Python\_API\_Guides\_Control\_Flow

目录

[流控制操作 1](#_Toc497245359)

[逻辑操作 2](#_Toc497245360)

[比较操作 3](#_Toc497245361)

[调试操作 4](#_Toc497245362)

# 流控制操作

| **操作** | **描述** |
| --- | --- |
| tf. identity(input,name=None) | 返回相同形状和内容的tensor  with tf.control\_dependencies([x\_1]):  x = tf.identity(x\_1)  #x=x\_1执行前必须执行x\_1，若没有tf.identity直接使用x=x\_1，由于graph中未添加该运算，所以不会执行 |
| tf.tuple(tensors,name=None,control\_inputs=None) | 元组内张量的值仅在计算所有张量的值之后才返回 |
| tf. group(\*inputs,\*\*kwargs) | 返回一个执行所有输入op的op |
| tf. no\_op(name=None) | 什么也没做。仅用作控制流edge的占位符。 |
| tf.count\_up\_to(ref,limit,name=None) | 增加 #TODO |
| tf. cond(pred,true\_fn=None,false\_fn=None, strict=False,name=None,fn1=None,fn2=None) | 如果pred为true,返回true\_fn否则返回false\_fn |
| tf.case(pred\_fn\_pairs,default,exclusive=False, strict=False,name='case') | pred\_fn\_pairs:以下两种形式都是正确的  1. [(pred\_1, fn\_1), (pred\_2, fn\_2)]  2. {pred\_1:fn\_1, pred\_2:fn\_2}  if pred\_1:  return fn\_1()  elif pred\_2:  return fn\_2()  else:  return default() |
| tf. while\_loop(cond,body,loop\_vars, shape\_invariants=None, parallel\_iterations=10, back\_prop=True, swap\_memory=False, name=None) | 循环  loop\_vars = [...]  while cond(\*loop\_vars):  loop\_vars = body(\*loop\_vars) |

# 逻辑操作

| **操作** | **描述** |
| --- | --- |
| tf. logical\_and(x, y, name=None) | 输出x和y的按位与 |
| tf. logical\_not(x, name=None) | 输出x的按位非 |
| tf. logical\_or(x, y, name=None) | 输出x和y的按位或 |
| tf. logical\_xor(x, y, name='LogicalXor') | 输出一个带有音频的Summary缓冲区 |

# 比较操作

| **操作** | **描述** |
| --- | --- |
| tf. equal(x, y, name=None) | 输出x和y的按位== |
| tf.not\_equal(x, y, name=None) | 输出x和y的按位!= |
| tf.less(x, y, name=None) | 输出x和y的按位(x < y) |
| tf.less\_equal(x, y, name=None) | 输出x和y的按位(x <= y) |
| tf.greater(x, y, name=None) | 输出x和y的按位(x > y) |
| tf.greater\_equal(x, y, name=None) | 输出x和y的按位(x >= y) |
| tf. where(condition, x=None, y=None,name=None) | #TODO |

# 调试操作

| **操作** | **描述** |
| --- | --- |
| tf. is\_finite(x, name=None) | 返回同样大小的bool矩阵 |
| tf. is\_inf(x, name=None) | 返回同样大小的bool矩阵 |
| tf. is\_nan(x, name=None) | 返回同样大小的bool矩阵 |
| tf.verify\_tensor\_all\_finite | 输出一个带有音频的Summary缓冲区 |
| tf. check\_numerics(tensor, message, name=None) | 报告InvalidArgument错误，如果NaN或Inf |
| tf.add\_check\_numerics\_ops() | 连接check\_numerics到每个浮点张量。 |
| tf. Assert(condition,data,summarize=None, name=None) | 如果condition评估为false，打印张量列表data |
| tf. Print(input\_,data,message=None,first\_n=None, summarize=None, name=None) | 打印tensor到标准错误，不兼容jupyter |