tensor.min

```

Copy fnmin(tensors:Span>)->Tensor;

```

Returns the element-wise minumum values from a list of input tensors The input tensors must have either:

- Exactly the same shape
- The same number of dimensions and the length of each dimension is either a common length or 1.
-

Args

- tensors
- (Span>,
-) - Array of the input tensors
-

Returns

A newTensor containing the element-wise minimum values

Panics

- Panics if tensor array is empty
- Panics if the shapes are not equal or broadcastable
-

Examples

Case 1: Process tensors with same shape

```

Copy usecore::array::{ArrayTrait,SpanTrait};

useorion::operators::tensor::{TensorTrait,Tensor,U32Tensor};

fnmin_example()->Tensor { lettensor1=TensorTrait::new(shape:array![2,2].span(), data:array![0,1,2,3].span(),);
lettensor2=TensorTrait::new(shape:array![2,2].span(), data:array![0,3,1,2].span(),); letresult=TensorTrait::min(tensors:array!
[tensor1, tensor2].span()); returnresult; }

                [0,1,1,2]

result.shape

                (2,2)

```

Case 2: Process tensors with different shapes

```

Copy usecore::array::{ArrayTrait,SpanTrait};

useorion::operators::tensor::{TensorTrait,Tensor,U32Tensor};

fnmin_example()->Tensor { lettensor1=TensorTrait::new(shape:array![2,2].span(), data:array![0,1,2,3].span(),);
lettensor2=TensorTrait::new(shape:array![1,2].span(), data:array![1,4].span(),); letresult=TensorTrait::min(tensors:array!
[tensor1, tensor2].span()); returnresult; }

                [0,1,1,4]

result.shape

                (2,2)
```

```

Previous tensor.min_in_tensor Next tensor.max_in_tensor

Last updated3 months ago
```