BONUS TASK 1,2,3

1-FINITE ELEMENT ANALYSIS

2-CODE

3-PROTOTYPE

Submitted to: DR **Aya alHabbak**

Eng **Ahmed Hamed**

Ahmed Mostafa Elsayed 121026

Khaled Ayman Abdelazeem 1210099

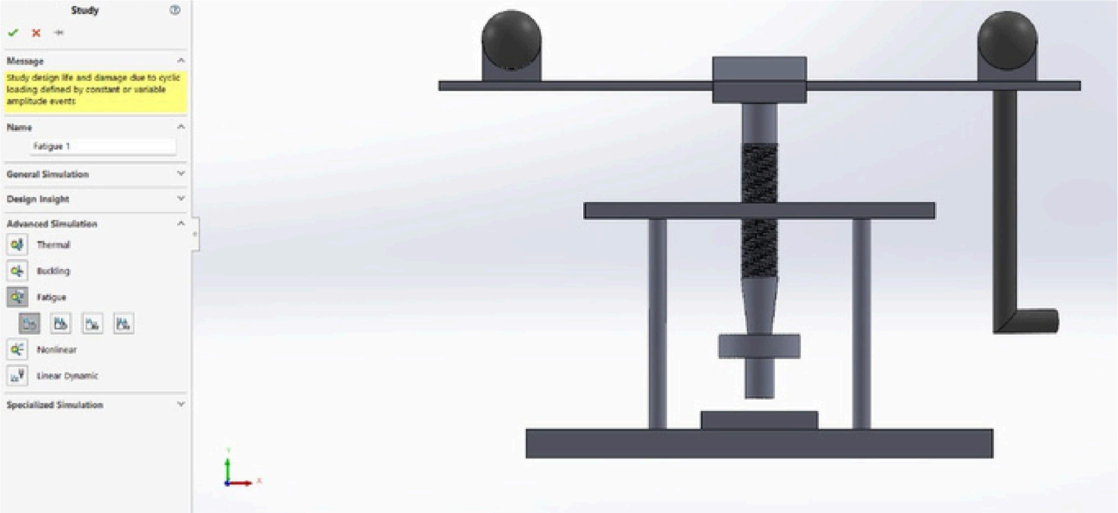
Omar sobhy ellethy 1210066

Waleed Ashraf 4220147

About **FEA**

Finite Element Analysis (**FEA**) is a computational method used to simulate physical phenomena by dividing complex structures into smaller, simpler elements. It solves partial differential equations to predict how these elements will behave under various conditions like stress, heat transfer, and fluid flow. FEA helps engineers optimize designs, predict failure points, and improve performance in diverse fields like aerospace, automotive, and civil engineering. By discretizing continuous systems into finite elements, it allows for accurate modeling of real-world behaviors and complex geometries. FEA software packages employ numerical methods to iteratively solve equations, providing insights into structural integrity, thermal behavior, and fluid dynamics. It's a powerful tool for virtual prototyping, reducing development costs and time-to-market while enhancing product reliability and efficiency.

Step 1: Fatigue type (contant amplitude even with defined cyles)

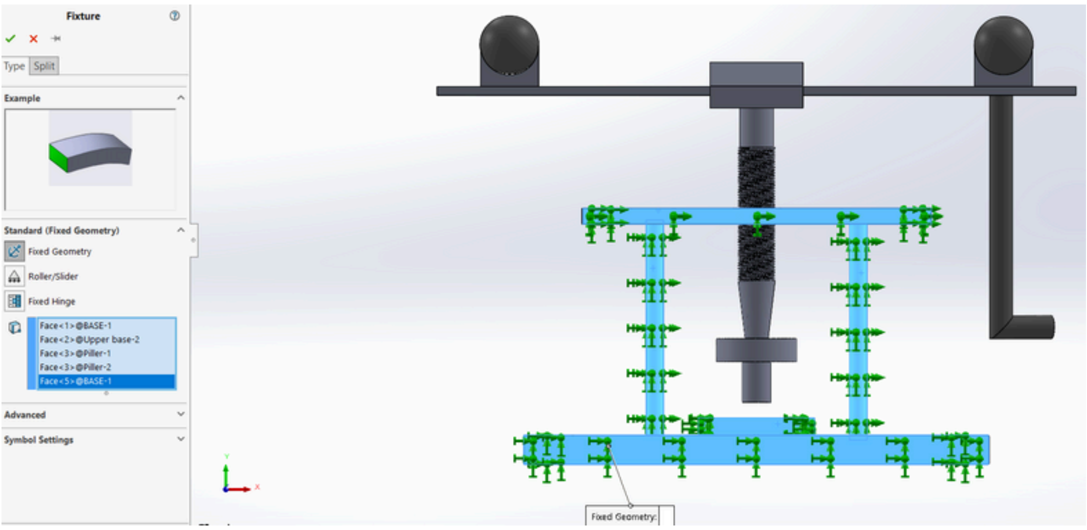


Message

Study design life and damage due to cyclic loading defined by constant or variable amplitude events

Name: Fatigue 1

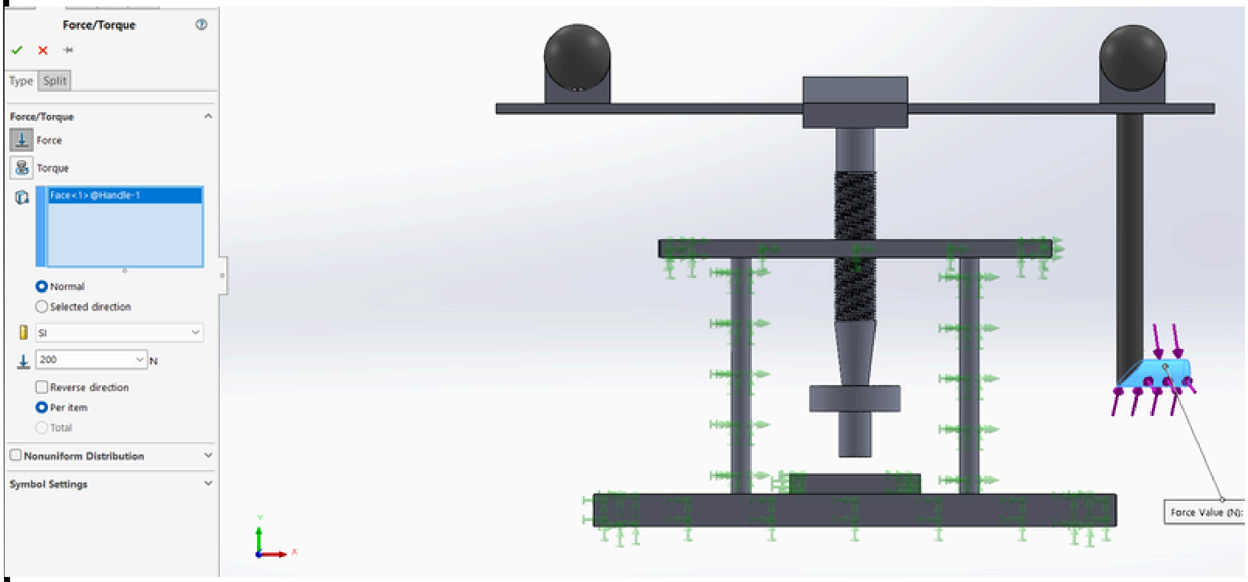
Step 2: choose fixed geometries and load the forces



Face<1>@BASE-1
Face<2>@Upper base-2
Face<3>@Piller-1
Face<3>@Piller-2
Face<5>@BASE-1

Step 3: setting the external (force applied the worker) that we exactly calculated on the main report

200N ON THE HANDLE



Force/Torque

Type | Split

Force/Torque

Force

Torque

Face<1>@Handle-1

● Normal

○ Selected direction

SI

200 N

□ Reverse direction

● Per item

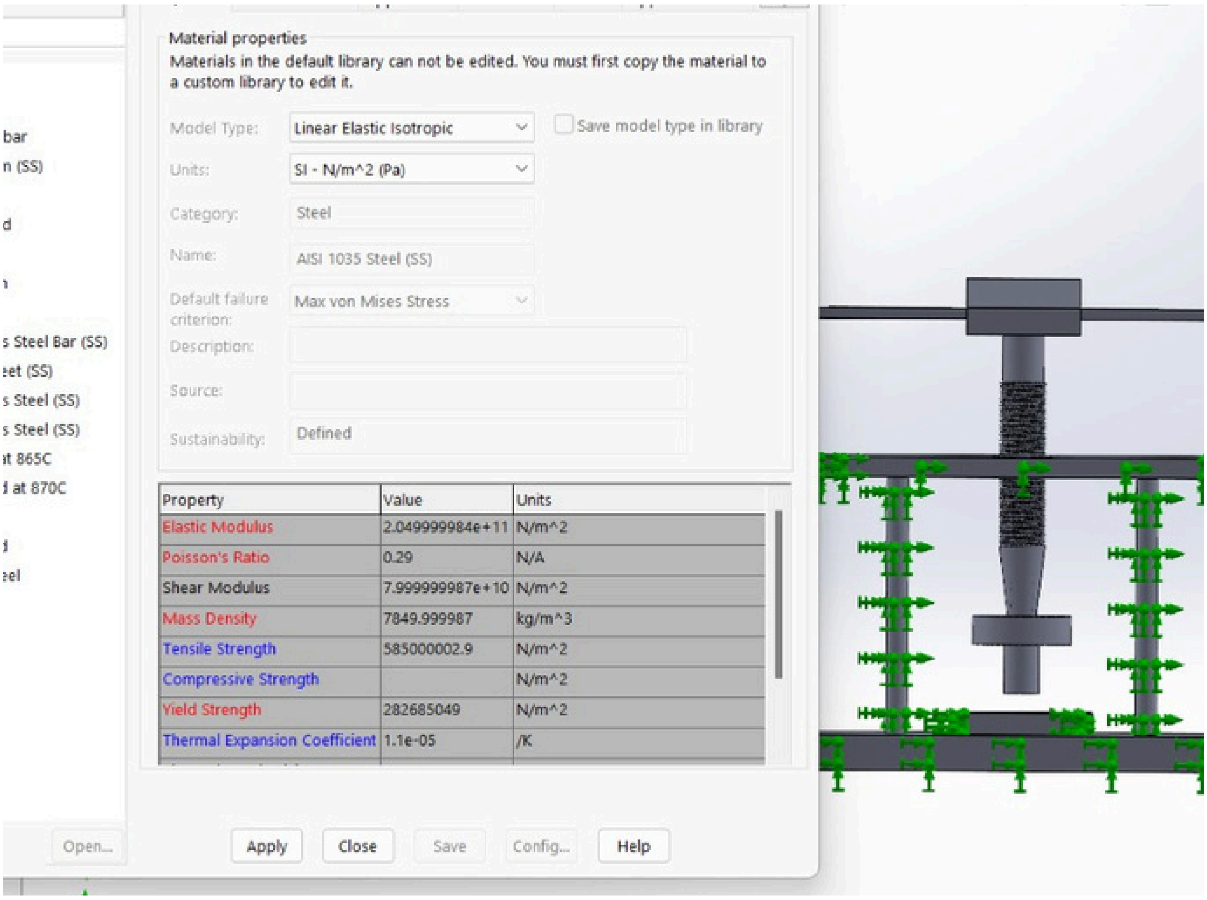
○ Total

□ Nonuniform Distribution

Symbol Settings

Step 4:

choosing the selected material (we have select them on the report)



Material properties

Materials in the default library can not be edited. You must first copy the material to a custom library to edit it.

Model Type: Linear Elastic Isotropic

□ Save model type in library

Units: SI - N/m^2 (Pa)

Category: Steel

Name: AISI 1035 Steel (SS)

Default failure criterion: Max von Mises Stress

Description:

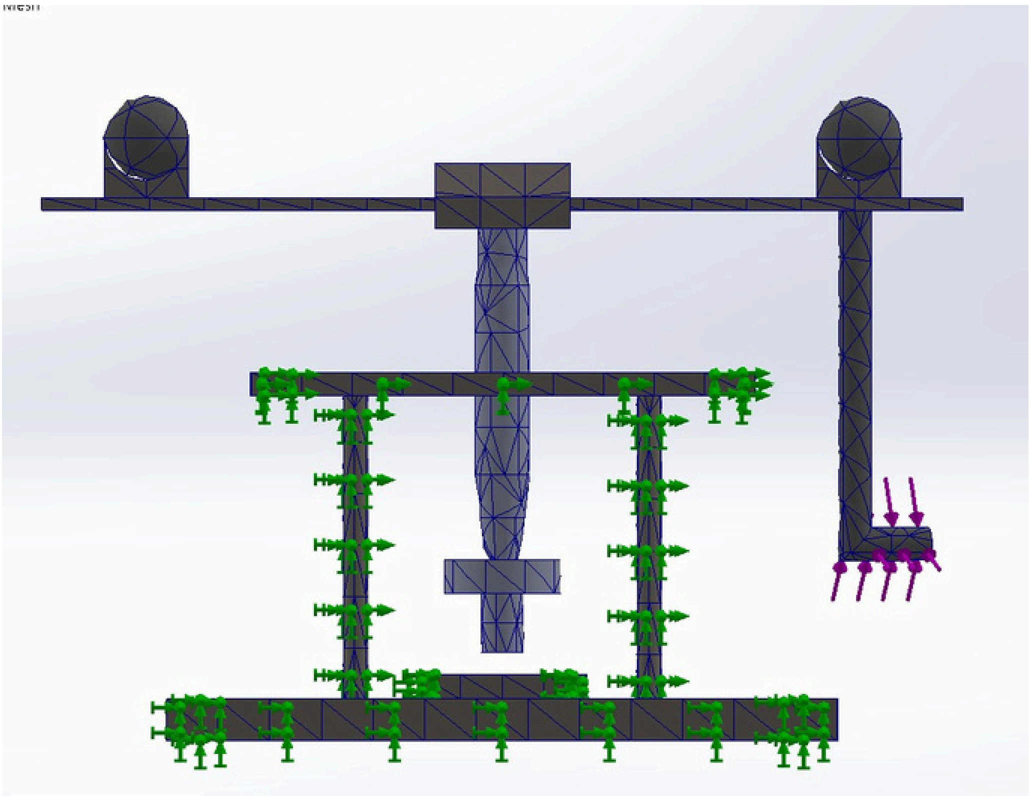
Source:

Sustainability: Defined

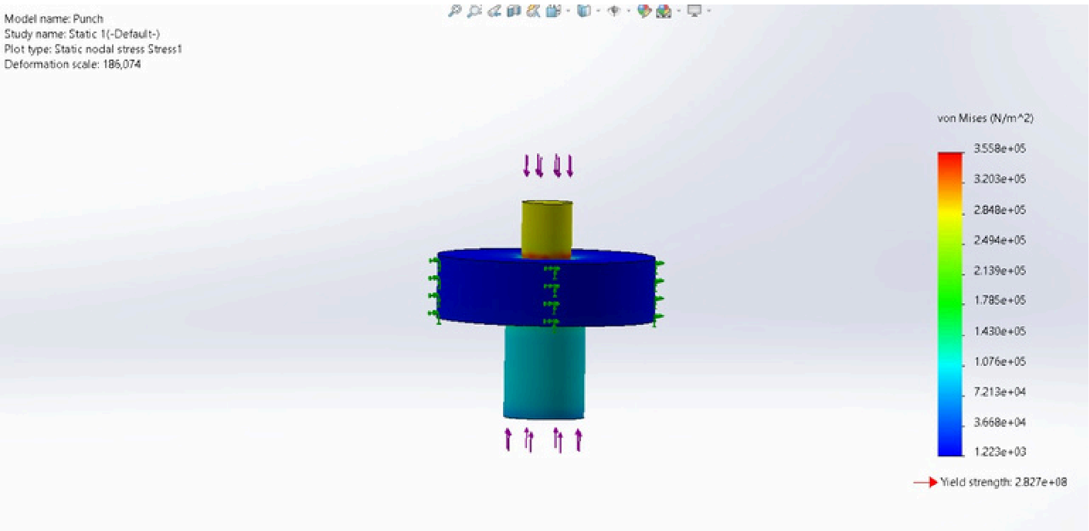
Property	Value	Units
Elastic Modulus	2.049999984e+11	N/m^2
Poisson's Ratio	0.29	N/A
Shear Modulus	7.999999987e+10	N/m^2
Mass Density	7849.999987	kg/m^3
Tensile Strength	585000002.9	N/m^2
Compressive Strength		N/m^2
Yield Strength	282685049	N/m^2
Thermal Expansion Coefficient	1.1e-05	/K

Step 5: MESHING

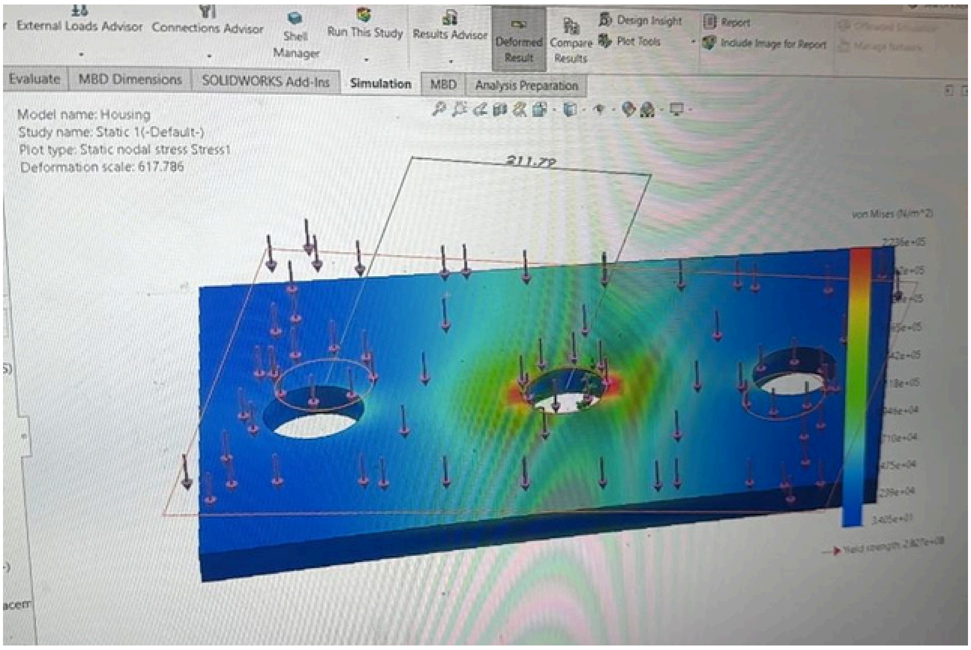
Meshing in FEA is the process of dividing complex structures into smaller elements to numerically solve equations. It determines the accuracy and efficiency of simulations by balancing element size and computational resources, ensuring precise analysis of stress, heat transfer, and fluid flow within the model. Mesh quality directly impacts simulation results, influencing the reliability of engineering predictions and optimizations.



Step 5: results

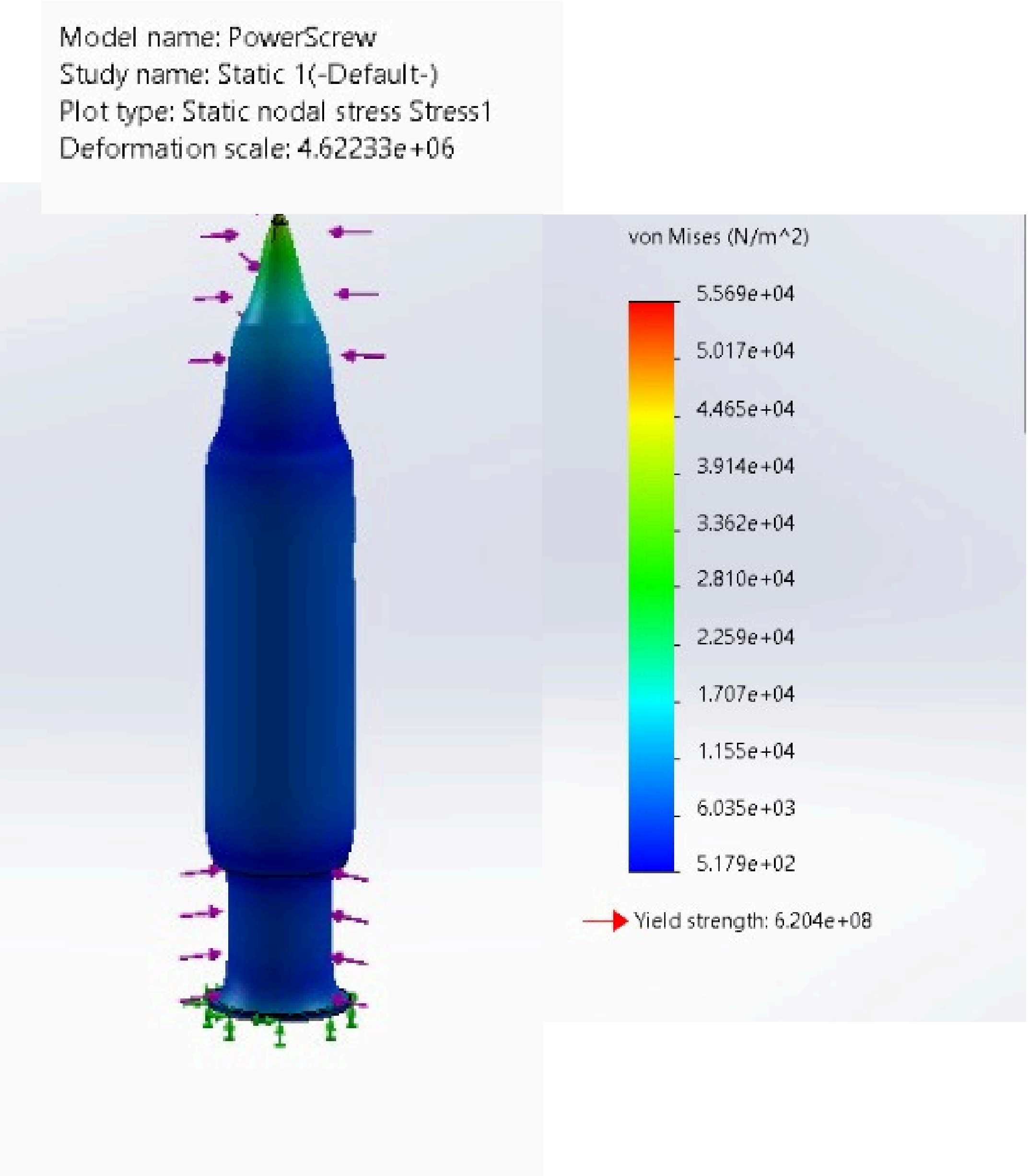


Model name: Punch
Study name: Static 1(-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 186,074



Model name: Housing
Study name: Static 1(-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 617.786

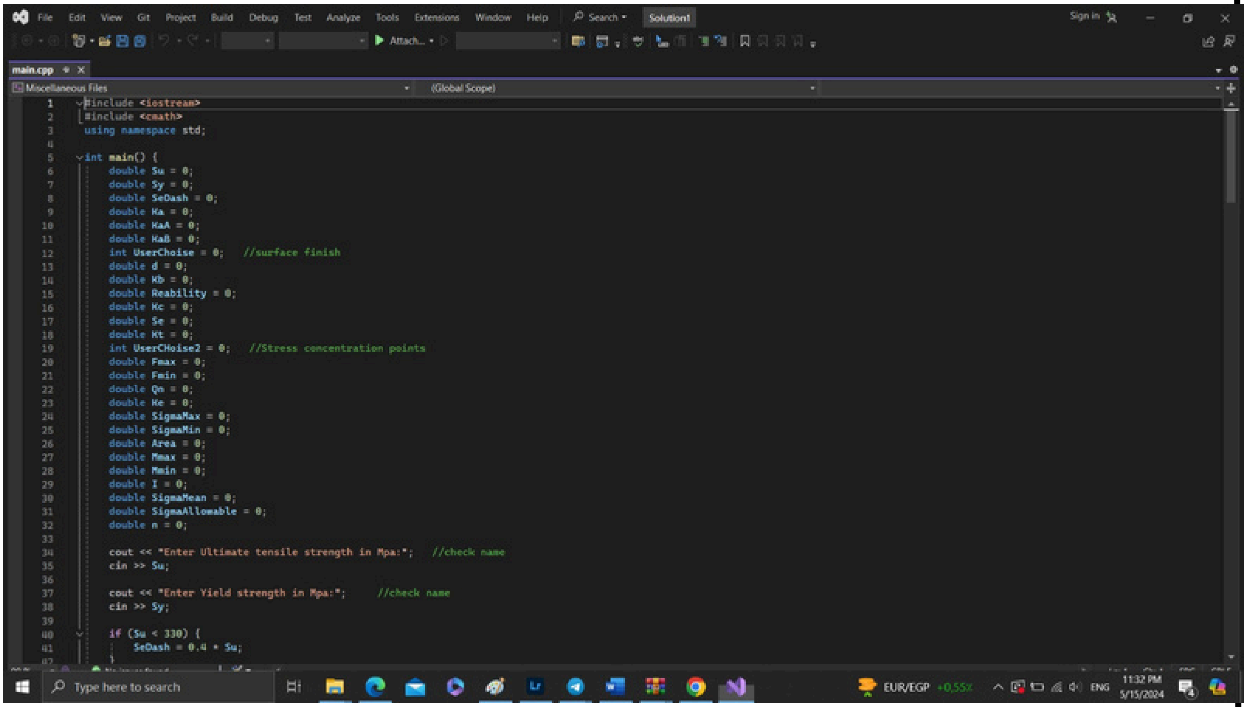
Step 5: results



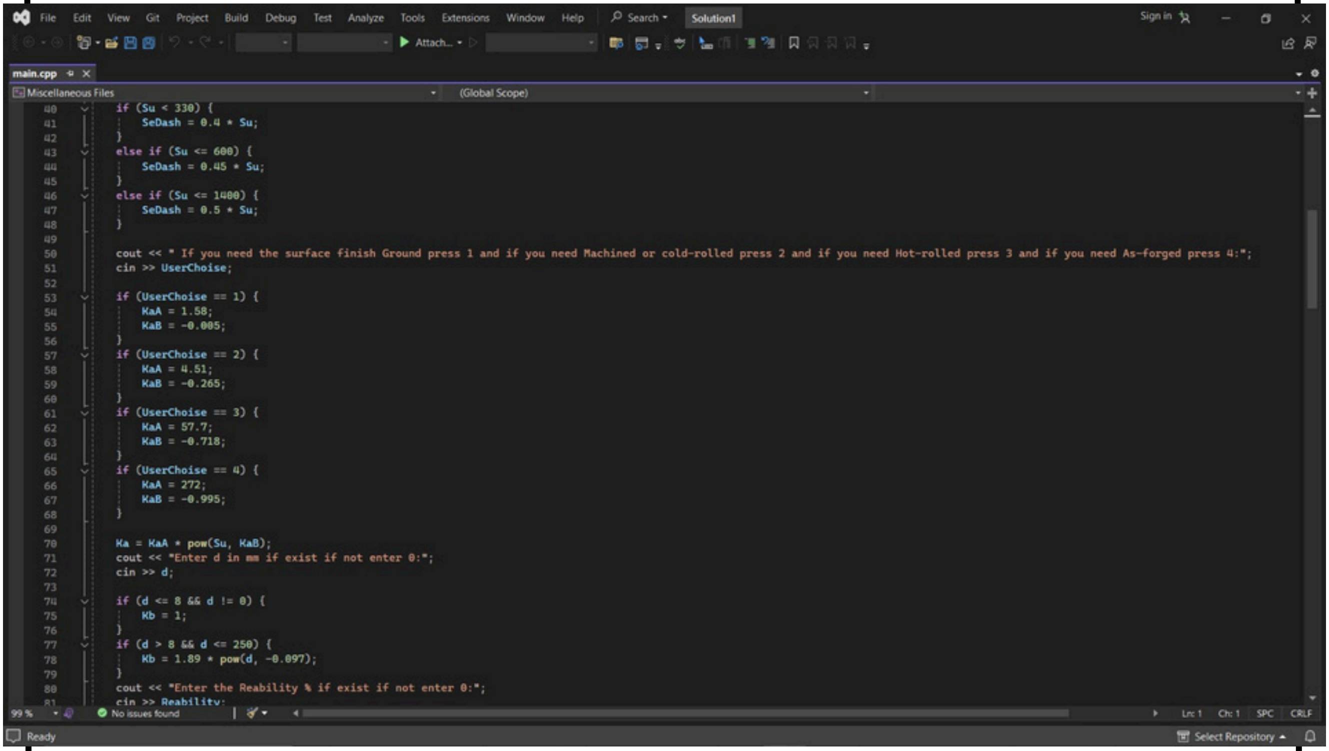
Model name: PowerScrew
Study name: Static 1(-Default-)
Plot type: Static nodal stress Stress1
Deformation scale: 4.62233e+06

Yield strength: 6.204e+08

CODE USING C++ TO CALCULATE FATIGUE

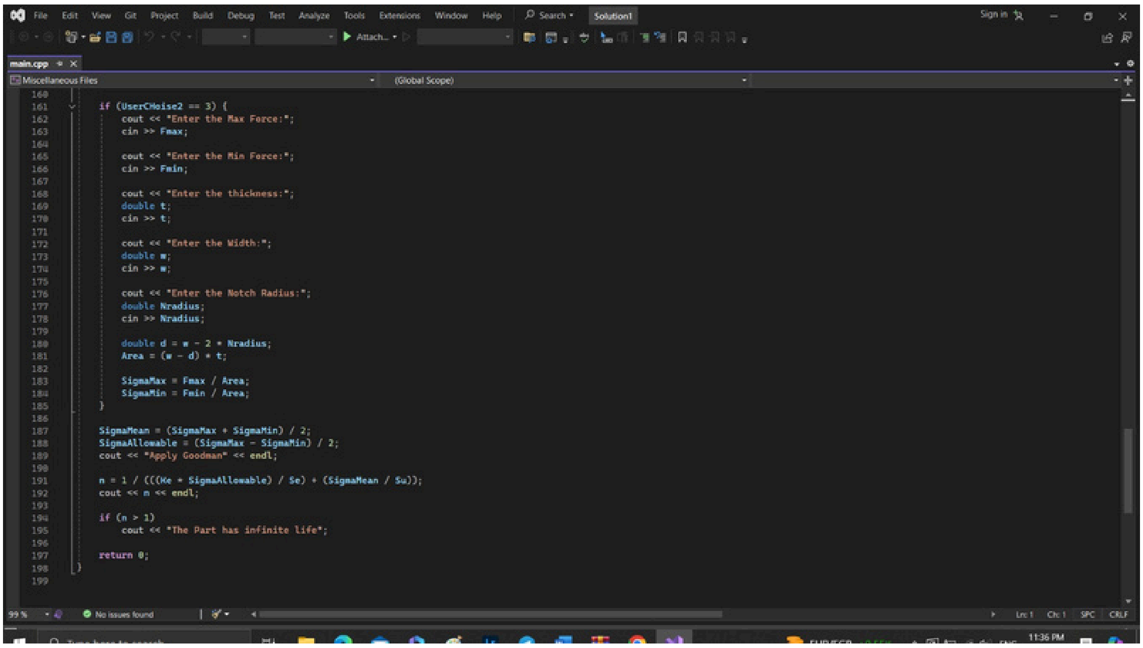


```cpp
#include <iostream>
#include <cmath>
using namespace std;

int main() {
    double Su = 0;
    double Sy = 0;
    double SeDash = 0;
    double Ka = 0;
    double KaA = 0;
    double KaB = 0;
    int UserChoise = 0;   //surface finish
    double d = 0;
    double Kb = 0;
    double Reability = 0;
    double Kc = 0;
    double Se = 0;
    double Kt = 0;
    int UserCHoise2 = 0;   //Stress concentration points
    double Fmax = 0;
    double Fmin = 0;
    double Qn = 0;
    double Ke = 0;
    double SigmaMax = 0;
    double SigmaMin = 0;
    double Area = 0;
    double Mmax = 0;
    double Mmin = 0;
    double I = 0;
    double SigmaMean = 0;
    double SigmaAllowable = 0;
    double n = 0;

    cout << "Enter Ultimate tensile strength in Mpa:";   //check name
    cin >> Su;

    cout << "Enter Yield strength in Mpa:";     //check name
    cin >> Sy;

    if (Su < 330) {
        SeDash = 0.4 * Su;
    }
```

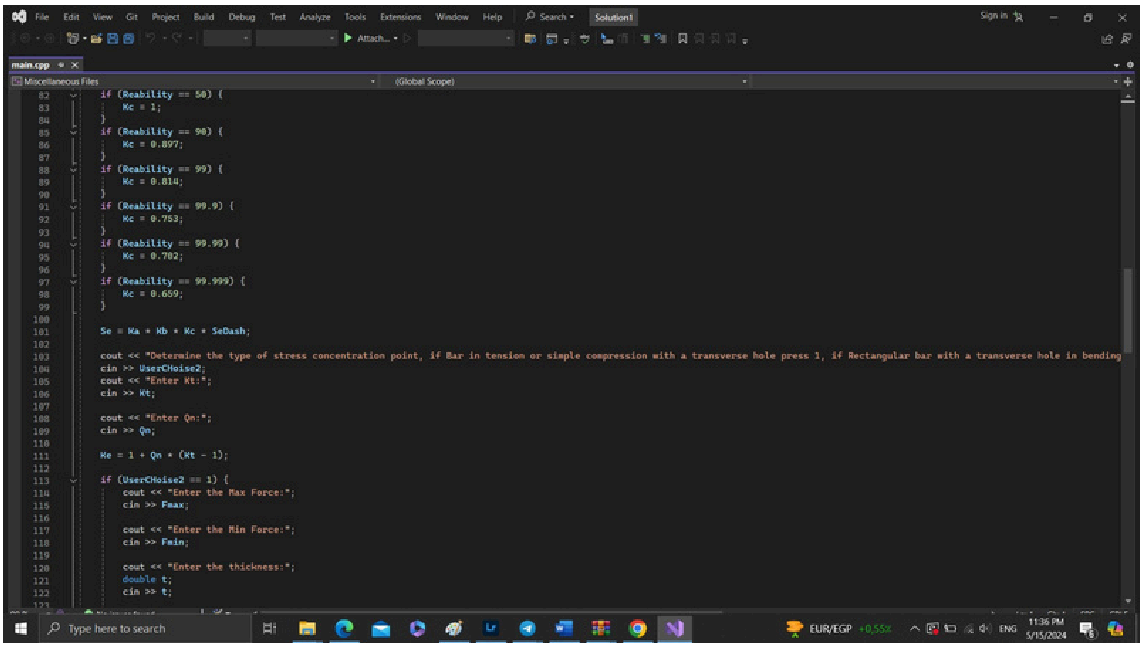


```cpp
    if (Su < 330) {
        SeDash = 0.4 * Su;
    }
    else if (Su <= 600) {
        SeDash = 0.45 * Su;
    }
    else if (Su <= 1400) {
        SeDash = 0.5 * Su;
    }

    cout << " If you need the surface finish Ground press 1 and if you need Machined or cold-rolled press 2 and if you need Hot-rolled press 3 and if you need As-forged press 4:";
    cin >> UserChoise;

    if (UserChoise == 1) {
        KaA = 1.58;
        KaB = -0.085;
    }
    if (UserChoise == 2) {
        KaA = 4.51;
        KaB = -0.265;
    }
    if (UserChoise == 3) {
        KaA = 57.7;
        KaB = -0.718;
    }
    if (UserChoise == 4) {
        KaA = 272;
        KaB = -0.995;
    }

    Ka = KaA * pow(Su, KaB);
    cout << "Enter d in mm if exist if not enter 0:";
    cin >> d;

    if (d <= 8 && d != 0) {
        Kb = 1;
    }
    if (d > 8 && d <= 250) {
        Kb = 1.89 * pow(d, -0.097);
    }
    cout << "Enter the Reability % if exist if not enter 0:";
    cin >> Reability;
```

CODE USING C++ TO CALCULATE FATIGUE

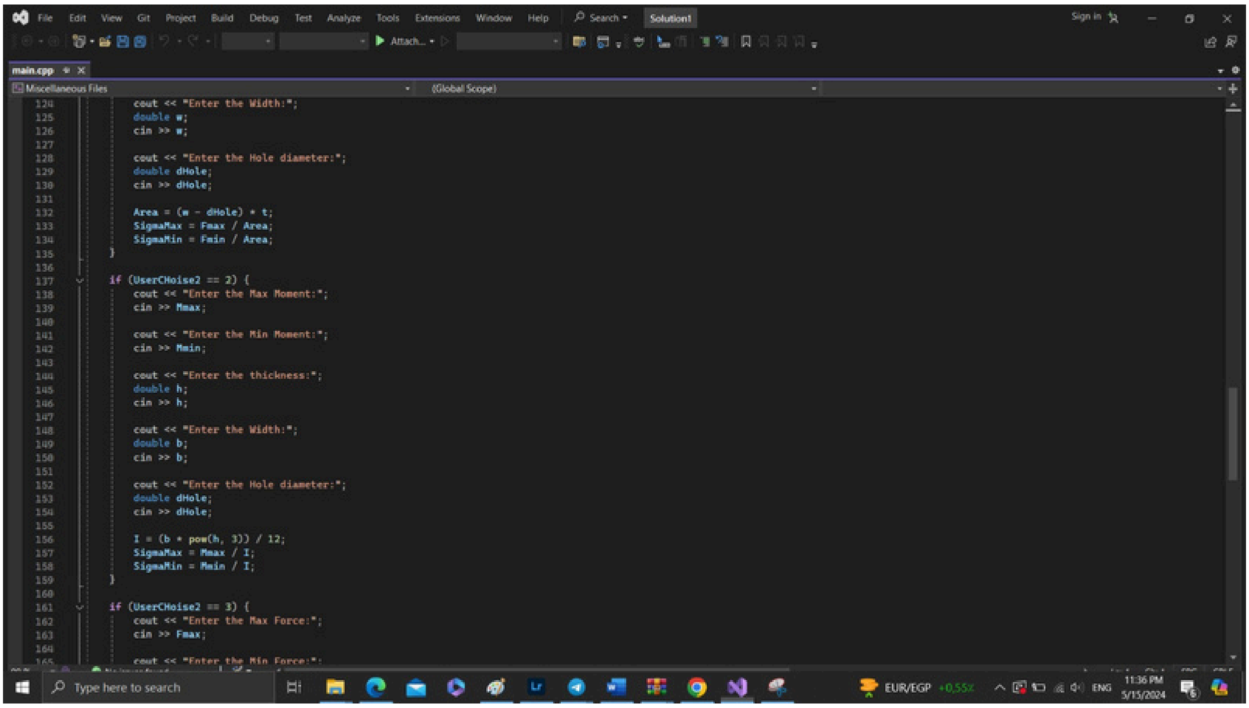


```cpp

    if (UserCHoise2 == 3) {
        cout << "Enter the Max Force:";
        cin >> Fmax;

        cout << "Enter the Min Force:";
        cin >> Fmin;

        cout << "Enter the thickness:";
        double t;
        cin >> t;

        cout << "Enter the Width:";
        double w;
        cin >> w;

        cout << "Enter the Notch Radius:";
        double Nradius;
        cin >> Nradius;

        double d = w - 2 * Nradius;
        Area = (w - d) * t;

        SigmaMax = Fmax / Area;
        SigmaMin = Fmin / Area;
    }

    SigmaMean = (SigmaMax + SigmaMin) / 2;
    SigmaAllowable = (SigmaMax - SigmaMin) / 2;
    cout << "Apply Goodman" << endl;

    n = 1 / (((Ke * SigmaAllowable) / Se) + (SigmaMean / Su));
    cout << n << endl;

    if (n > 1)
        cout << "The Part has infinite life";

    return 0;
}

```



```cpp
    if (Reability == 50) {
        Kc = 1;
    }
    if (Reability == 90) {
        Kc = 0.897;
    }
    if (Reability == 99) {
        Kc = 0.814;
    }
    if (Reability == 99.9) {
        Kc = 0.753;
    }
    if (Reability == 99.99) {
        Kc = 0.702;
    }
    if (Reability == 99.999) {
        Kc = 0.659;
    }

    Se = Ka * Kb * Kc * SeDash;

    cout << "Determine the type of stress concentration point, if Bar in tension or simple compression with a transverse hole press 1, if Rectangular bar with a transverse hole in bending
    cin >> UserCHoise2;
    cout << "Enter Kt:";
    cin >> Kt;

    cout << "Enter Qn:";
    cin >> Qn;

    Ke = 1 + Qn * (Kt - 1);

    if (UserCHoise2 == 1) {
        cout << "Enter the Max Force:";
        cin >> Fmax;

        cout << "Enter the Min Force:";
        cin >> Fmin;

        cout << "Enter the thickness:";
        double t;
        cin >> t;
```

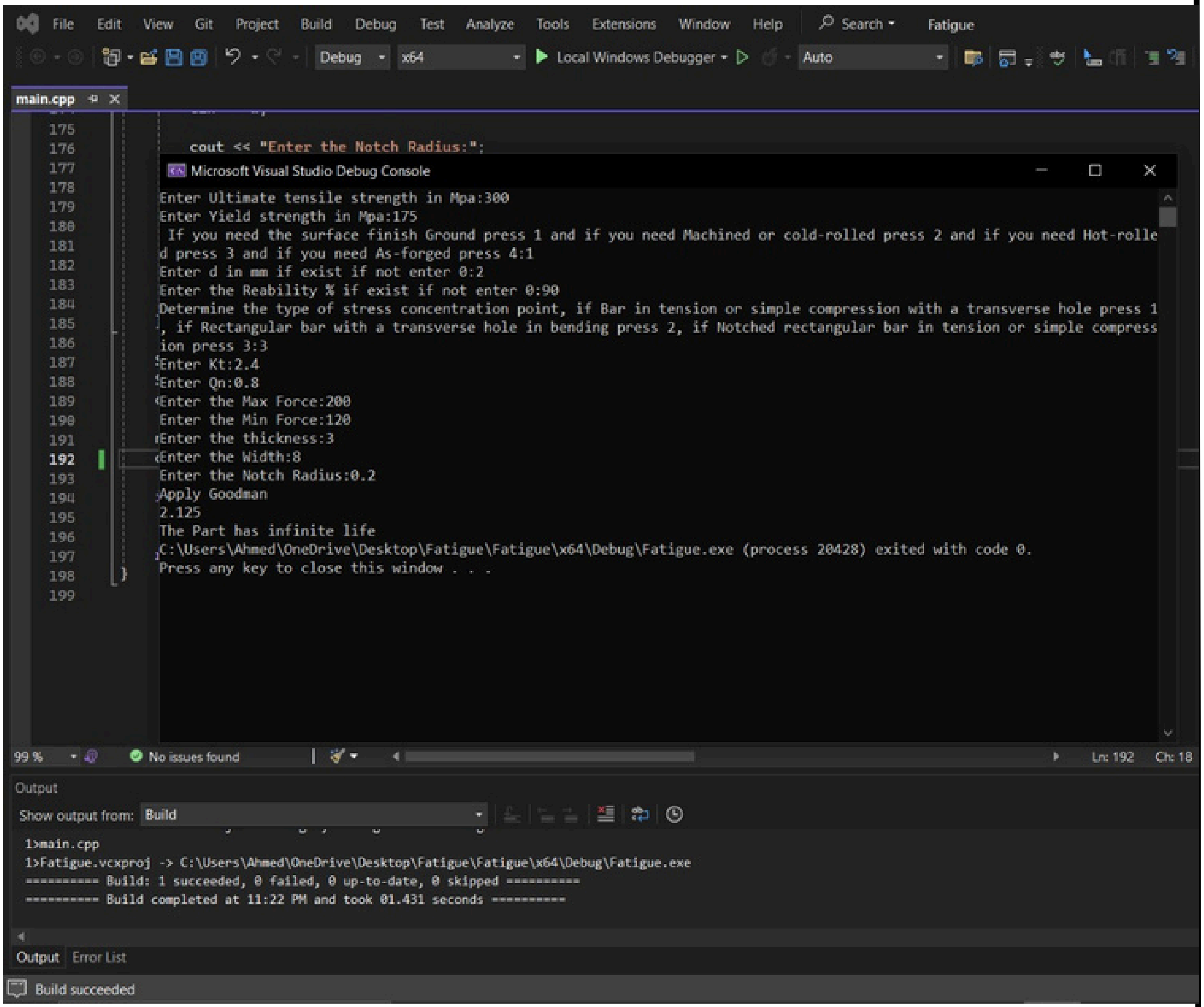
CODE USING C++ TO CALCULATE FATIGUE



```cpp
        cout << "Enter the Width:";
        double w;
        cin >> w;

        cout << "Enter the Hole diameter:";
        double dHole;
        cin >> dHole;

        Area = (w - dHole) * t;
        SigmaMax = Fmax / Area;
        SigmaMin = Fmin / Area;
    }

    if (UserCHoise2 == 2) {
        cout << "Enter the Max Moment:";
        cin >> Mmax;

        cout << "Enter the Min Moment:";
        cin >> Mmin;

        cout << "Enter the thickness:";
        double h;
        cin >> h;

        cout << "Enter the Width:";
        double b;
        cin >> b;

        cout << "Enter the Hole diameter:";
        double dHole;
        cin >> dHole;

        I = (b * pow(h, 3)) / 12;
        SigmaMax = Mmax / I;
        SigmaMin = Mmin / I;
    }

    if (UserCHoise2 == 3) {
        cout << "Enter the Max Force:";
        cin >> Fmax;

        cout << "Enter the Min Force:";
```

CODE USING C++ TO CALCULATE FATIGUE

HERE IS AN EXAMPLE OF THE OUTPUT



```
    cout << "Enter the Notch Radius:";
```

```
Enter Ultimate tensile strength in Mpa:300
Enter Yield strength in Mpa:175
 If you need the surface finish Ground press 1 and if you need Machined or cold-rolled press 2 and if you need Hot-rolle
d press 3 and if you need As-forged press 4:1
Enter d in mm if exist if not enter 0:2
Enter the Reability % if exist if not enter 0:90
Determine the type of stress concentration point, if Bar in tension or simple compression with a transverse hole press 1
, if Rectangular bar with a transverse hole in bending press 2, if Notched rectangular bar in tension or simple compress
ion press 3:3
Enter Kt:2.4
Enter Qn:0.8
Enter the Max Force:200
Enter the Min Force:120
Enter the thickness:3
Enter the Width:8
Enter the Notch Radius:0.2
Apply Goodman
2.125
The Part has infinite life
C:\Users\Ahmed\OneDrive\Desktop\Fatigue\Fatigue\x64\Debug\Fatigue.exe (process 20428) exited with code 0.
Press any key to close this window . . .
```

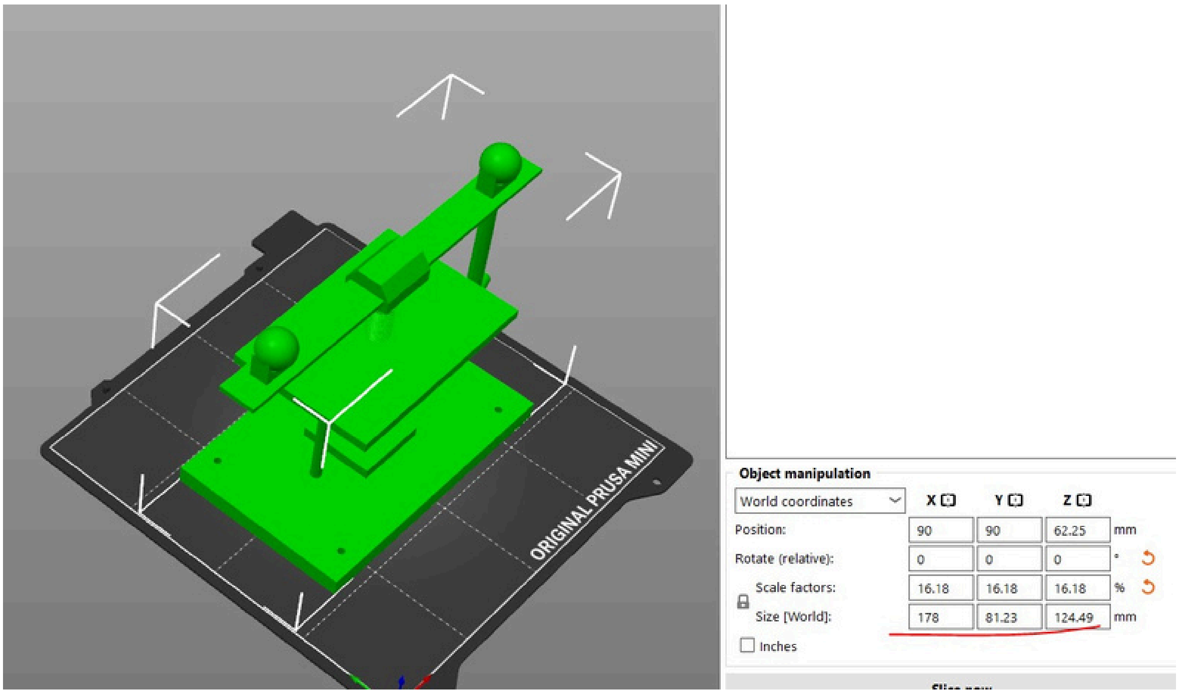
```
1>main.cpp
1>Fatigue.vcxproj -> C:\Users\Ahmed\OneDrive\Desktop\Fatigue\Fatigue\x64\Debug\Fatigue.exe
========== Build: 1 succeeded, 0 failed, 0 up-to-date, 0 skipped ==========
========== Build completed at 11:22 PM and took 01.431 seconds ==========
```

HERE IS THE LINK OF THE DETAILED . .CPP CODING FILE WE DID

DRIVE LINK FOR THE CODE WE DID “CLICK HERE”

Bonus 3:

HERE IS THE 3D PRINTING SETTINGS WE APPLY



Object manipulation

World coordinates	X	Y	Z	
Position:	90	90	62.25	mm
Rotate (relative):	0	0	0	°
Scale factors:	16.18	16.18	16.18	%
Size [World]:	178	81.23	124.49	mm

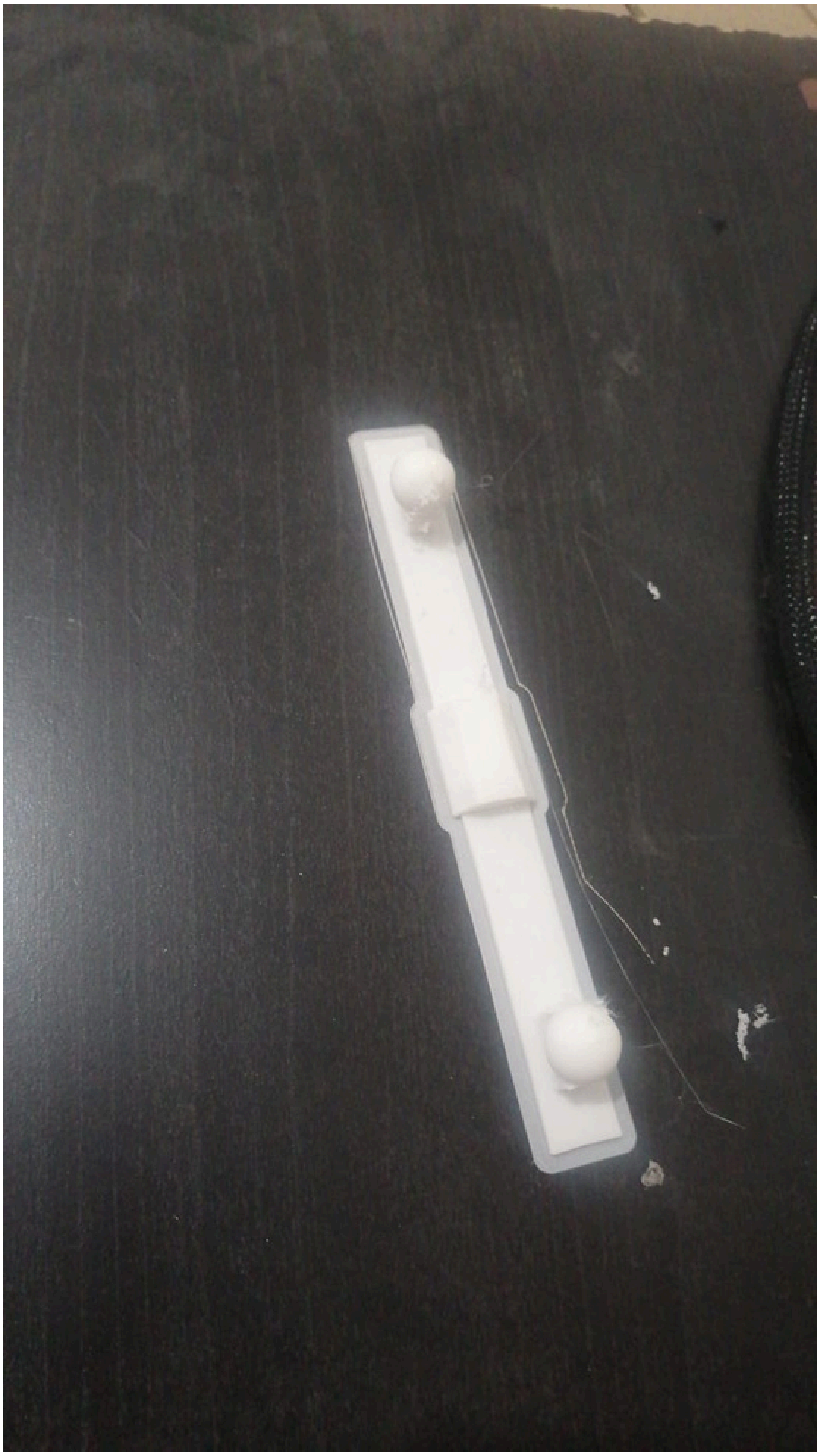
☐ Inches

Bonus 3:

IT IS 11:30PM THE PROTOTYPE IS NOW 3D PRINTING AND HERE IS THE COMPONENTS WE HAVE NOW AND WE WILL SEND IT FINALIZED AFTER TIME

CREALITY
KEEP HANDS AWAY FROM MOVING PARTS
DO NOT TOUCH THE HEATED BED
警告WARNING
警告WAR

Bonus 3:



its dimensions now is
17.8cm*8.1cm*12.5cm