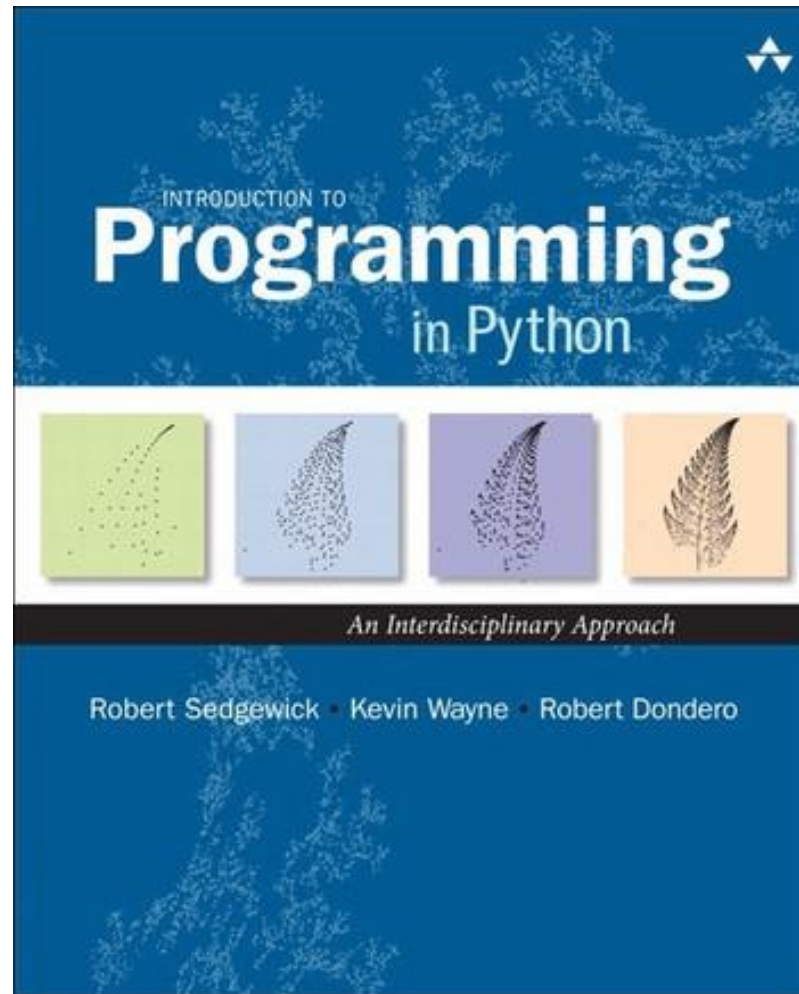


توابع و کتابخانه‌ها: کتابخانه‌ها

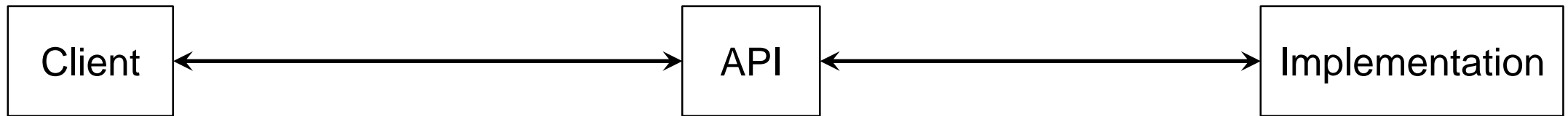
سید ناصر رضوی www.snrazavi.ir

۱۳۹۸



برنامه‌نویسی ماژولار (پیمان‌های)

۳



برنامه مشتری

برنامه‌ای که توابع موجود در کتابخانه را فراخوانی می‌کند

```
y[i] = gaussian.pdf(x[i])
```

Gaussian_plot.py

واسط برنامه‌نویسی (API)

تعریف امضای توابع، توصیف عملکرد توابع قرارداد میان برنامه مشتری و پیاده‌سازی

فراخوانی تابع

```
gaussian.pdf(x, mu, sigma)
```

```
gaussian.cdf(x, mu, sigma)
```

توصیف

تابع چگالی احتمال گوسی

تابع توزیع تجمعی گوسی

پیاده‌سازی

برنامه‌ای که توابع موجود در API را پیاده‌سازی می‌کند

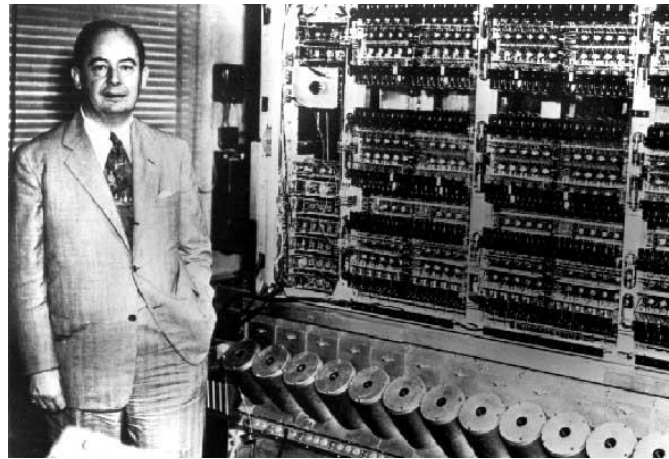
```
def pdf(x, mu, sigma):
```

```
def cdf(x, mu, sigma):
```

gaussian.py

اعداد تصادفی

« تولید اعداد تصادفی بسیار مهم‌تر از آن است که به شانس واگذار شود.
هر کسی که تلاش کند اعداد تصادفی را با استفاده از ابزارهای قطعی تولید
کند، البته که گناه‌کار است. »



جان ون نیومن و کامپیوتر انیاک

کتابخانه استاندارد تولید اعداد تصادفی: واسط

□ کتابخانه `stdrandom`. یک کتابخانه استاندارد برای تولید اعداد شبه تصادفی.

فراخوانی تابع	توصیف
<code>uniformInt(lo, hi)</code>	اعداد صحیح از توزیع یکنواخت در بازه $[lo, hi)$
<code>uniformFloat(lo, hi)</code>	اعداد حقیقی از توزیع یکنواخت در بازه $[lo, hi)$
<code>bernoulli(p)</code>	<code>true</code> با احتمال p (مقدار پیش فرض 0.5)
<code>binomial(n, p)</code>	تعداد شیرها در n پرتاب سکه (احتمال شیر در هر پرتاب برابر با p)
<code>gaussian(mu, sigma)</code>	توزیع نرمال، میانگین mu و انحراف معیار $sigma$
<code>discrete(a)</code>	i با احتمالی متناسب با $a[i]$
<code>shuffle(a)</code>	بر زدن آرایه $a[]$ به صورت تصادفی

کتابخانه استاندارد تولید اعداد تصادفی: پیاده‌سازی

۶

```
import random
```

```
def uniformInt(lo, hi):  
    return random.randrange(lo, hi)
```

```
def uniformFloat(lo, hi):  
    return random.uniform(lo, hi)
```

```
def bernoulli(p=0.5):  
    return random.random() < p
```

```
def binomial(n, p=0.5):  
    heads = 0  
    for i in range(n):  
        if bernoulli(p):  
            heads += 1  
    return heads
```

```
...
```

آزمایش واحد

۷

□ آزمایش واحد. قرار دادن متد main() به منظور آزمایش هر کتابخانه به صورت جداگانه.

```
def main():  
    import sys  
    n = int(sys.argv[1])  
    for i in range(n):  
        print(' %2d ' % uniformInt(10, 100), end='')  
        print('%8.5f' % uniformFloat(10.0, 99.0), end='')  
        print('%5s ' % bernoulli(), end='')  
        print('%7.5f ' % gaussian(9.0, 0.2), end='')  
        print('%2d ' % discrete([.5, .3, .1, .1]), end='')  
        print()  
  
if __name__ == '__main__':  
    main()
```

```
% python stdrandom.py 5  
61 21.76541 true 9.30910 0  
57 43.64327 false 9.42369 0  
31 30.86201 true 9.06366 0  
92 39.59314 true 9.00896 0  
36 28.27256 false 8.66800 1
```

استفاده از کتابخانه

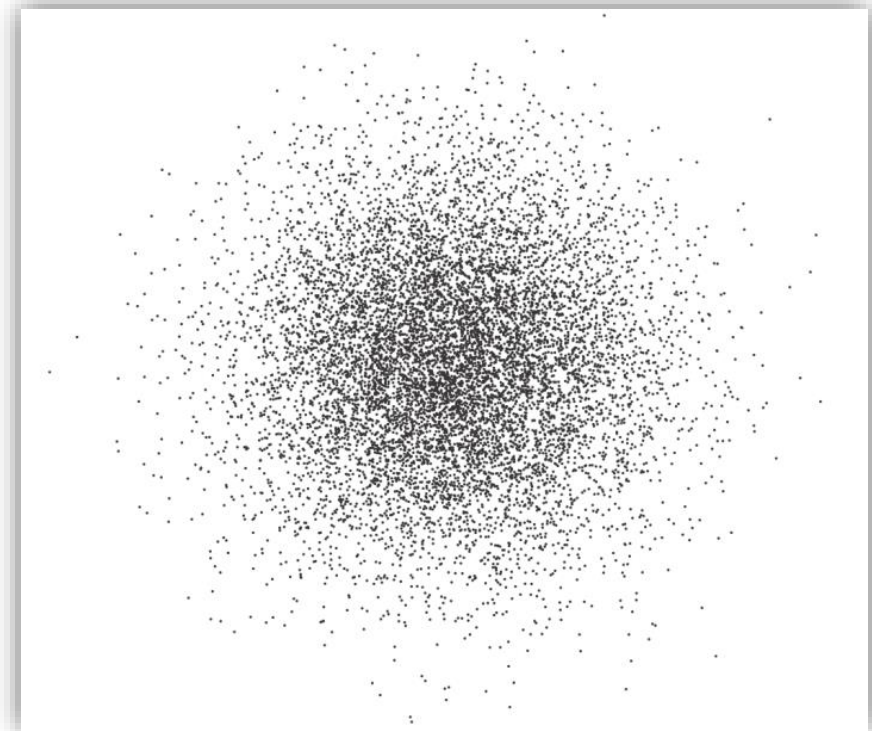
۸

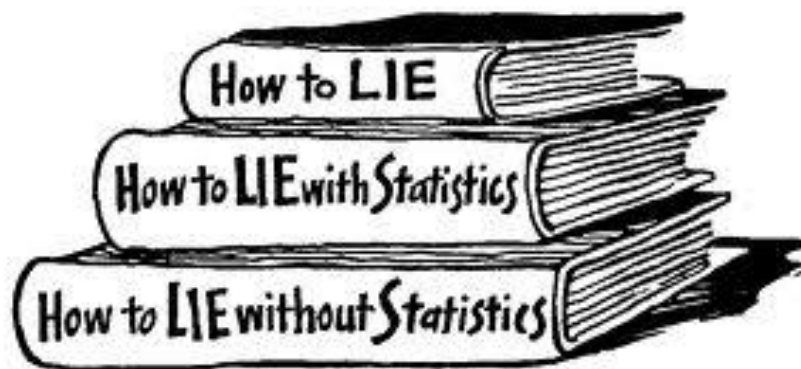
```
import sys
import stdrandom
import stddraw

def main():
    n = int(sys.argv[1])
    for i in range(n):
        x = stdrandom.gaussian(0.5, 0.2)
        y = stdrandom.gaussian(0.5, 0.2)
        stddraw.point(x, y)
    stddraw.show()

if __name__ == '__main__':
    main()
```

```
% python random_points.py 10000
```



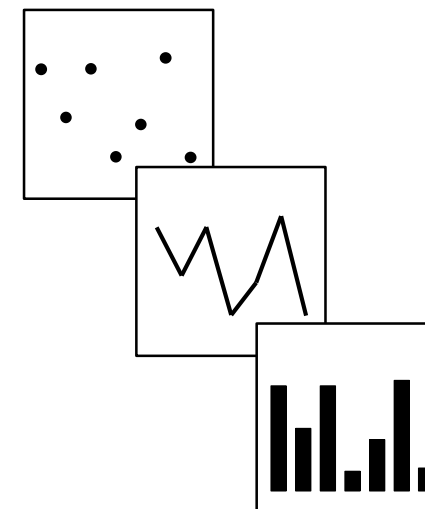


کتابخانه استاندارد آمار

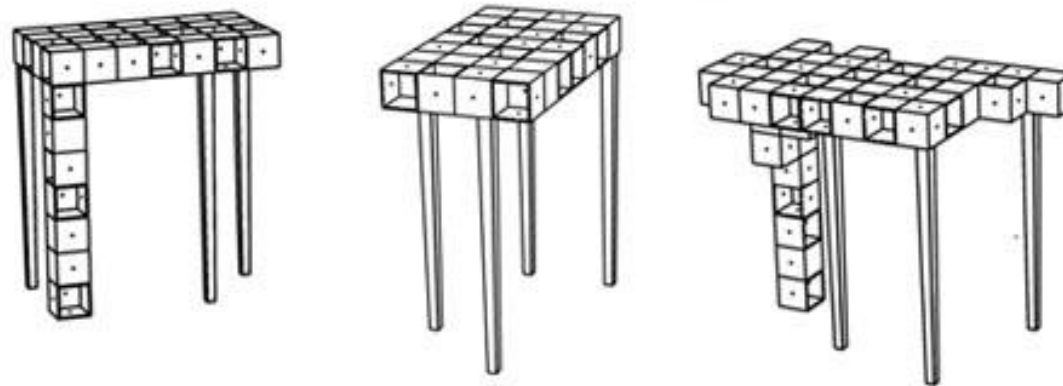
۱۰

□ کتابخانه `stdstats`. یک کتابخانه استاندارد برای انجام عملیات آماری بر روی آرایه‌های از اعداد حقیقی.

فراخوانی تابع	توصیف
<code>mean(a)</code>	میانگین مقادیر در آرایه عددی <code>a[]</code>
<code>var(a)</code>	واریانس نمونه مقادیر در آرایه عددی <code>a[]</code>
<code>stddev(a)</code>	انحراف معیار نمونه مقادیر در آرایه عددی <code>a[]</code>
<code>median(a)</code>	میانه مقادیر در آرایه عددی <code>a[]</code>
<code>plotPoints(a)</code>	ترسیم یک نمودار نقطه‌ای از مقادیر آرایه عددی <code>a[]</code>
<code>plotLines(a)</code>	ترسیم یک نمودار خطی از مقادیر آرایه عددی <code>a[]</code>
<code>plotBars(a)</code>	ترسیم یک نمودار میله‌ای از مقادیر آرایه عددی <code>a[]</code>



برنامه‌نویسی پیمان‌های



برنامه‌نویسی پیمان‌های

□ برنامه‌نویسی پیمان‌های.

□ تقسیم برنامه به بخش‌های مستقل.

□ آزمایش هر بخش به صورت جداگانه.

□ ترکیب بخش‌ها برای ایجاد برنامه.

□ مثال. شمارش تعداد شیرها در پرتاب n سکه.

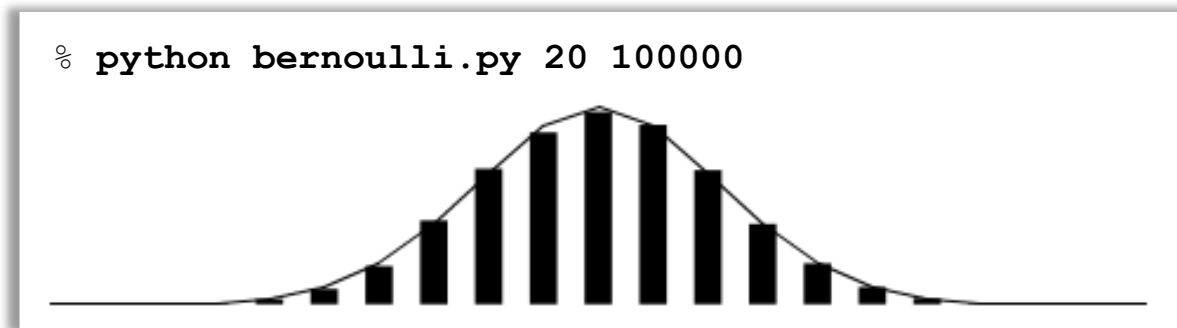
□ گرفتن مقادیر ورودی از کاربر.

□ پرتاب یک سکه.

□ پرتاب n سکه و شمارش تعداد شیرها.

□ ترسیم هیستوگرام مربوط به نتایج.

□ مقایسه با پیش‌بینی‌های نظری



آزمایش برنولی

۱۳

```
import sys, math
import stddraw, stdrandom, stdstats
import gaussian

n = int(sys.argv[1])
trials = int(sys.argv[2])

freq = [0] * (n + 1)
for t in range(trials):
    heads = stdrandom.binomial(n, 0.5)
    freq[heads] += 1

norm = [0.0] * (n + 1)
for i in range(n + 1):
    norm[i] = 1.0 * freq[i] / trials

phi = [0.0] * (n + 1)
stddev = math.sqrt(n) / 2.0
for i in range(n + 1):
    phi[i] = gaussian.pdf(i, n / 2.0, stddev)

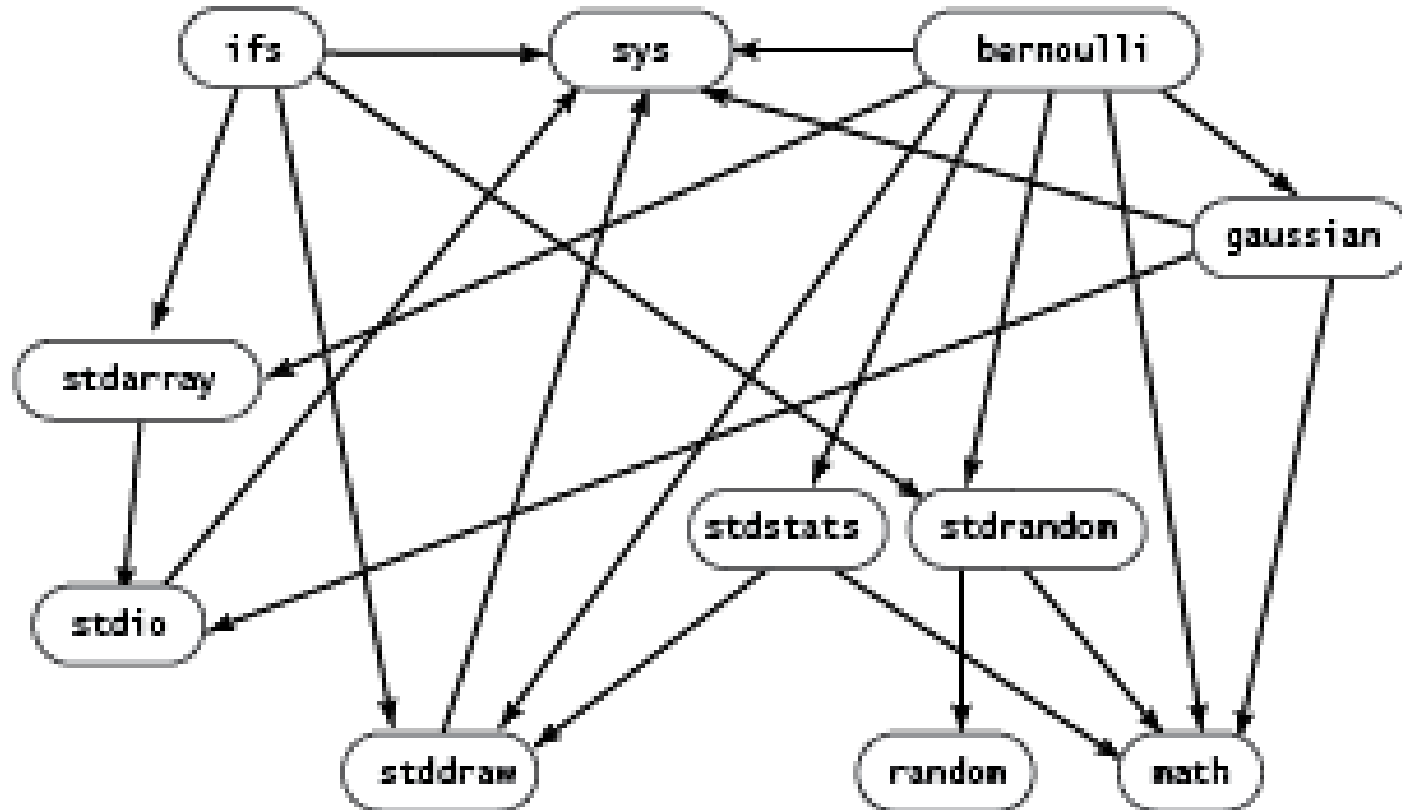
stddraw.setCanvasSize(1000, 400)
stddraw.setYscale(0, 1.1 * max(max(norm), max(phi)))
stdstats.plotBars(norm)
stdstats.plotLines(phi)
stddraw.show()
```

```
% python bernoulli.py 20 100000
```



گراف وابستگی

□ برنامه‌نویسی پیمان‌های. ایجاد برنامه‌های نسبتاً پیچیده با ترکیب چندین بخش کوچک و مجزا به نام ماجول (پیمان‌ه).



□ دلایل استفاده از کتابخانه‌ها.

□ آسان‌تر کردن درک کد.

□ آسان‌تر کردن فرایند اشکال‌زدایی.

□ آسان‌تر کردن فرایند نگهداری و بهبود.

□ آسان‌تر کردن استفاده مجدد از کد.