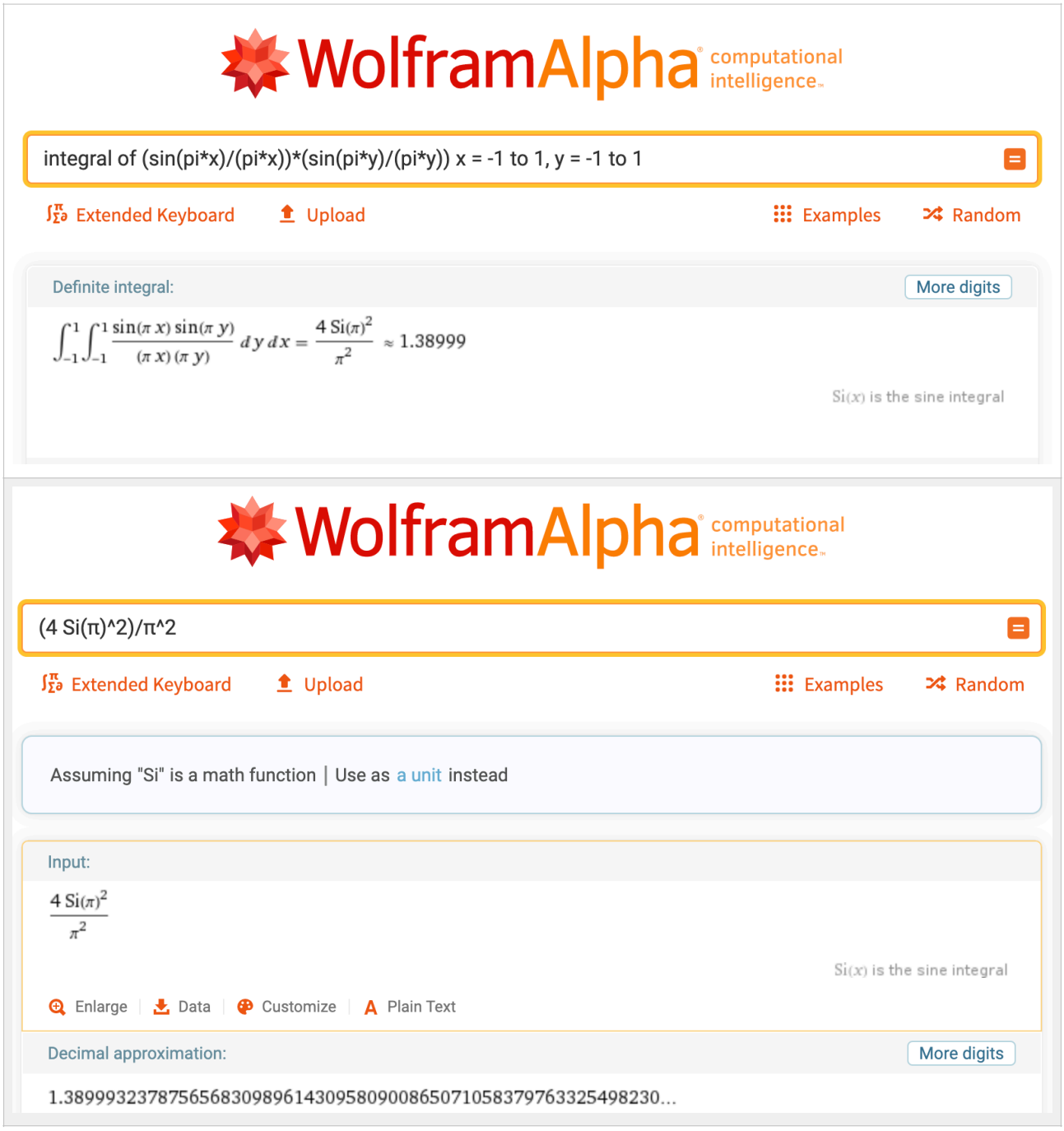
Numerical Analysis HW 3

2D Gaussian Quadrature

學號：00557152
姓名：林君翰
系級：資工 4 B

1. Exact solution by Wolfram Alpha



2. 2D Gaussian quadrature

```
~/code/cpp/numerical_analysis/hw3  g++ --std=c++11 gaus_quad.cpp
~/code/cpp/numerical_analysis/hw3  ./a.out      23:08:30  13%
n_sample, split = (2, 1), 1.1454595652, error = 0.2445336727
n_sample, split = (2, 4), 1.3849457570, error = 0.0050474809
n_sample, split = (2, 9), 1.3890933738, error = 0.0008998641
n_sample, split = (3, 1), 1.4062731703, error = 0.0162799325
n_sample, split = (3, 4), 1.3900517841, error = 0.0000585462
n_sample, split = (3, 9), 1.3899977561, error = 0.0000045182
n_sample, split = (4, 1), 1.3894075087, error = 0.0005857291
n_sample, split = (4, 4), 1.3899928244, error = 0.0000004135
n_sample, split = (4, 9), 1.3899932240, error = 0.0000000139
```

3. Can we improve the accuracy by dividing D into a finer mesh, if the same number of sample points is fixed? Can we improve the accuracy by using more sample points, if the resolution of the mesh is fixed?

兩者皆可。以下是我把 D 切成 1~100 份的結果，可以發現將 domain 切越多塊，精準度可以上升得越高。
又可以發現，增加 sample 點數也可以有效增加精確度。

```
n_sample, split = (2, 1), 1.1454595652, error = 0.2445336727
n_sample, split = (2, 4), 1.3849457570, error = 0.0050474809
n_sample, split = (2, 9), 1.3890933738, error = 0.0008998641
n_sample, split = (2, 16), 1.3897179439, error = 0.0002752940
n_sample, split = (2, 25), 1.3898821832, error = 0.0001110547
n_sample, split = (2, 36), 1.3899401184, error = 0.0000531195
n_sample, split = (2, 49), 1.3899647057, error = 0.0000285321
n_sample, split = (2, 64), 1.3899765659, error = 0.0000166720
n_sample, split = (2, 81), 1.3899828522, error = 0.0000103856
n_sample, split = (2, 100), 1.3899864344, error = 0.0000068035
```

```
n_sample, split = (3, 1), 1.4062731703, error = 0.0162799325
n_sample, split = (3, 4), 1.3900517841, error = 0.0000585462
n_sample, split = (3, 9), 1.3899977561, error = 0.0000045182
n_sample, split = (3, 16), 1.3899940088, error = 0.0000007709
n_sample, split = (3, 25), 1.3899934362, error = 0.0000001983
n_sample, split = (3, 36), 1.3899933036, error = 0.0000000657
n_sample, split = (3, 49), 1.3899932638, error = 0.0000000259
n_sample, split = (3, 64), 1.3899932495, error = 0.0000000116
n_sample, split = (3, 81), 1.3899932436, error = 0.0000000057
n_sample, split = (3, 100), 1.3899932409, error = 0.0000000030
```

```
n_sample, split = (4, 1), 1.3894075087, error = 0.0005857291
n_sample, split = (4, 4), 1.3899928244, error = 0.0000004135
n_sample, split = (4, 9), 1.3899932240, error = 0.0000000139
n_sample, split = (4, 16), 1.3899932365, error = 0.0000000013
n_sample, split = (4, 25), 1.3899932377, error = 0.0000000002
n_sample, split = (4, 36), 1.3899932378, error = 0.0000000000
n_sample, split = (4, 49), 1.3899932379, error = 0.0000000000
n_sample, split = (4, 64), 1.3899932379, error = 0.0000000000
n_sample, split = (4, 81), 1.3899932379, error = 0.0000000000
n_sample, split = (4, 100), 1.3899932379, error = 0.0000000000
```

4. Which factor is more important, the number of sample point or the mesh resolution?

分析 Truncation error 可得知，若 sample points 為 n，則 error(n+1)/error(n) 會趨近於 1/(n^2) ，也就是若 n+1，則精準度提高 n^2 倍，而提高 mesh resolution 的精準度成長則較少，因此我認為 sample point 的數量是比較重要的因素。

5. Relative errors graphs