

Docogen Example

Kevin Cyu

kevinbird61@gmail.com

Nation Cheng Kung University

Yung-Sheng Lu

yungshenglu1994@gmail.com

Nation Chiao Tung University

November 19, 2017

Abstract

I went down to the river, I set down on the bank. I tried to think but couldn't, So I jumped in and sank.

1 Getting Start

Merging test

- Building your document and website together.
- New feature support. Next-line testing.

1.1 Why we create Docogen?

Why we create Docogen?

- Building your document and website together.
- Generate a beautiful introduction paper with simple command.

2 Introduction

What is Docogen?

- Building your document and website together.
- New feature support. Next-line testing.

2.1 Why we create Docogen?

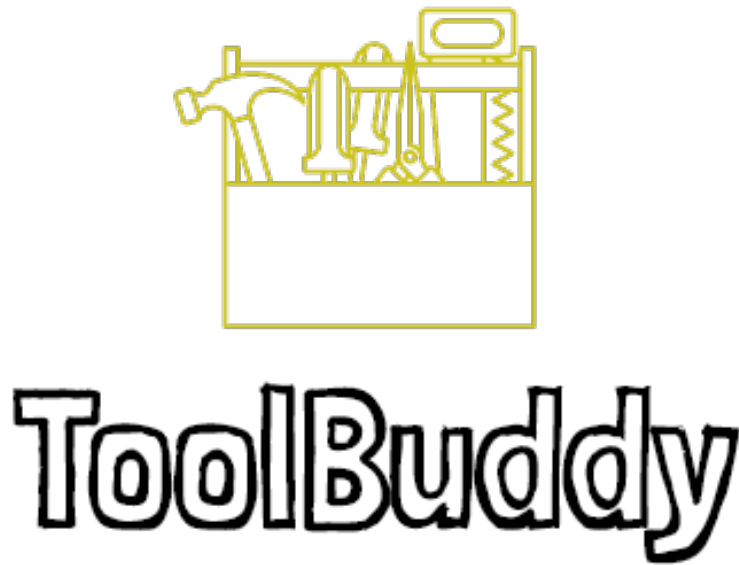
Why we create Docogen?

- Building your document and website together.
- Generate a beautiful introduction paper with simple command.

3 About us

What is toolbuddy[2]?

- An group of good programmer that solve the problem!



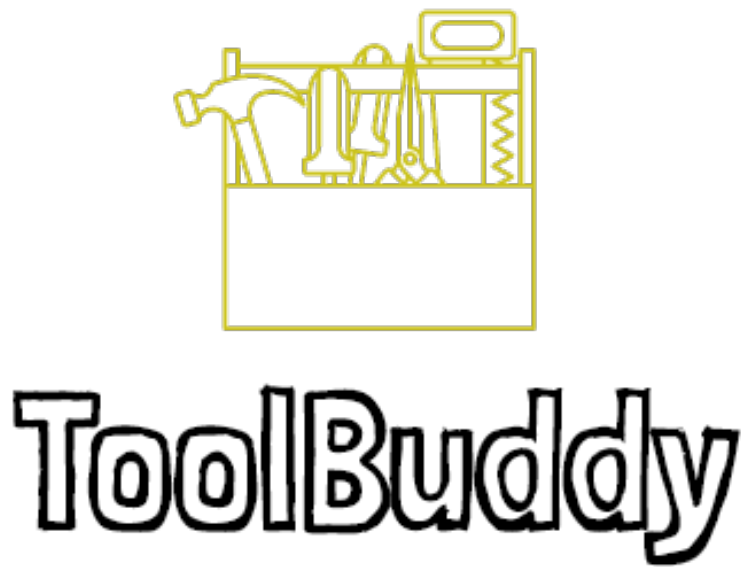
SQUARESPACE.COM/LOGO – ICONS BY THE NOUN PROJECT

Figure 1: Relative ToolBuddy logo

3.1 How to join toolbuddy?

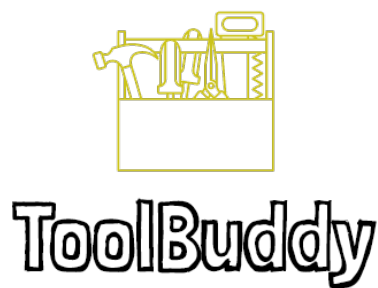
The way:

- Just email to Kevin and pass your github ID and he will find your by himself!



SQUARESPACE.COM/LOGO – ICONS BY THE NOUN PROJECT

Figure 2: Absolute ToolBuddy logo



SQUARESPACE.COM/LOGO – ICONS BY THE NOUN PROJECT

Figure 3: Sub Rel ToolBuddy logo

4 Different usage of content

List Structure

List Structure Example:

Demo

1. List 1
 - (a) List 1-1
 - (b) List 1-2
 - i. List 1-2-1
 - ii. List 1-2-2
 - A. List 1-2-2-1
 - (c) List 1-3

2. List 2

List

- Listing structure append

- New feature support.

5 Table Demo

Table 1		
Name	Age	Job
Kevin	23	programmer
Eric	22	student
Lu	24	engineer
Cyu	52	professor
Lai	50	soldier

6 Code listing Demo

Code listing 1

```

1 #include <stdio.h>
2
3 int main(){
4     return 0;}

```

Listing 1: C mini exampe

```

1 import numpy as np
2
3 def incmatrix(genl1,genl2):
4     m = len(genl1)
5     n = len(genl2)
6     M = None #to become the incidence matrix
7     VT = np.zeros((n*m,1), int) #dummy variable
8
9     #compute the bitwise xor matrix
10    M1 = bitxormatrix(genl1)
11    M2 = np.triu(bitxormatrix(genl2),1)
12
13    for i in range(m-1):
14        for j in range(i+1, m):
15            [r,c] = np.where(M2 == M1[i,j])
16            for k in range(len(r)):
17                VT[(i)*n + r[k]] = 1;
18                VT[(i)*n + c[k]] = 1;
19                VT[(j)*n + r[k]] = 1;
20                VT[(j)*n + c[k]] = 1;
21
22            if M is None:
23                M = np.copy(VT)
24            else:
25                M = np.concatenate((M, VT), 1)
26
27            VT = np.zeros((n*m,1), int)
28
29    return M

```

Listing 2: Python example

```

1 /* Prime modulus multiplicative linear congruential generator
2    Z[i] = (630360016 * Z[i-1]) (mod(pow(2,31) - 1)), based on Marse
   and Roberts'
3    portable FORTRAN random-number generator UNIRAN. Multiple (100)
   streams are
4    supported, with seeds spaced 100,000 apart. Throughout, input
   argument
5    "stream" must be an int giving the desired stream number. The
   header file
6    lcgrand.h must be included in the calling program (#include "
   lcgrand.h")
7    before using these functions.
8
9    Usage: (Three functions)
10
11    1. To obtain the next U(0,1) random number from stream "stream,"
   execute

```

```

12     u = lcgrand(stream);
13     where lcgrand is a float function. The float variable u will
14     contain the
15     next random number.
16
17 2. To set the seed for stream "stream" to a desired value zset,
18     execute
19     lcgrandst(zset, stream);
20     where lcgrandst is a void function and zset must be a long
21     set to the
22     desired seed, a number between 1 and 2147483646 (inclusive).
23     Default
24     seeds for all 100 streams are given in the code.
25
26 3. To get the current (most recently used) integer in the
27     sequence being
28     generated for stream "stream" into the long variable zget,
29     execute
30     zget = lcgrandgt(stream);
31     where lcgrandgt is a long function. */
32
33 /* Define the constants. */
34
35 #define MODLUS 2147483647
36 #define MULT1 24112
37 #define MULT2 26143
38
39 /* Set the default seeds for all 100 streams. */
40
41 static long zrng[] =
42 {
43     1,
44     1973272912, 281629770, 20006270, 1280689831, 2096730329, 1933576050,
45     913566091, 246780520, 1363774876, 604901985, 1511192140, 1259851944,
46     824064364, 150493284, 242708531, 75253171, 1964472944, 1202299975,
47     233217322, 1911216000, 726370533, 403498145, 993232223, 1103205531,
48     762430696, 1922803170, 1385516923, 76271663, 413682397, 726466604,
49     336157058, 1432650381, 1120463904, 595778810, 877722890, 1046574445,
50     68911991, 2088367019, 748545416, 622401386, 2122378830, 640690903,
51     1774806513, 2132545692, 2079249579, 78130110, 852776735, 1187867272,
52     1351423507, 1645973084, 1997049139, 922510944, 2045512870, 898585771,
53     243649545, 1004818771, 773686062, 403188473, 372279877, 1901633463,
54     498067494, 2087759558, 493157915, 597104727, 1530940798, 1814496276,
55     536444882, 1663153658, 855503735, 67784357, 1432404475, 619691088,
56     119025595, 880802310, 176192644, 1116780070, 277854671, 1366580350,
57     1142483975, 2026948561, 1053920743, 786262391, 1792203830, 1494667770,
58     1923011392, 1433700034, 1244184613, 1147297105, 539712780, 1545929719,
59     190641742, 1645390429, 264907697, 620389253, 1502074852, 927711160,
60     364849192, 2049576050, 638580085, 547070247 };
61
62 /* Generate the next random number. */
63
64 float lcgrand(int stream)
65 {
66     long zi, lowprd, hi31;
67
68     zi = zrng[stream];
69     lowprd = (zi & 65535) * MULT1;

```

```

63     hi31    = (zi >> 16) * MULT1 + (lowprd >> 16);
64     zi      = ((lowprd & 65535) - MODLUS) +
65               ((hi31 & 32767) << 16) + (hi31 >> 15);
66     if (zi < 0) zi += MODLUS;
67     lowprd = (zi & 65535) * MULT2;
68     hi31    = (zi >> 16) * MULT2 + (lowprd >> 16);
69     zi      = ((lowprd & 65535) - MODLUS) +
70               ((hi31 & 32767) << 16) + (hi31 >> 15);
71     if (zi < 0) zi += MODLUS;
72     zrng[stream] = zi;
73     return (zi >> 7 | 1) / 16777216.0;
74 }
75
76
77 void lcgrandst (long zset, int stream) /* Set the current zrng for
78                                         stream
79                                         "stream" to zset. */
79 {
80     zrng[stream] = zset;
81 }
82
83
84 long lcgrandgt (int stream) /* Return the current zrng for stream "
85                               stream". */
85 {
86     return zrng[stream];
87 }

```

Listing 3: C example

7 Formula Demo

Formula 1

Now we will introduce the basic equation usage (inline mode): $x^2 + y^2 = z^2$
Then we can see the display mode:

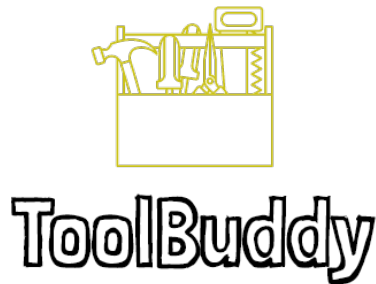
$$x^n + y^n = z^n$$

And about equation tag:

$$E = mc^2 \tag{1}$$

8 Image Demo

Image/Figure inside the content



SQUARESPACE.COM/LOGO - ICONS BY THE NOUN PROJECT

Figure 4: ToolBuddy logo

9 Web extension

Restful Api support

[Online] Register New User

Method: post

Url: <https://kevin.imslab.org/register>

Description: Enroll new user to local service

Parameter:

Field Name	Data Type
username	String
password	String
email	String

Error Msg: duplicated internal error

Success Msg: success

[Online] Checking mail

Method: get

Url: <https://kevin.imslab.org/checkmail>

Description: Enroll new user to local service

Parameter:

Field Name	Data Type
email	String

Error Msg: internal error

Success Msg: existed not found

10 Relative Image Demo

Image/Figure Relative Test

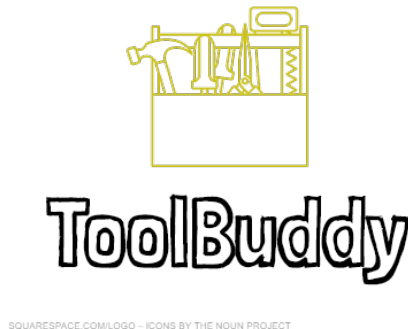


Figure 5: Test ToolBuddy logo

References

- [1] Kevin Cyu, From NCKU, personal website: <https://github.com/kevinbird61>
- [2] ToolBuddy, A good, non-profit organization.