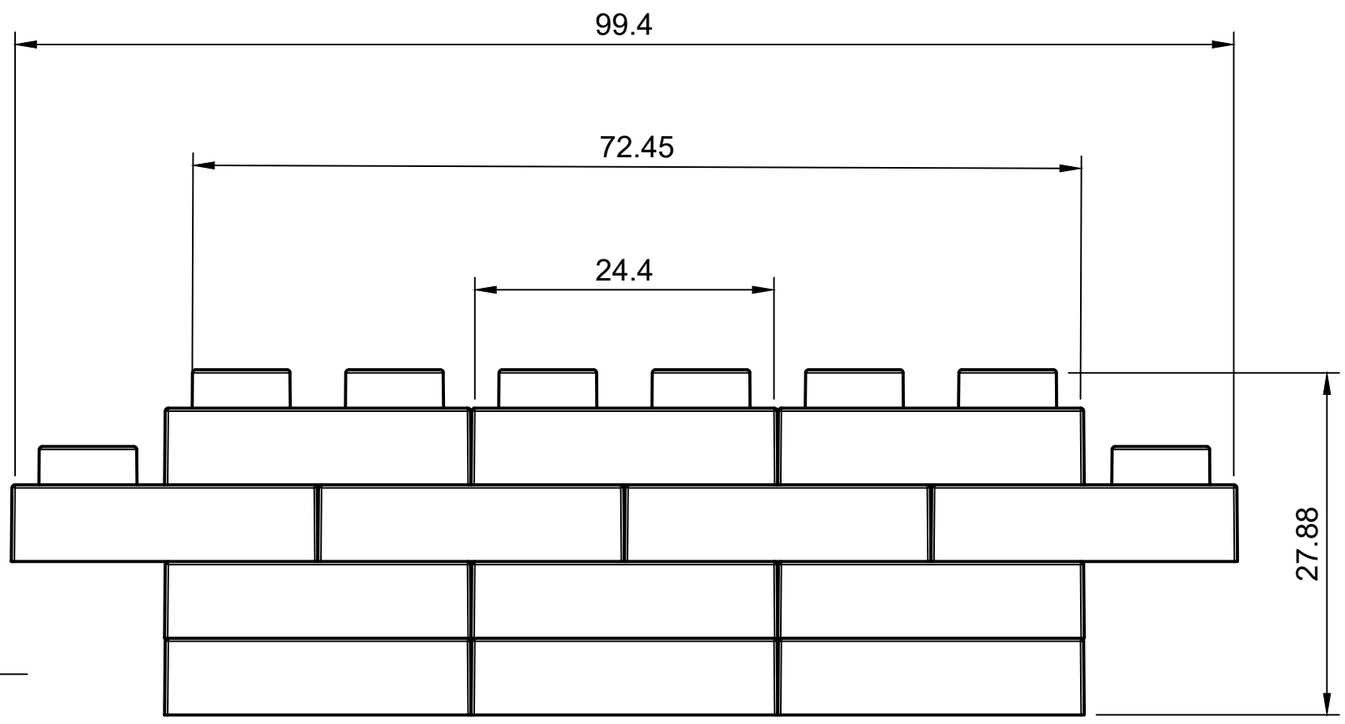


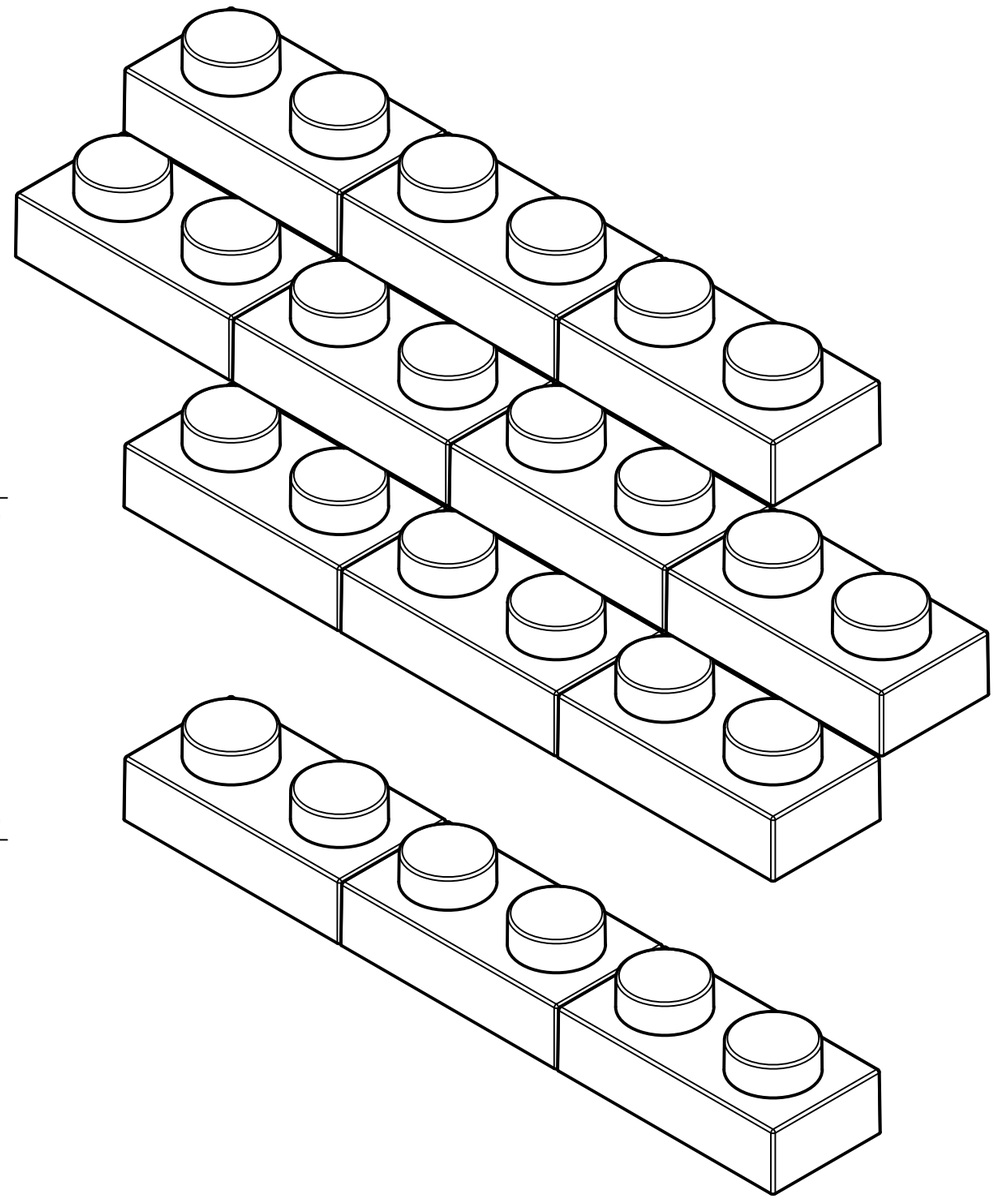
Contact Pressure

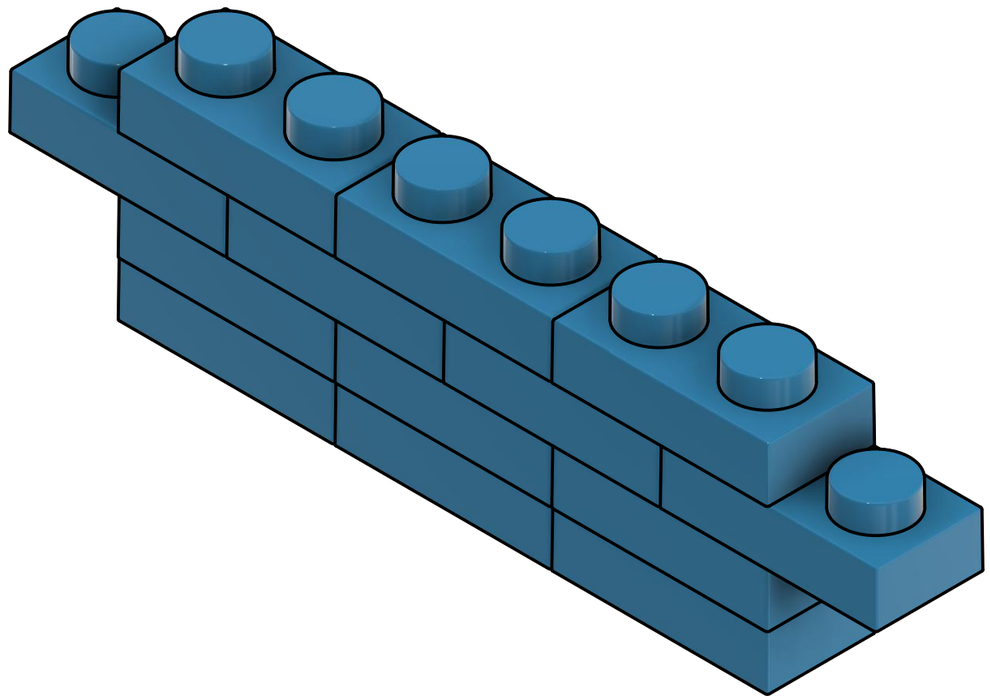
Total

[MPa] 0 1.023

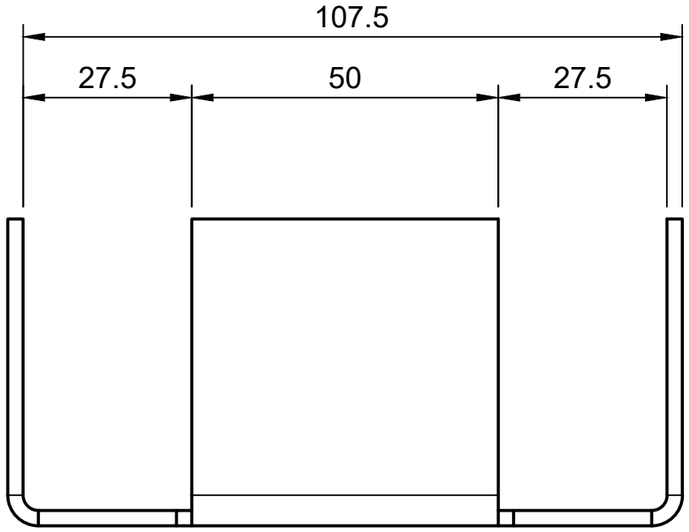


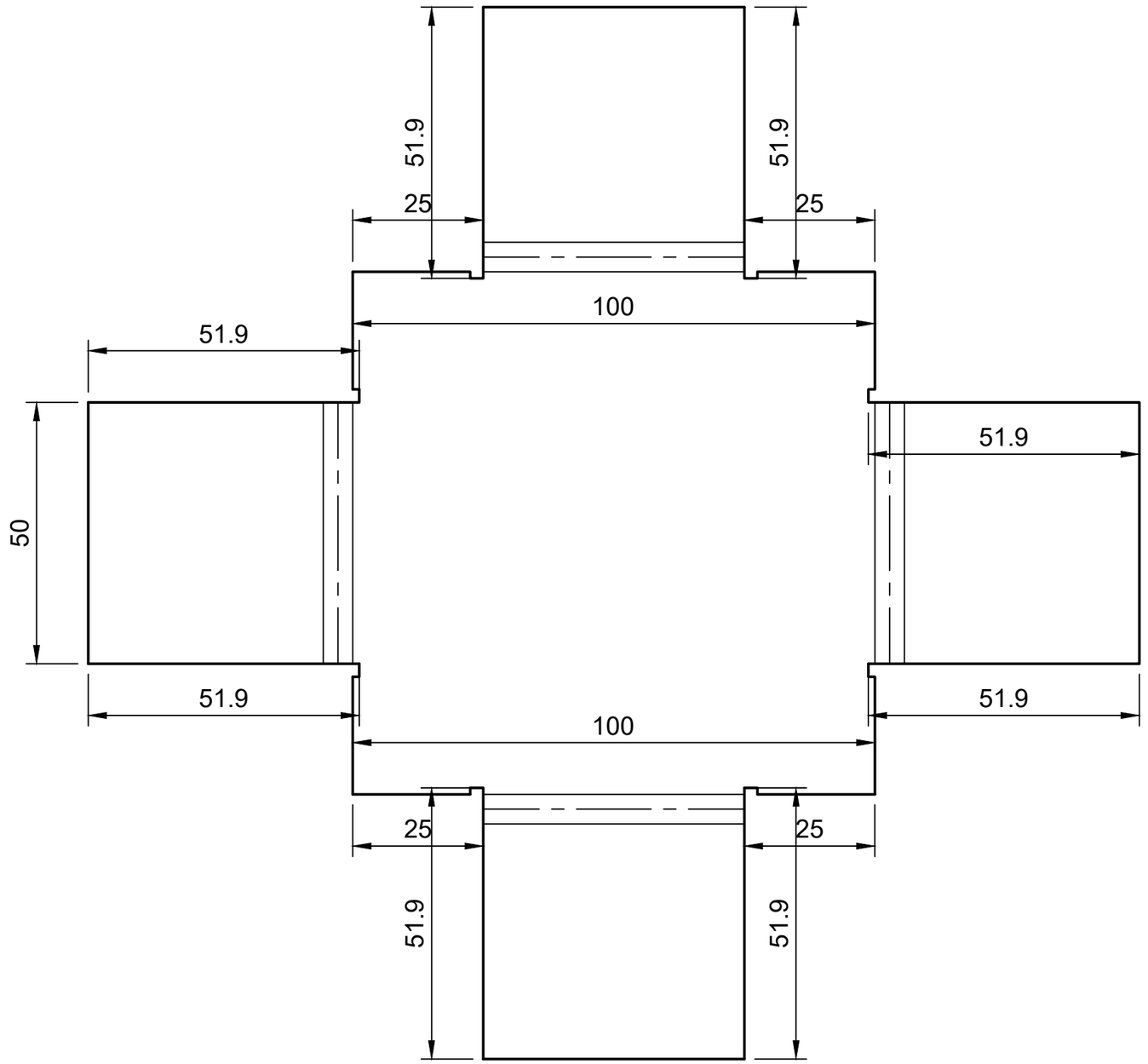


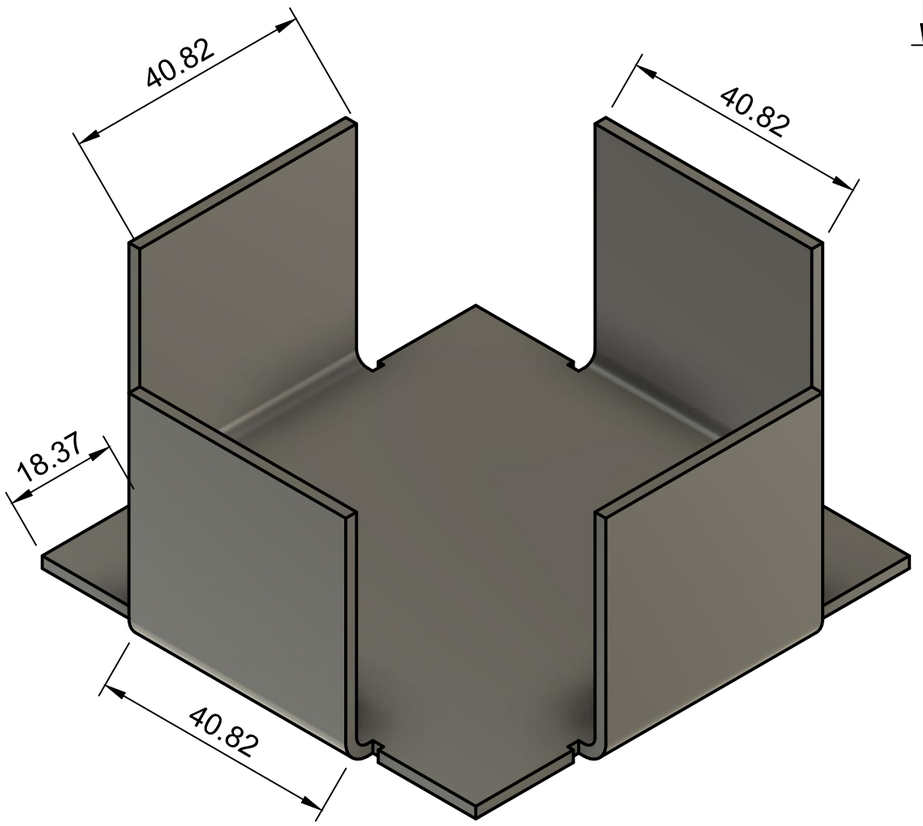




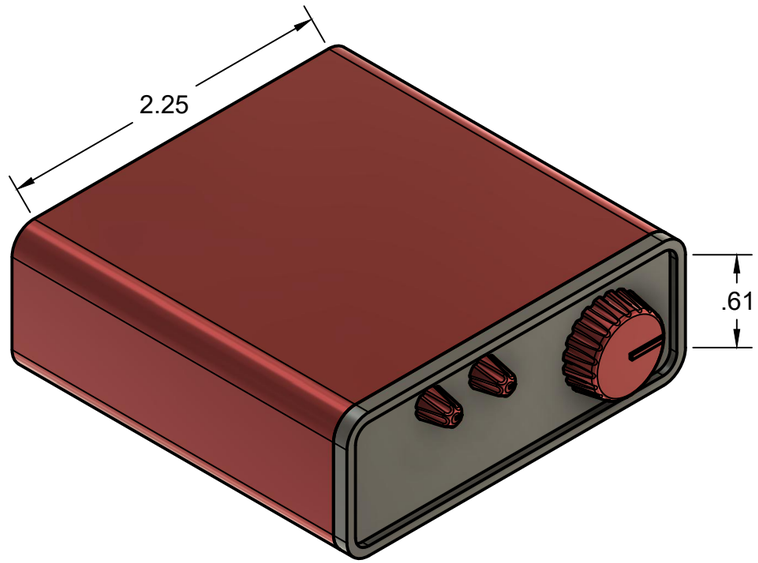
Dept.	Technical reference	Created by	Approved by
		Sterling Richard 9/14/2020	
Unit of Measurement: Millimeters		Document type	Document status
		Title: Building Block Assembly - Drawing	DWG No.
			Rev. / Date of issue / Sheet: 1/1

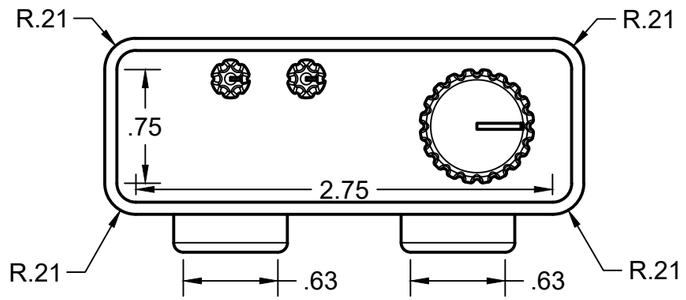


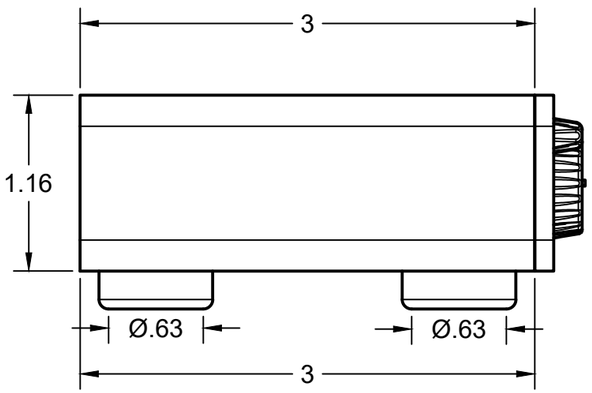


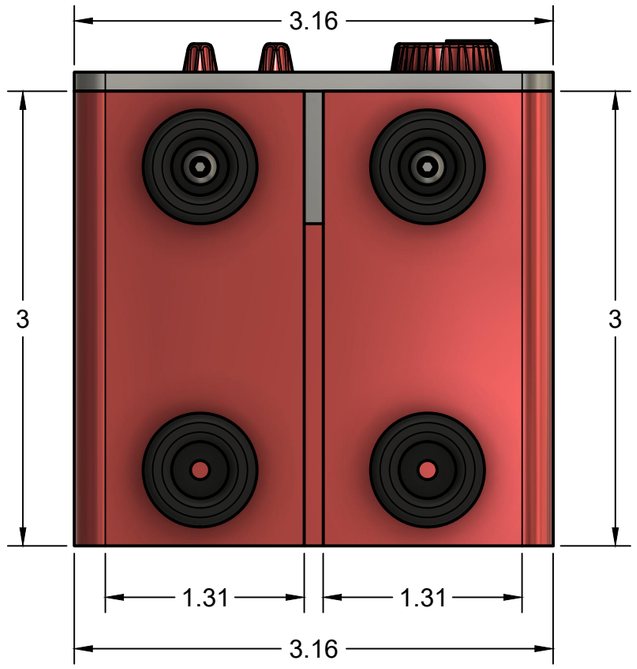


Dept.	Technical reference	Created by Sterling Richard 9/26/2020	Approved by
Unit of Measurement: Millimeters		Document type	Document status
		Title Sheet Metal-PE	DWG No.
		Rev. / Date of issue	Sheet 1/1









Unit of Measurement: Inches	PROJECT: Practice Assignments				
	TITLE: Control Box - PE v3				
APPROVED	SIZE: B	CODE	DWG NO		REV
CHECKED					
DRAWN Sterling Richard 9/26/2020	SCALE 1:1	WEIGHT	SHEET 1/1		

Software Engineering Projects

Attached are various object oriented programming examples written in C#, C++, and Python to supplement the Duke University Software Engineering certification.

Working With Arithmetic Expression - C#

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;

/*
Write a C# Sharp program that takes the radius of a sphere as input, calculates and displays the surface and volume of the sphere.
*/

namespace SoloLearn
{
        class Program
        {
        static void Main(string[] args)
        {
        double pi = 3.14;
        double radius = int.Parse(Console.ReadLine());
        double surface = 4*pi*radius*radius;
        double volume = (4/3)*pi*radius;
        Console.WriteLine("The sphere's radius is {0}, its surface area is {1}, and its volume is {2}.",radius,surface,volume);
        }
        }
}
```

INPUTS = 40
OUTPUT : The sphere's radius is 40, its surface area is 20,096, and its volume is 125.6.

Fahrenheit to Celsius Converter - C++

```
/* Sterling James Richard
Fahrenheit to Celsius Conversion
C++
*/

#include <iostream>
using namespace std;

int main() {
double f;
cin >> f;
double c = (f - 32.0) * (5.0/9.0);
cout << c;
}
```

INPUTS = 70
OUTPUT: 21.1111

Fibonacci Sequence Calculator - Python

```
/*
Sterling James Richard
Fibonacci Sequence Calculator
The Fibonacci Sequence is an additive sequence such that each number is the sum of two preceding ones.
Source: Live Science - LiveScience.com
*/
first_number = int(input())
second_number = int(input())
count = 0
how_many = int(input())

while count < how_many:
   print(first_number)
            nth = first_number + second_number
            first_number = second_number
            # update values
            second_number = nth
            count +=}
```

INPUTS = [1,2,10]
OUTPUTS = [1, 2, 3, 5, 8, 13, 21, 34, 55, 89]