

راهنمای برنامه‌ی پیاده‌سازی الگوریتم
رمزنگاری SPN
Substitution Permutation Network

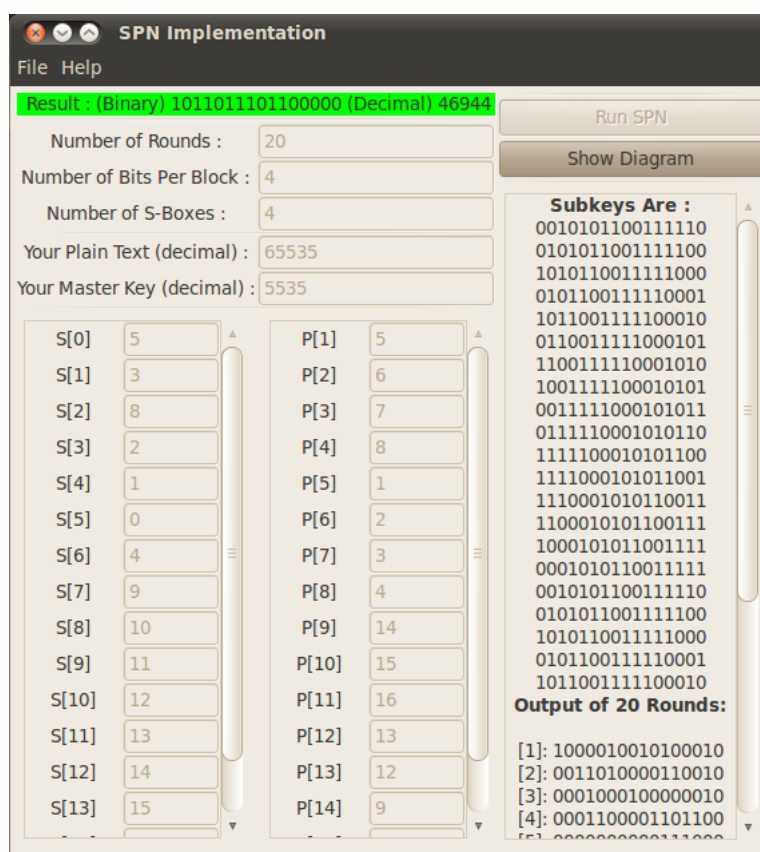
بهرام بهرام‌بیگی

۲ آبان ۱۳۹۲

۱۰۰ این برنامه چیست ؟

این برنامه یک پیاده‌سازی از الگوریتم SPN^۱ است که در دو نسخه‌ی CLI^۲ و GTK^۳ پیاده‌سازی شده است. در نسخه‌ی متنی برنامه در حالت متنی ترمینال اجرا شده و در نسخه‌ی گرافیکی برنامه ورودی‌ها و خروجی‌ها را به صورت کاملاً گرافیکی از کاربر گرفته و نمایش می‌دهد. این برنامه بعد از گرفتن تعداد دورها، تعداد بیت‌های هر بلاک و تعداد بلاک‌ها و همچنین متن مورد نظر جهت رمز شدن و کلید اصلی، دو جدول S-Box و P-Box را گرفته و اجرا شده و نتیجه را به همراه خروجی هر مرحله و زیرکلیدهای هر مرحله نمایش می‌دهد. ویژگی کلیدی این برنامه امکان انتخاب هر تعداد بیت و بلاک برای اجرای الگوریتم است.

در شکل زیر نمونه‌ای از نحوه‌ی اجرای این برنامه را در سیستم عامل Ubuntu 10.04 می‌بینید :



^۱ Substitution Permutation Network

^۲ Command Line Interface

^۳ GUI Version (GIMP Toolkit) : <http://www.gtk.org>

۲.۰ نحوه‌ی نصب این برنامه

۱.۲.۰ پیش‌نیازهای این برنامه

این برنامه برای اجرا نیاز به این پیش‌نیازها دارد :

۱. Python نسخه‌ی دوم : در بیشتر توزیع‌های گنو/لینوکس پایتون ۲ به صورت پیشفرض نصب است. در غیر اینصورت می‌توانید آن را با اسم python یا python2 در مخازن توزیع خود پیدا کنید. این برنامه با پایتون نسخه‌ی سوم کار نخواهد کرد !

۲. PyGTK2, GTK2 : این دو کتابخانه فقط برای نسخه‌ی گرافیکی برنامه مورد نیاز است. در صورتی که از نسخه‌ی متنی برنامه استفاده می‌کنید به این دو کتابخانه نیاز نخواهید داشت. این دو کتابخانه هم در بیشتر توزیع‌های گنو/لینوکس به صورت پیشفرض نصب است. در غیر اینصورت می‌توانید دو بسته‌ی gtk, pygtk را در مخازن توزیع خود جستجو کنید یا اینکه آن‌ها را به ترتیب از دو سایت <http://www.pygtk.org> و <http://www.gtk.org> دانلود و طبق دستورالعمل‌های موجود عمل کنید. همانند نسخه‌ی پایتون، نسخه‌ی سوم از این کتابخانه‌ها با برنامه منطبق نیستند ! (فقط نسخه‌ی دوم کار خواهند کرد)

۳. PyGLET : این کتابخانه معمولاً به طور پیشفرض بر روی توزیع‌ها نصب نیست. از این کتابخانه که یک واسط برای OpenGL است برای رسم دیاگرام نهایی استفاده می‌شود. ولی در مخازن بیشتر توزیع‌های گنو/لینوکس وجود دارد. برای مثال برای نصب آن در اوبونتو دستور زیر را در ترمینال وارد کنید :

```
sudo apt-get install python-pyglet
```

یا به سادگی در رابط گرافیکی دنبال pyglet بگردید ! در صورتی که به هر دلیلی آن را پیدا نکردید، برای دانلود آن به سایت <http://www.pyglet.org> بروید و از صفحه‌ی دانلود آن را بگیرید. بعد از دانلود آن را از حالت فشرده خارج و به مسیر دایرکتوری استخراج شده بروید و دستور زیر را برای نصب این کتابخانه وارد کنید :

```
sudo python setup.py install
```

۲.۲.۰ نحوه‌ی اجرای برنامه

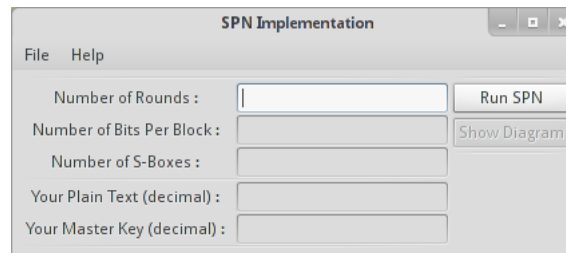
برنامه نیازی به نصب ندارد ! در صورتی که تمامی این پیش‌نیازهای بالا را داشته باشید به راحتی برای اجرای برنامه در حالت متنی کافی است دستور زیر را در مسیری که برنامه وجود دارد وارد کنید :

```
python SPN_CLI.py
```

همچنین با زدن دستور زیر می‌توانید نسخه‌ی گرافیکی را اجرا کنید (قبلاً به مسیر جاری رفته باشید) :

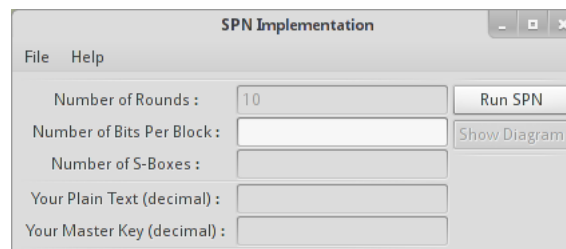
```
python SPN_GTK.py
```

با زدن این دستور بایستی برنامه به شکلی شبیه زیر برایتان ظاهر شود :



۳.۰ نحوه‌ی استفاده از این برنامه

نحوه‌ی استفاده از رابط گرافیکی را توضیح می‌دهیم و استفاده از رابط متنی ساده خواهد بود. البته برای تست سریع برنامه در حالت متنی می‌توانید از فایلی به نام SPN_CLI_Inputs در مسیر جاری برنامه استفاده کنید و مقادیر موجود در فایل را به صورت مستقیم در محیط متنی کپی کنید. برای استفاده از محیط گرافیکی طبق بخش قبلی صفحه‌ی ابتدایی برنامه برای شما ظاهر می‌شود. ورودی‌های برنامه به ترتیب از کاربر گرفته می‌شود. برای مثال در ابتدا تعداد دوره‌های الگوریتم^۴ از کاربر گرفته می‌شود که برای وارد کردن مقدار بعدی برنامه کافی است کاربر کلید ENTER یا TAB را فشرده و مقدار فعلی تثبیت شده (فیلد فعلی غیرفعال می‌شود) و فیلد بعدی برای گرفتن مقدار فعال می‌شود. برای مثال در شکل زیر می‌بینید که مقدار ۱۰ برای تعداد دورها وارد شده است :



در هر مرحله که مقدار نامناسب وارد شود در قسمت پیغام برنامه (سمت چپ بالا) پیغامی مناسب جهت رفع مشکل نمایش داده می‌شود. بعد از وارد کردن تعداد دورها، تعداد بیت‌ها در هر بلاک و تعداد بلاک‌ها به طور درست، برنامه حداکثر مقداری که می‌توان برای متن ساده (جهت رمز کردن) و کلید اصلی انتخاب کرد را در فیلد متن ساده وارد می‌کند (شکل زیر):

^۴Number of rounds

SPN Implementation

File Help

Number of Rounds : 10

Number of Bits Per Block : 5

Number of S-Boxes : 4

Your Plain Text (decimal) : 1048575

Your Master Key (decimal) : up to 1048575

Run SPN

Show Diagram

بعد از پرکردن پنج فیلد اولیه‌ی برنامه به طور درست و صحیح، جداول S-Box و P-Box برای کاربر نمایش داده می‌شود که به ترتیب جداول جایگزینی و درهم‌سازی را معرفی می‌کند. در فیلدهای این جداول هم در صورتی که مقدار نامناسبی درج شود، به کاربر اخطار داده می‌شود. برای مثال در شکل زیر مقدار ۲۰ برای یکی از فیلدها درج شده است (که حداکثر آن ۱۵ است) :

SPN Implementation

File Help

Error! Very Big value in SBox[2]!

Number of Rounds : 10

Number of Bits Per Block : 4

Number of S-Boxes : 5

Your Plain Text (decimal) : 1048575

Your Master Key (decimal) : 48575

Run SPN

Show Diagram

S[0]	12
S[1]	13
S[2]	20
S[3]	
S[4]	
S[5]	
S[6]	
S[7]	
S[8]	
S[9]	
S[10]	
S[11]	
S[12]	
S[13]	
S[14]	
S[15]	

P[1]	
P[2]	
P[3]	
P[4]	
P[5]	
P[6]	
P[7]	
P[8]	
P[9]	
P[10]	
P[11]	
P[12]	
P[13]	
P[14]	
P[15]	
P[16]	

بعد از پرکردن تمامی فیلدهای جدول جایگزینی، برای جدول P-Box بدلیل اینکه بایستی اعداد وارد شده حالت تقارنی داشته باشند، برنامه به طور خودکار این کار را انجام می‌دهد. برای مثال در شکل زیر فیلد اول ۷ درج شده است که در فیلد هفتم هم عدد ۱ درج شده است و به همین ترتیب ادامه پیدا می‌کند تا این جدول هم کامل شود :

SPN Implementation

File Help

Number of Rounds : 10 Run SPN

Number of Bits Per Block : 4 Show Diagram

Number of S-Boxes : 5

Your Plain Text (decimal) : 1048575

Your Master Key (decimal) : 48575

S[0]	12
S[1]	13
S[2]	11
S[3]	1
S[4]	5
S[5]	0
S[6]	3
S[7]	8
S[8]	10
S[9]	14
S[10]	2
S[11]	7
S[12]	15
S[13]	9
S[14]	4
S[15]	6

P[1]	7
P[2]	8
P[3]	5
P[4]	
P[5]	3
P[6]	
P[7]	1
P[8]	2
P[9]	
P[10]	
P[11]	
P[12]	
P[13]	
P[14]	
P[15]	
P[16]	

حال می‌توان برنامه را با فشردن دکمه‌ی Run SPN اجرا کرده و خروجی را در سمت راست و زیر همان دکمه مشاهده کرد. نتیجه‌ی نهایی در قسمت پیغام‌های برنامه و با رنگ سبز نمایش داده می‌شود :

SPN Implementation

File Help

Result: (Binary) 10010001010011001111 (Decimal) 59515

Number of Rounds : 10

Number of Bits Per Block : 4

Number of S-Boxes : 5

Your Plain Text (decimal) : 1048575

Your Master Key (decimal) : 48575

Run SPN

Show Diagram

Subkeys Are :

```

00010111101101111110
00101111011011111100
01011110110111111000
10111101101111110000
01111011011111100001
11110110111111000010
11101101111110000101
11011011111100001011
10110111111000010111
01101111110000101111
1101111100001011110

```

Output of 10 Rounds:

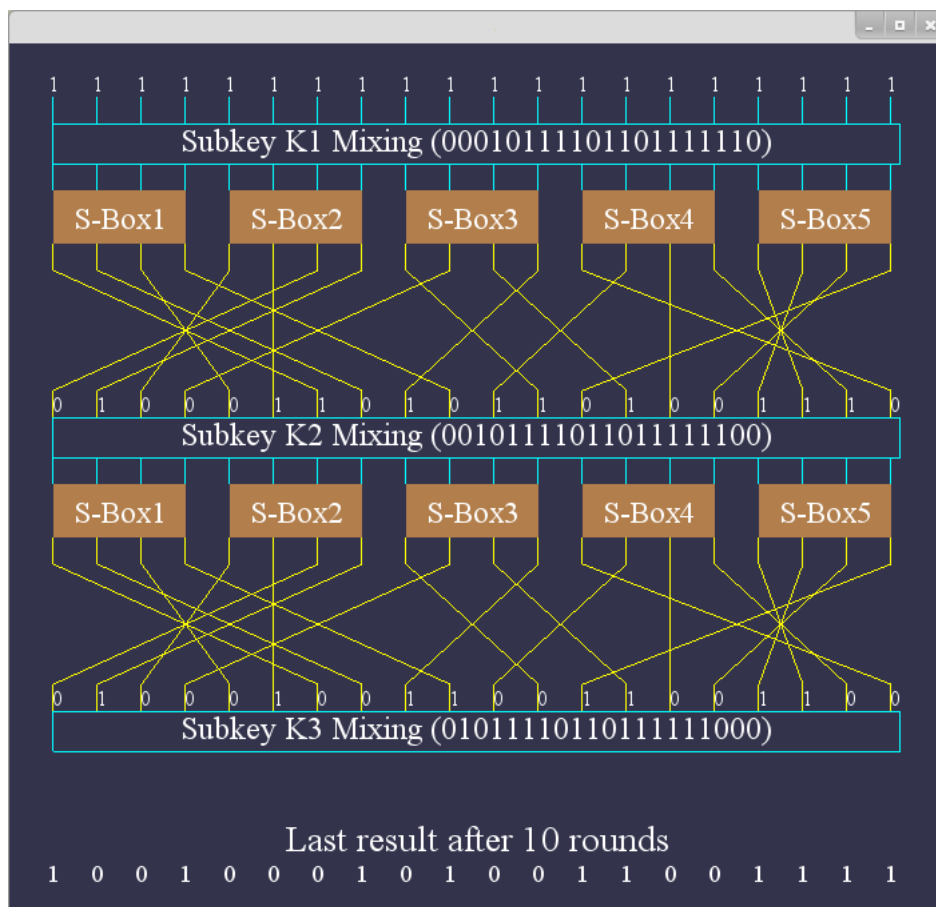
```

[1]: 01000110101101001110
[2]: 01000100110011001100
[3]: 00010011001110010111
[4]: 00110000100010000010
[5]: 00001010010110001110
[6]: 11000010000010110011
[7]: 00100011011110000011
[8]: 01010110101010001010
[9]: 00100110101111010010
[10]: 01001110110010010001

```

S[0]	12	P[1]	7
S[1]	13	P[2]	8
S[2]	11	P[3]	5
S[3]	1	P[4]	10
S[4]	5	P[5]	3
S[5]	0	P[6]	6
S[6]	3	P[7]	1
S[7]	8	P[8]	2
S[8]	10	P[9]	12
S[9]	14	P[10]	4
S[10]	2	P[11]	14
S[11]	7	P[12]	9
S[12]	15	P[13]	20
S[13]	9	P[14]	11
S[14]	4	P[15]	15
S[15]	6	P[16]	19

در صورتی که خطایی پیش نیامده باشد، دکمه‌ی Show Diagram هم فعال شده و با فشردن این دکمه، دیاگرام شبیه‌سازی شده از الگوریتم اجرا شده به نمایش در می‌آید. البته به دلیل محدودیت فضا تنها سه مرحله از اجرای الگوریتم به نمایش در می‌آید. شکل زیر نمونه‌ای از نمایش دیاگرام تولید شده را نشان می‌دهد :



۴۰۰ این برنامه چگونه کار می کند ؟

همان طور که در ابتدا هم ذکر گردید، این برنامه یک پیاده سازی ساده از الگوریتم رمزنگاری SPN به طور اصلی جهت اهداف آموزشی است. این برنامه جهت ترکیب کلید و متن ساده از یای انحصاری (XOR) استفاده می کند و همچنین برای بدست آوردن زیرکلید هر مرحله از شیفت کلید اصلی استفاده می کند. هر کدام از این موارد را می توان به دلیل توضیحات کافی در کد، تغییر داده و به ترکیب یا درهم سازی دلخواه تغییر داد.

در این پیاده سازی می توان برای الگوریتم SPN هر تعداد مرحله یا Round تعریف کرد و هر تعداد S-Box و با ظرفیت متغیر به آن اضافه کرد. البته چون این برنامه با زبان پایتون پیاده سازی شده اعداد بزرگ محدود به حداکثر امکان این زبان است. این برنامه از خطاهایی که ممکن است هنگام وارد کردن اعداد پیش بیاید جلوگیری می کند. اعداد را به صورت دهدهی یا Decimal از کاربر می گیریم تا فهم آن برای کاربر راحت تر باشد. در طول برنامه تبدیل به دودویی و بالعکس ممکن است چندین بار انجام شود. بعد از وارد کردن تعداد دورها و تعداد ها S-Box و ظرفیت آنها (تعداد بیت ها)، از کاربر متن ساده (PlainText) گرفته می شود. قبل از گرفتن به کاربر گفته می شود که با توجه به اعداد وارد شده، عدد متن اصلی باید در

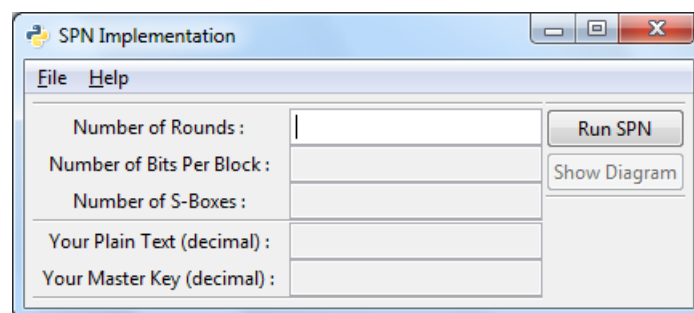
چه محدوده‌ای باشد و بیشتر از آن توسط برنامه پذیرفته نخواهد شد. بعد جداول S-Box و سپس P-Box درایه به درایه از کاربر گرفته می‌شود. در صورتی که نیاز نباشد هربار این جداول تغییر کنند، می‌توان برای تسریع کار آن‌ها در متن برنامه به صورت دستی وارد کرد. در هنگام وارد کردن درایه‌های جدول S-Box از وارد کردن دو مقدار برای عناصر مختلف جدول جلوگیری می‌شود. برای مثال نمی‌توان به دو درایه‌ی $sbox[2]$ و $sbox[9]$ عدد ۲۰ را نسبت داد. و بایستی دو عدد متفاوت نسبت داده شود تا یک به یک بودن جدول حفظ شود. همچنین در هنگام وارد کردن اعداد برای جدول P-Box، مقادیر برعکس به طور اتوماتیک به درایه‌ها داده می‌شود و به کاربر اطلاع داده می‌شود. برای مثال در صورتی که کاربر $pbox[9]$ را عدد ۱۴ وارد کند، برنامه به طور اتوماتیک درایه‌ی ۱۴ را عدد ۹ خواهد زد یعنی $pbox[14]=9$. این کار پوشا بودن و یک به یک بودن جدول P-Box را تضمین می‌کند و از اشتباه کاربر جلوگیری می‌کند.

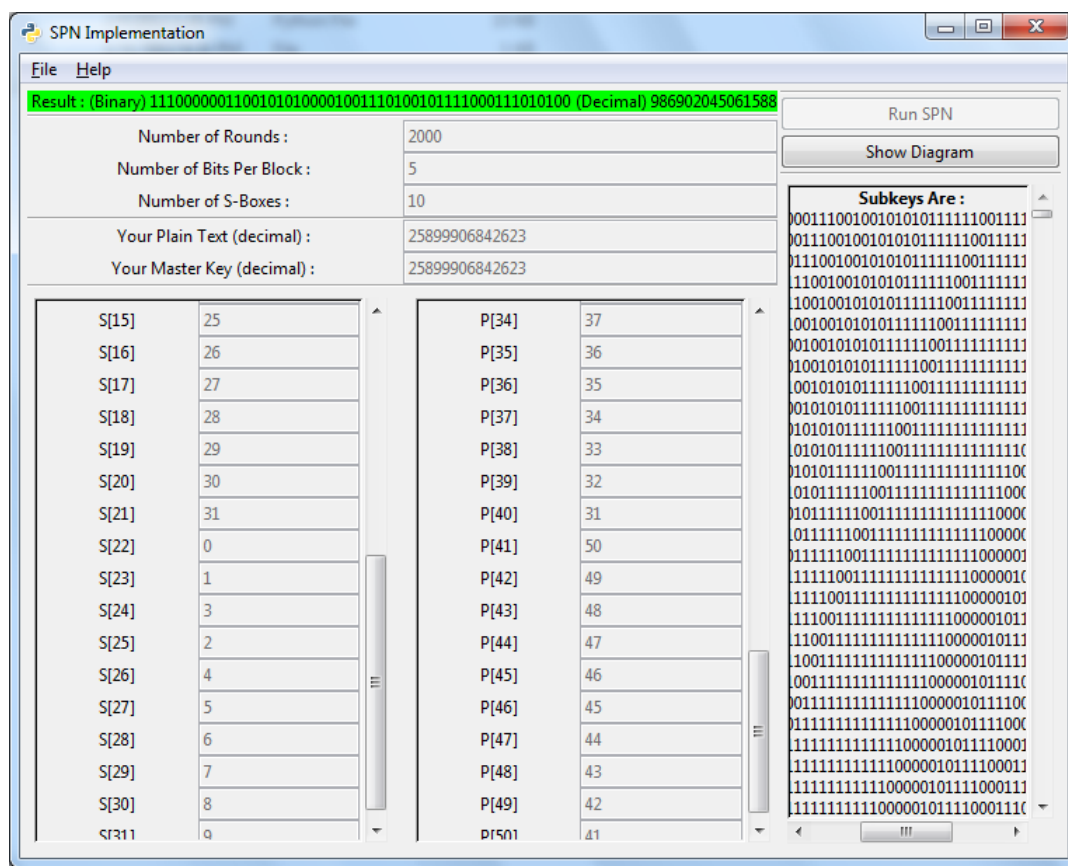
کلید اصلی بایستی طولی برابر متن اصلی (عدد اصلی وارد شده) داشته باشد. که البته چون اعداد به صورت دهدهی وارد می‌شوند فقط کافی است که عددی که وارد می‌شود از مقدار حداکثر تجاوز نکند، برنامه به طور خودکار در حین تبدیل به دودویی، طول‌ها را مساوی خواهد کرد. برنامه به میزان یک عدد بیشتر از تعداد دورها زیرکلید (SubKey) ایجاد می‌کند.

در انتها به تعداد دور انجام شده، برنامه اجرا شده و نتیجه‌ی نهایی به همراه جزئیات آن یعنی زیرکلیدهای تولید شده و خروجی رمزنگاری در هر دور در خروجی نمایش داده می‌شود. برای تولید زیرکلیدها به اندازه‌ی شماره‌ی دور یک شیفت ساده داده می‌شود. بیشتر کد هم مستندسازی شده و در بیشتر خط‌ها توضیح داده شده که در حال انجام چه کاری هستیم و چه کاری باید انجام شود.

۵.۰ قابلیت انتقال این برنامه

به دلیل اینکه این برنامه بر روی بسترهای python و GTK+ نوشته شده است، قابلیت انتقال به هر سیستم‌عاملی که بتواند این پیش‌نیازها را تامین کند، دارد. برای مثال این برنامه بر روی سیستم‌عامل ویندوز ۷ به صورت زیر قابل اجراست :





برای نصب آن در ویندوز کافی است نسخه‌ی ویندوزی python 2.7 و همچنین پکیج کامل pygtk all in one را از سایت <http://www.pygtk.org> برای پایتون ۲,۷ بگیرید.

پایان
با تشکر
بهرام بهرام‌بیگی
bahramwhh@gmail.com