

Task 3b

- Description of implementation of *fkt* function
 - I created 5 lists (*theta_offset*, *d*, *a*, *alpha* and *theta*) for each DH parameter + one for the position of theta in order to assign the values from the dictionaries (mechanism and joints)
 - In a for loop, I multiplied each A matrix ($A_{10} \cdot A_{21} \cdot A_{32} \cdot A_{43} \cdot A_{54} \cdot A_{65}$) and got the Transformation matrix T60
 - From T60 I extracted the rotation matrix and the translation vector (2 arrays)
- Visualization of the range of end-effector positions projected into the x-y plane

-In order to visualize the range of the end-effector positions I took the x and y component from the translation vector and I computed it for different values of theta1:

```
mechanism1 = {"theta1 offset": 0, "theta2 offset": 1.5707963268, "theta3 offset": 0, "theta4 offset": 0, "theta5 offset": -1.5707963268, "theta6 offset": 0, "d1": 0.45, "d2": 0, "d3": 0, "d4": 0.64, "d5": 0, "d6": -0.2, "a1": 0.15, "a2": 0.614, "a3": 0.2, "a4": 0, "a5": 0.03, "a6": 0, "alpha1": 1.5707963268, "alpha2": 0, "alpha3": 1.5707963268, "alpha4": 1.5707963268, "alpha5": -1.5707963268, "alpha6": 0}

# The turning/sweep of the manipulator is in range (-170,170) degrees
n = 10000
a = -2.9670597284
b = 2.9670597284
values = [] # list for the values within (a,b) range
x_values = [] # list for x component of the theta vector
y_values = [] # list for y component of the theta vector
for i in range(0, n):
    values.append(random.uniform(a, b)) # appending values from (a,b) range
    joints1 = {"theta1": values[i], "theta2": 0, "theta3": 0, "theta4": 0, "theta5": 0, "theta6": 0}
    # Creating x and y lists
    x_values.append(fkt(mechanism1, joints1).get('t')[0][0])
    y_values.append(fkt(mechanism1, joints1).get('t')[1][0])

# Creating and plotting the heatmap
heatmap, xedges, yedges = np.histogram2d(x_values, y_values, bins=50)
extent = [xedges[0], xedges[-1], yedges[0], yedges[-1]]

plt.clf()
plt.imshow(heatmap.T, extent=extent, origin='lower')
plt.colorbar()
plt.show()
```

Heatmap representation

