



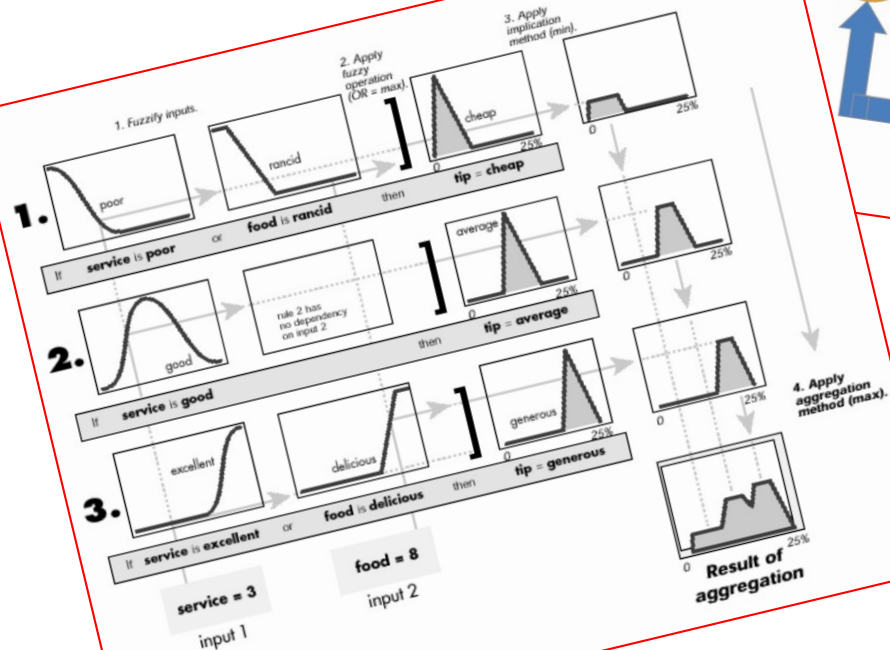
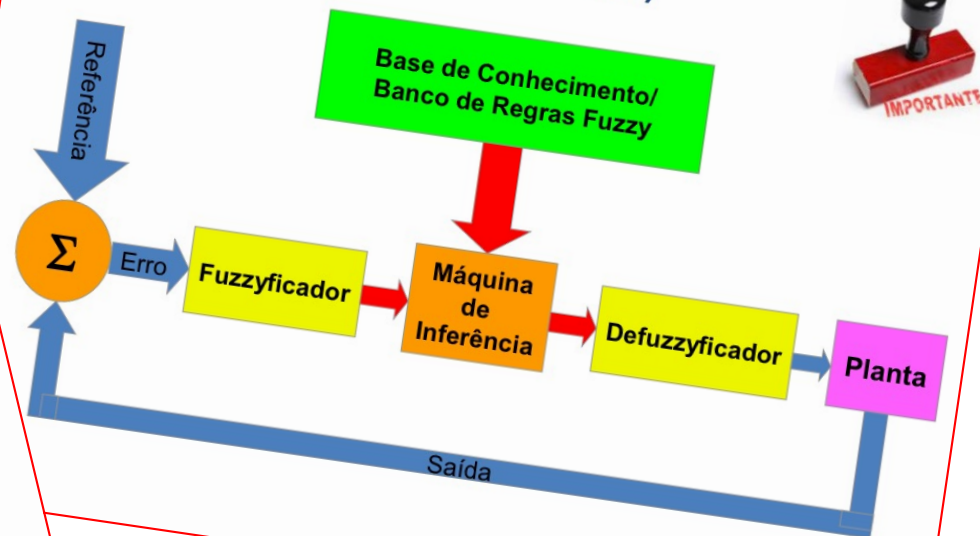
# Aula Laboratório

## Lógica Fuzzy

# Introdução

Inteligência Artificial e Robótica - CC 7711  
Lógica Fuzzy

## Controle Fuzzy (esquema geral)





# Construção de um Sistema Fuzzy

- Identificar o problema.
- Definir as variáveis de entrada e saída.
- Definir os conjuntos fuzzy usados.
- Definir as regras necessárias.
- Selecionar o método de inferência.
- Testar e validar o sistema.

Um exemplo bem didático:

<https://www.youtube.com/watch?v=0Cbv618SoZ0>



# Lógica Fuzzy no Python

```
%pip install scikit-fuzzy
```

```
import numpy as np  
import skfuzzy as fuzz  
from skfuzzy import control as ctrl
```

# Definindo as variáveis

```
#Variaveis de Entrada (Antecedent)
qualidade = ctrl.Antecedent(np.arange(0, 11, 1), 'qualidade')
servico = ctrl.Antecedent(np.arange(0, 11, 1), 'servico')

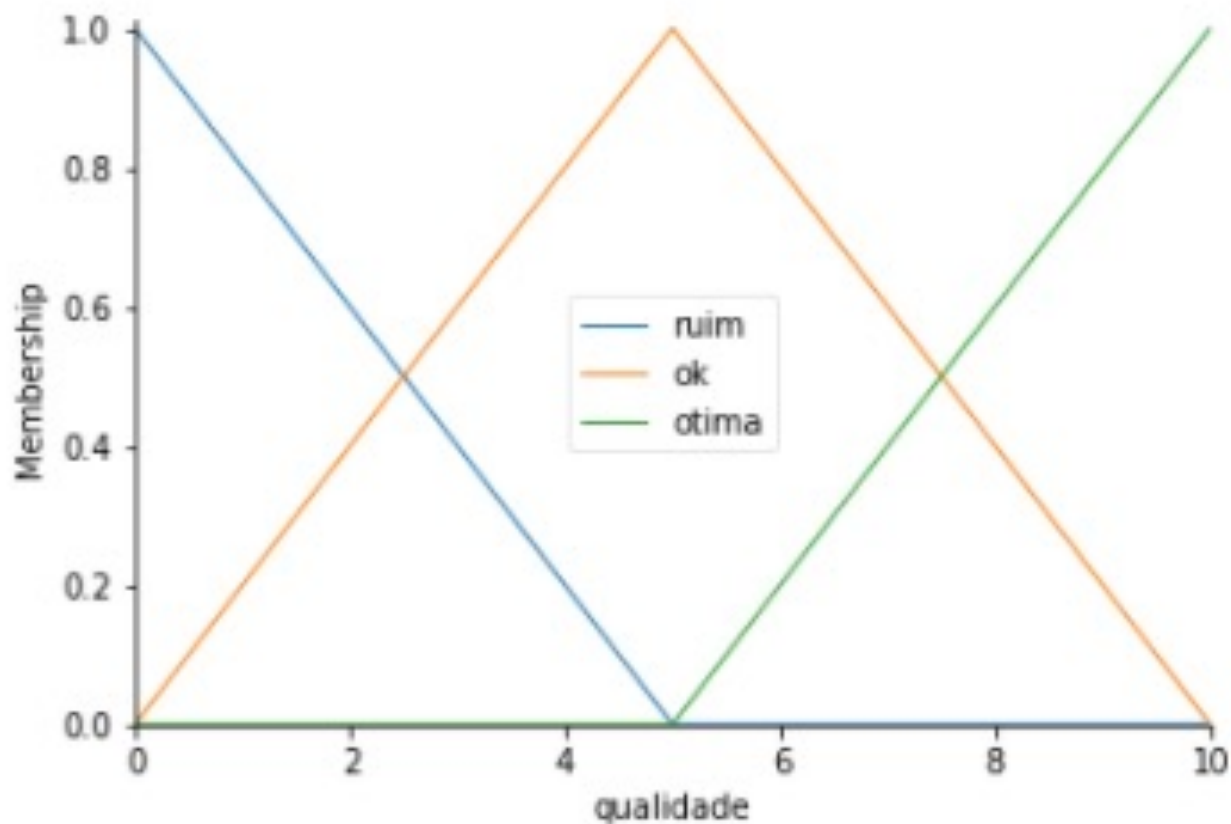
#Variaveis de saída (Consequent)
gorjeta = ctrl.Consequent(np.arange(0, 31, 1), 'tip')
```

`np.arange(0, 11, 1)`

`array([ 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10])`

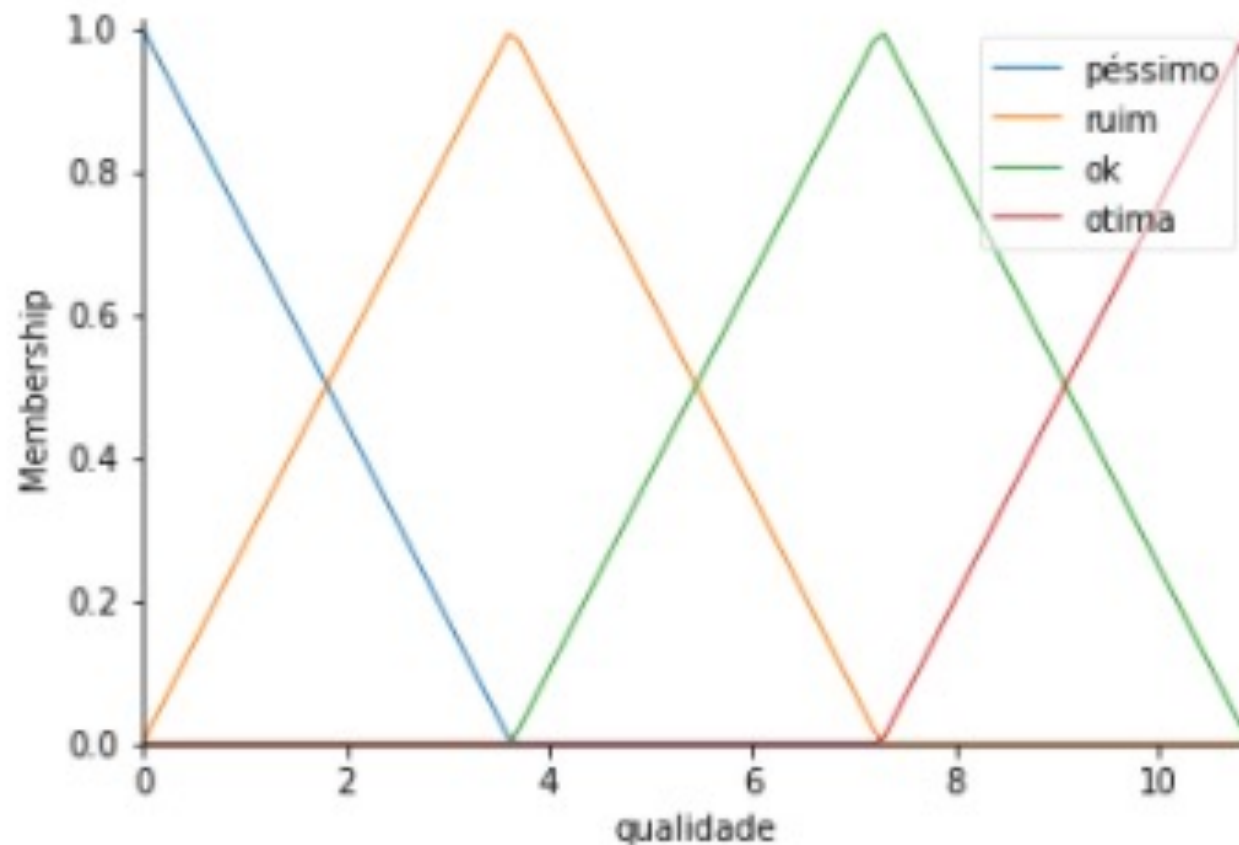
# Conjunto Fuzzy

```
# automf -> Atribuição de categorias automaticamente  
qualidade.automf(names=[ 'ruim', 'ok', 'otima'],)  
qualidade.view()
```



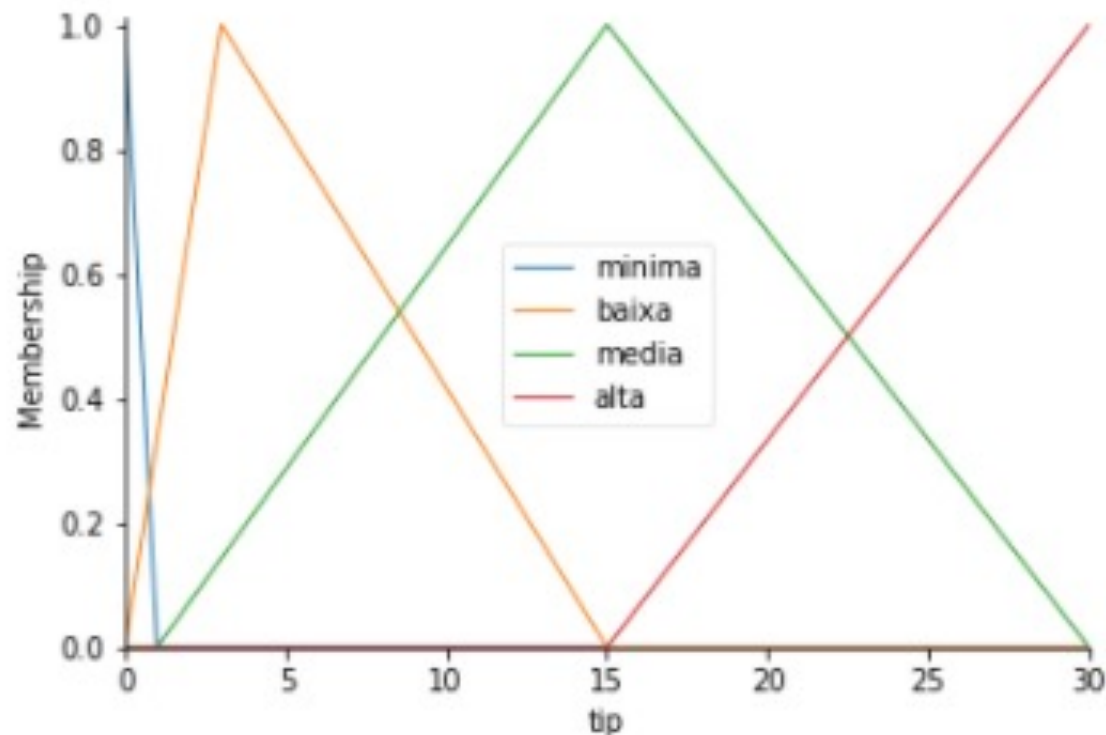
# Conjunto Fuzzy

```
# automf -> Atribuição de categorias automaticamente  
qualidade.automf(names=[ 'péssimo', 'ruim', 'ok', 'ótima' ],)  
qualidade.view()
```



# Conjunto Fuzzy

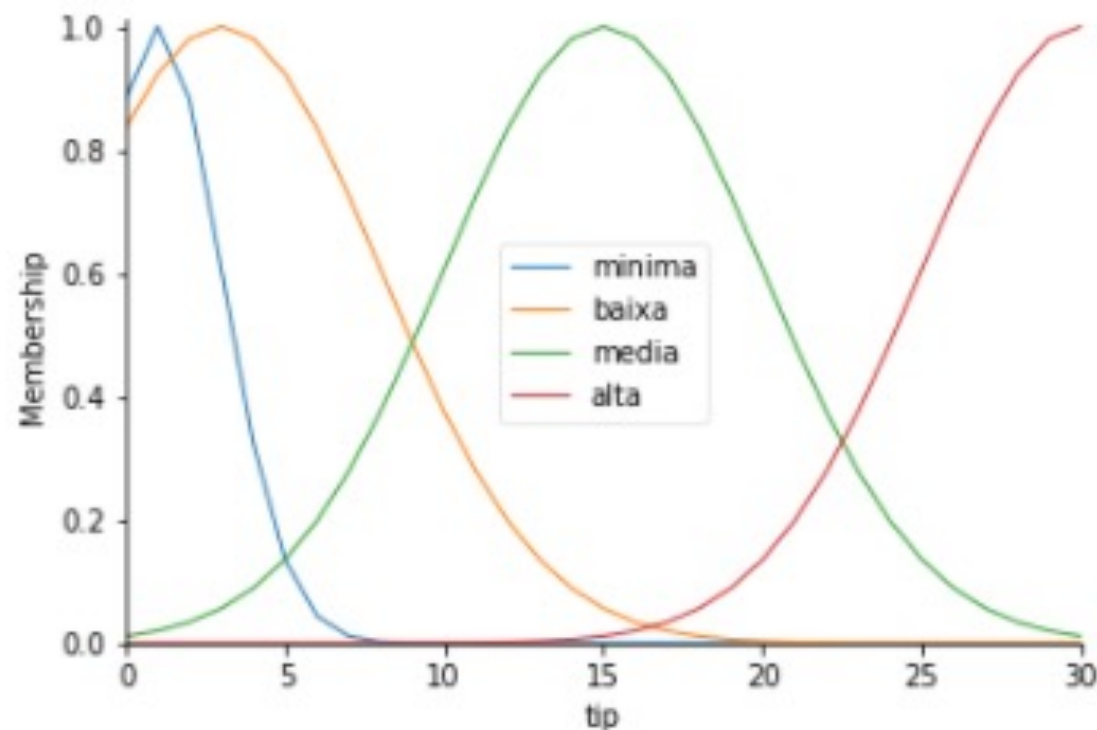
```
# sem o automf
gorjeta['minima'] = fuzz.trimf(gorjeta.universe, [-1,0,1])
gorjeta['baixa'] = fuzz.trimf(gorjeta.universe, [0,3,15])
gorjeta['media'] = fuzz.trimf(gorjeta.universe, [1,15,30])
gorjeta['alta'] = fuzz.trimf(gorjeta.universe, [15,30,45])
gorjeta.view()
```





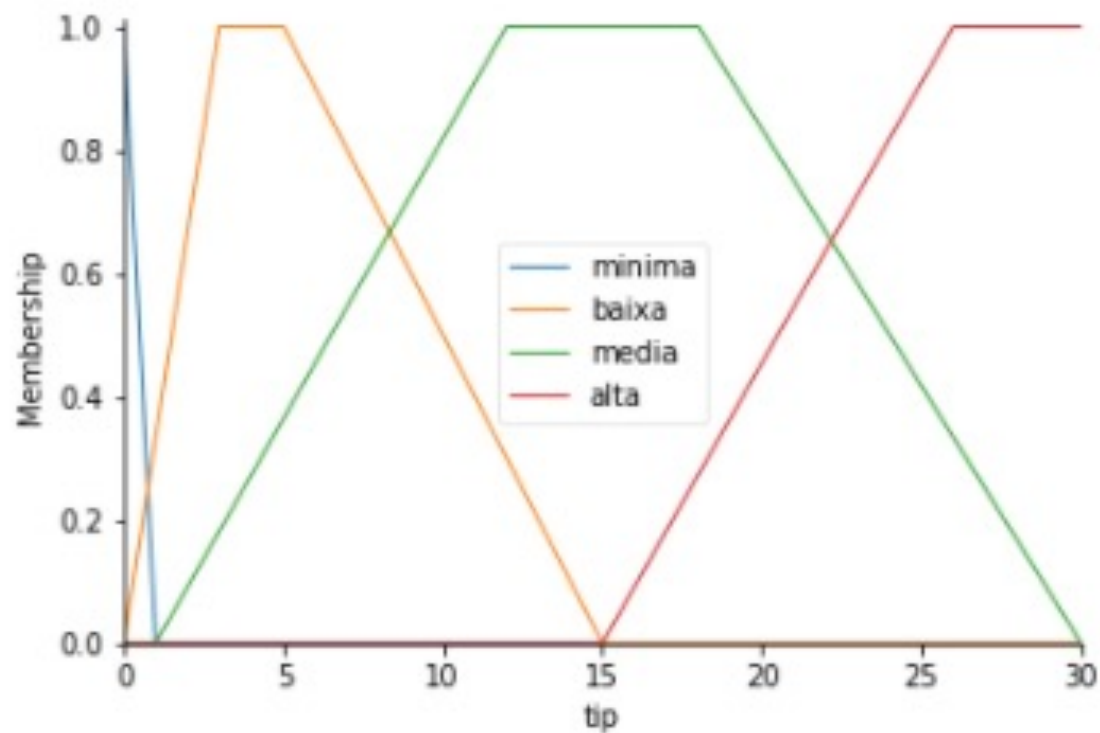
# Conjunto Fuzzy


```
# sem o automf
gorjeta['minima'] = fuzz.gaussmf(gorjeta.universe, 1,2)
gorjeta['baixa'] = fuzz.gaussmf(gorjeta.universe, 3, 5)
gorjeta['media'] = fuzz.gaussmf(gorjeta.universe, 15,5)
gorjeta['alta'] = fuzz.gaussmf(gorjeta.universe, 30,5)
gorjeta.view()
```



# Conjunto Fuzzy

```
# sem o automf
gorjeta['minima'] = fuzz.trapmf(gorjeta.universe, [-1,0,0,1])
gorjeta['baixa'] = fuzz.trapmf(gorjeta.universe, [0,3,5,15])
gorjeta['media'] = fuzz.trapmf(gorjeta.universe, [1,12,18,30])
gorjeta['alta'] = fuzz.trapmf(gorjeta.universe, [15,26,30,45])
gorjeta.view()
```





# Criando as regras

```
#Criando as regras
regra_1 = ctrl.Rule(qualidade['ruim'] & servico['ruim'], gorjeta['minima'])
regra_2 = ctrl.Rule(qualidade['ruim'] | servico['ruim'], gorjeta['baixa'])
regra_3 = ctrl.Rule(servico['medio'], gorjeta['media'])
regra_4 = ctrl.Rule(servico['bom'] | qualidade['otima'], gorjeta['alta'])

controlador = ctrl.ControlSystem([regra_1, regra_2, regra_3, regra_4])
```



# Executando o controlador

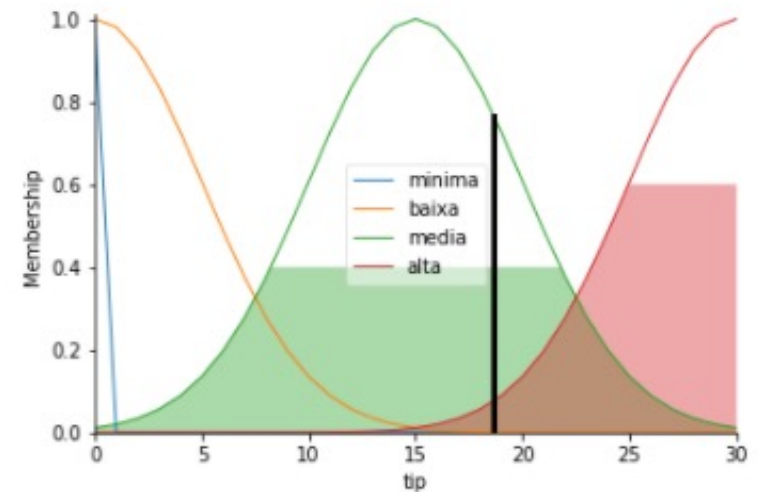
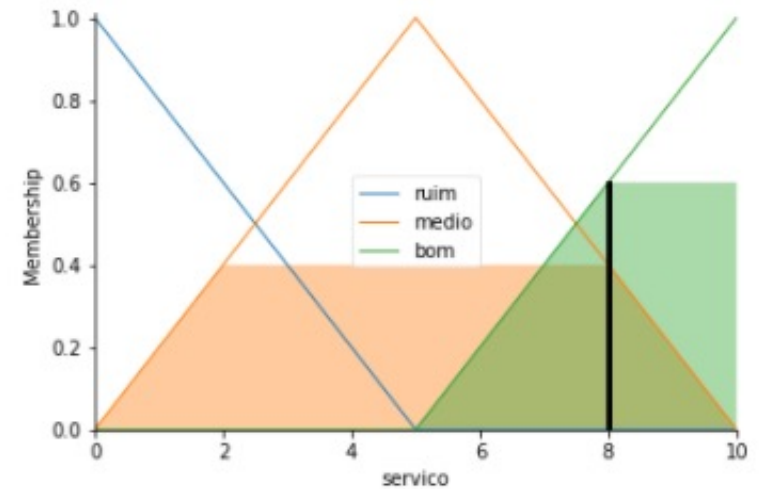
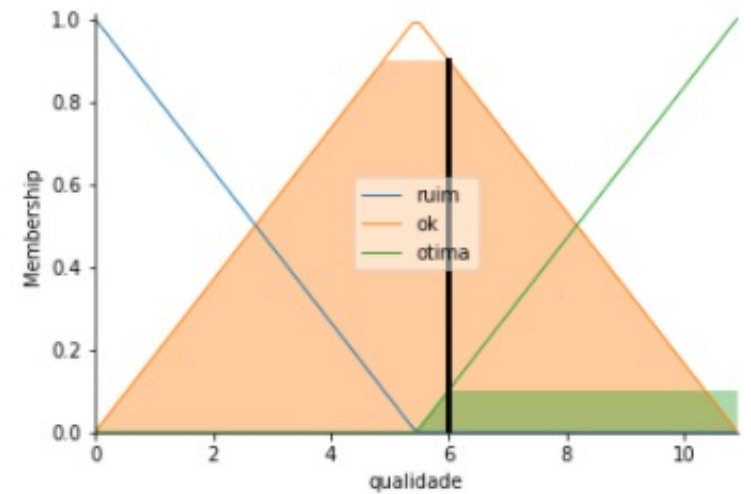
```
#Simulando
CalculoGorjeta = ctrl.ControlSystemSimulation(controlador)

CalculoGorjeta.input['qualidade'] = 6
CalculoGorjeta.input['servico'] = 8

# Crunch the numbers
CalculoGorjeta.compute()

print(CalculoGorjeta.output['tip'])
qualidade.view(sim=CalculoGorjeta)
servico.view(sim=CalculoGorjeta)
gorjeta.view(sim=CalculoGorjeta)
```

# Executando o controlador





# Relatório

- Construir o sistema para o cálculo da OBESIDADE
  - Slides Teoria
- Teste as saídas para diferentes tipos de função de pertinência (triangular, gaussiana e trapezoidal).
- Faça uma análise de sensibilidade entre as variáveis de entrada e de saída.
  - Explore os valores limites de cada uma delas!
- Crie uma nova variável de entrada e novas regras  
Exemplo: tempo de Atividade Física
- Compare os resultados entre os vários modelos
- Descreva sua experiência resolvendo este problema e dê pelo menos um outro exemplo de onde poderia aplicar a lógica fuzzy.