1 Introduction

The L^2-norm, or Euclidean distance, is often used to measure the distance between two points in an n-dimensional vector space. However, it can be slow to compute as it requires finding the square root of a number. As such, it is common to use the squared L^2-norm instead, which is often notated $\|\cdot\|_2^2$.

2 Implementation

Here is an implementation of the squared L^2-norm for 2D vectors.

```
\fun{SqL2Norm2D}{v}{
  \fun{Sq}{n}{ return n * n; }
  SqX := Sq(v.x);
  SqY := Sq(v.y);
  return SqX + SqY;
}
```

3 Usage

To use `SqL2Norm2D`, you will need a class that encapsulates a vector.

```
\class{Vector2}{x, y}{
  self.x = x;
  self.y = y;
}
```

Let's test it out

```
\fun{main}{}{
  vector := Vector2(3, 4);
  print(SqL2Norm2D(vector));
}
```

This program should print 25 to the console when run.

4 Conclusion

`SqL2Norm2D` is a really simple function and is fast on most architectures. Note that the factoring out of the `Sq` function may lead to a slight slowdown due to function call overhead. As such, it may be good to consider inlining the entire computation into a single return statement.