《计算复杂性理论》
第13讲 Fibonacci数

段会川
山东师范大学信息科学与工程学院

2014年12月

Fibonacci数—目录

- 算法与数学
  - 理想兔模型
  - 达芬奇密码
  - 递归算法
  - 动态规划算法
  - 幂函数—Θ(logn)算法
  - 指数表示及Θ(logn)算法
  - 通项公式—矩阵解
  - 通项公式——一般解
  - 指数式增长
- 普适性
  - 黄金分割
  - 杨辉三角
  - 黄金螺旋
  - 自然的法则
  - 建筑与艺术设计
  - 光伏阵列设计
  - 五角星

莱昂纳多·斐波那契(Leonardo Fibonacci)

公元1170 – 1250

意大利数学家；
“中世纪(公元5-15世纪)最有才华的数学家”；
他1202年的著作Liber Abaci(计算之书)向欧洲传播了印度—阿拉伯的十进制计数系统；
“斐波那契数列”以他的名字命名，尽管该数列已被6世纪的印度数学家知晓，但斐波那契是将其引入到欧洲的人。

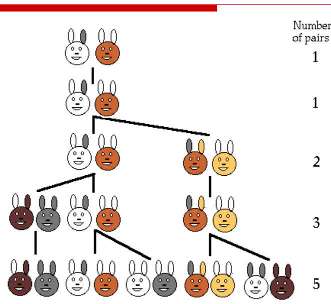
Fibonacci的理想兔模型

- 模型表述
  - 某年1月初有雌雄一对兔，2月初成婚，3月初产一对雌雄仔，以后每一月初均产一对雌雄仔
  - 任何一对雌雄仔均在出生后的第二个月成婚，其后每月产一对雌雄仔
  - 所有兔子均长命不老
- 数学表达

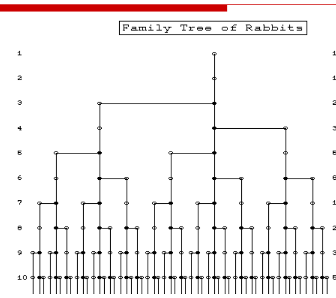
$$F_n = \begin{cases} 1 & n=1 \\ 1 & n=2 \\ F_{n-1}+F_{n-2} & n>2 \end{cases}$$

- 前20个数
  - 1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765

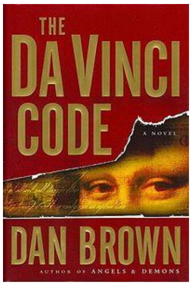
Fibonacci理想兔

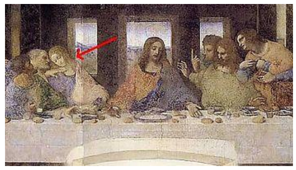


Fibonacci理想兔



小说：达芬奇密码Da Vinci Code (2003)





The Last Supper by Leonardo da Vinci

美国作家丹·布朗(Dan Brown)的一部小说，2003年3月18日由兰登书屋(Random House)出版

电影：达芬奇密码Da Vinci Code (2006)

导演： 朗·霍华德
主演： 汤姆·汉克斯
奥黛丽·塔图
伊安·麦克莱恩
首映： 2006.5.19, 美国

1, 1, 2, 3, 5, 8, 13, …

电影：达芬奇密码Da Vinci Code (2006)

Fibonacci数求解—递归算法

```
function FibR(n)
  if n=1 then f = 1
  else if n=2 then f = 1
  else
    f = FibR(n-1)+FibR(n-2)
  return f
end
```

- 递归
  - recursive, recursion
- 调用FibR次数
  - n=1: FibR(1), 共1次
  - n=2: FibR(2), 共1次
  - n=3: FibR(3) (FibR(1), FibR(2)), 共3次
  - n=4: FibR(4) (FibR(3), FibR(2)), 共5次
  - n=5: FibR(5) (FibR(4), FibR(3)), 共9次

Fibonacci数求解—递归算法

- 调用次数公式 $T(n)=\begin{cases}1 & n=1\\ 1 & n=2\\ T(n-1)+T(n-2)+1 & n>2\end{cases}$
- 调用次数公式计算
  - 1, 1, 3, 5, 9, 15, 24, …
- 对比Fibonacci数
  - 1, 1, 2, 3, 5, 8, 13
- 递归算法的运行次数是一个比Fibonacci数还大的数，即Ω(Fn)，是随n指数式增加的！

Fibonacci数求解—动态规划算法

```
function FibDP(n)
  for i=1 to n
    if n=1 then f[1] = 1
    else if n=2 then f[2] = 1
    else
      f[i] = f[i-1]+f[i-2]
  endfor
  return f[n]
end
```

- 动态规划
  - Dynamic Programming
  - 保存较小规模的计算结果，使较大规模的计算直接使用较小规模的结果，免去重新计算
- 技巧：使用数组
- 运算次数：Θ(n)，线性的！
- 代价：数组需要耗费存储空间，对巨大的n将因存储空间耗尽而不可行

Fibonacci数求解—简化的动态规划算法

```
function FibDP1(n)
  if n=1 then f=1
  elseif n=2 then f=1
  else
    f1 = 1;   f2 = 1
    for i=3 to n
      f = f1+f2
      f2 = f1; f1 = f
    endfor
  endif
  return f
end
```

- 技巧：使用两个变量只保存前两个值
- 运算次数：Θ(n)，线性的！
- 完全解决了吗？
  - No：对于巨大的n，Fn将变得超出计算机的整数范围，需要新的方法保存数据，也会带来新的运算开销

幂函数—Θ(n)算法

```
//计算x^n, Θ(n)算法
function Power(x,n)
  y = 1;
  for i=1 to n
    y = y*x
  endfor
  return y
end
```

- 算法循环n次，线性复杂度Θ(n)

幂函数—Θ(logn)算法

```
//计算x^n , Θ(logn)算法
function PLogN(x,n)
  if n = 1 then y=x
  else
    y0 = PLogN(x, n/2)
    y = y0*y0
    if n is odd then y=y*x
  endif
  return y
end
```

- 原理
  - x^8
    - $x^8 = x^4 \cdot x^4$
    - $x^4 = x^2 \cdot x^2$
    - $x^2 = x \cdot x$
    - 3次乘法！
  - x^{11}
    - $x^{11} = x \cdot x^5 \cdot x^5$
    - $x^5 = x \cdot x^2 \cdot x^2$
    - $x^2 = x \cdot x$
    - 5次乘法！

幂函数—Θ(logn)算法

```
//计算x^n , Θ(logn)算法
function PLogN(x,n)
  if n = 1 then y=x
  else
    y0 = PLogN(x, n/2)
    y = y0*y0
    if n is odd then y=y*x
  endif
  return y
end
```

- 算法递归调用PLogN共logn次
  - n>1时，PLogN执行1次y0*y0，0次或1次y*x
  - 最少的乘法次数是logn次
  - 最多的乘法次数是2(logn)次
  - 具体地说
    - 当n=1024时，执行10次乘法
    - 1024<n<2047时执行11-20次乘法
    - 当n=2048时，执行11次乘法
  - 即算法复杂度为Θ(logn)！

Fibonacci数求解—指数运算公式

- 以矩阵表达Fibonacci数

$$\begin{cases} F_{n+1} = 1\cdot F_n + 1\cdot F_{n-1} \\ F_n = 1\cdot F_n + 0\cdot F_{n-1} \end{cases} \Rightarrow \begin{pmatrix} F_{n+1} \\ F_n \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_n \\ F_{n-1} \end{pmatrix}$$

- 显然这也是一个递推式，因而可有

$$\begin{pmatrix} F_n \\ F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_{n-1} \\ F_{n-2} \end{pmatrix}$$

- 将它们合并起来，有

$$\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} F_n & F_{n-1} \\ F_{n-1} & F_{n-2} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^2 \begin{pmatrix} F_{n-1} & F_{n-2} \\ F_{n-2} & F_{n-3} \end{pmatrix}$$

Fibonacci数求解—指数运算公式

- 设F_0=0，则Fibonacci数可以从0开始，并且有

$$F_n = \begin{cases} 0 & n=0 \\ 1 & n=1 \\ F_{n-1}+F_{n-2} & n \ge 2 \end{cases}$$

- 因此，我们有

$$\begin{pmatrix} F_2 & F_1 \\ F_1 & F_0 \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$

- 最后可得关于Fibonacci数的幂表达式

$$\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n$$

Fibonacci数求解—指数运算公式

- 采用与幂函数相同的方式，则有

$$\begin{pmatrix} F_{12} & F_{11} \\ F_{11} & F_{10} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{11}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{11} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{5} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{5}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{5} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{2} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{2}$$

$$\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^{2} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$$

Fibonacci数求解—指数运算公式

- 设计算法时，需要下列式子
  - 假设已经算得矩阵 $\begin{pmatrix} F_a & F_b \\ F_b & F_c \end{pmatrix}$
  - 则 $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} F_a & F_b \\ F_b & F_c \end{pmatrix} = \begin{pmatrix} F_a + F_b & F_b + F_c \\ F_a & F_b \end{pmatrix} = \begin{pmatrix} F_a + F_b & F_a \\ F_a & F_b \end{pmatrix}$

$$\begin{pmatrix} F_a & F_b \\ F_b & F_c \end{pmatrix}\begin{pmatrix} F_a & F_b \\ F_b & F_c \end{pmatrix} = \begin{pmatrix} F_a \cdot F_a + F_b \cdot F_b & F_a \cdot F_b + F_b \cdot F_c \\ F_a \cdot F_b + F_b \cdot F_c & F_b \cdot F_b + F_c \cdot F_c \end{pmatrix}$$

Fibonacci数求解—Θ(logn)算法

```
function FLogN(n)
  if n = 0 then fa=0; fb=-1; fc=-1
  else if n=1 then
    fa=1; fb=0; fc=-1
  else if n=2 then
    fa=1; fb=1; fc=0
  else
    ga,gb,gc = FLogN((n-1)/2)
    fa,fb,fc = Mul2(ga,gb,gc)
    if n-1 is odd then
      fa,fb,fc = Mul1(fa,fb,fc)
    endif
  endif
  return fa,fb,fc
end
```

```
function Mul2(ga, gb, gc)
  ga2=ga*ga; gb2=gb*gb; gc2=gc*gc
  gab=ga*gb; gbc=gb*gc
  fa = ga2+gb2; fb=gab+gbc
  fc = gb2+gc2
  return fa, fb, fc
end

function Mul1(ga, gb, gc)
  fa = ga+gb;
  fb = ga; fc = gb
  return fa, fb, fc
end
```


Fibonacci数求解—Θ(logn)算法

- 算法递归调用FLogN共log(n-1)次
  - n>2时，FLogN执行1次Mul2，0次或1次Mul1
  - Mul2中包括5次乘法，3次加法
  - Mul1中包括1次加法
  - 算法的乘法次数是5(log(n-1)-1)次
  - 最少的加法次数是3(log(n-1)-1)次
  - 最多的加法次数是4(log(n-1)-1)次
  - 因此，算法复杂度为Θ(logn)!

Fibonacci数的数学解—矩阵特征值法

- $\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n$
- 令$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}$，则有特征方程$\begin{pmatrix} 1-\lambda & 1 \\ 1 & -\lambda \end{pmatrix} = 0$
- 即$(1-\lambda)(-\lambda) - 1 = 0$，$\lambda^2 - \lambda - 1 = 0, \lambda = \frac{1 \pm \sqrt{5}}{2}$
- 令$\varphi = \frac{1+\sqrt{5}}{2}, \psi = \frac{1-\sqrt{5}}{2}$
- 则特征向量
  - $\begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}\begin{pmatrix} a \\ b \end{pmatrix} = \varphi \begin{pmatrix} a \\ b \end{pmatrix} \Rightarrow \begin{cases} a + b = \varphi a \\ a = \varphi b \end{cases} \Rightarrow V_\varphi = \begin{pmatrix} \varphi \\ 1 \end{pmatrix} b$
  - 同理$V_\psi = \begin{pmatrix} \psi \\ 1 \end{pmatrix} b$

Fibonacci数的数学解—矩阵特征值法

- V_φ, V_ψ可以构成正交矩阵$Q = \begin{pmatrix} \varphi a & \psi b \\ a & b \end{pmatrix}$
- 且$Q^{-1} = \frac{adj(Q)}{\det(Q)} = \frac{1}{(\varphi-\psi)ab}\begin{pmatrix} b & -\psi b \\ -a & \varphi a \end{pmatrix}$
- 显然$M = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix} = Q \begin{pmatrix} \varphi & 0 \\ 0 & \psi \end{pmatrix} Q^{-1}$
- 因而$\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} 1 & 1 \\ 1 & 0 \end{pmatrix}^n = Q \begin{pmatrix} \varphi^n & 0 \\ 0 & \psi^n \end{pmatrix} Q^{-1}$
- $\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} \varphi^{n+1} a & \psi^{n+1} b \\ \varphi^n a & \psi^n b \end{pmatrix} Q^{-1}$

Fibonacci数的数学解—矩阵特征值法

- $Q^{-1} = \frac{1}{(\varphi-\psi)ab}\begin{pmatrix} b & -\psi b \\ -a & \varphi a \end{pmatrix}$
- $\begin{pmatrix} F_{n+1} & F_n \\ F_n & F_{n-1} \end{pmatrix} = \begin{pmatrix} \varphi^{n+1}a & \psi^{n+1}b \\ \varphi^n a & \psi^n b \end{pmatrix} Q^{-1}$
- $F_n = \frac{-\varphi^{n+1}\psi ab+\psi^{n+1}\varphi ab}{(\varphi-\psi)ab} = \frac{\varphi^n-\psi^n}{\varphi-\psi}$
- $\varphi - \psi = \frac{1+\sqrt{5}}{2} - \frac{1-\sqrt{5}}{2} = \sqrt{5}$
- 因此, $F_n = \frac{1}{\sqrt{5}}\varphi^n - \frac{1}{\sqrt{5}}\psi^n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$

Fibonacci数的数学解—通项公式

- 对于常系数的线性齐次递推式
  $a_n = c_1 a_{n-1} + c_2 a_{n-2} + \cdots c_d a_{n-d}$
- 存在一个系数相同的特征多项式
  $p(t) = t^d - c_1 t^{d-1} - c_2 t^{d-2} - \cdots - c_d$
- 若该特征多项式有d个不相等的根$r_1, r_2, \ldots, r_d$，则可以构造递推式的闭式解
  $a_n = k_1 r_1^n + k_2 r_2^n + \cdots + k_d r_d^n$
  - 其中，系数$k_1, k_2, \ldots, k_d$可以根据初条件得出

Fibonacci数的数学解—通项公式

- Fibonacci数列的递推式为$F_n = F_{n-1} + F_{n-2}$
- 其特征多项式为$p(t) = t^2 - t - 1$
- 该特征多项式的2个根为$\varphi = \frac{1+\sqrt{5}}{2}, \psi = \frac{1-\sqrt{5}}{2}$
- 则Fibonacci数列的通项为$F_n = k_1\varphi^n + k_2\psi^n$
- 由F_0=0, F_1=1得$\begin{cases} k_1 + k_2 = 0 \\ \frac{1+\sqrt{5}}{2}k_1 + \frac{1-\sqrt{5}}{2}k_2 = 1 \end{cases} \Rightarrow k_1 = \frac{1}{\sqrt{5}}, k_2 = -\frac{1}{\sqrt{5}}$
- 由此得Fibonacci数列的通项公式$F_n = \frac{1}{\sqrt{5}}\left(\frac{1+\sqrt{5}}{2}\right)^n - \frac{1}{\sqrt{5}}\left(\frac{1-\sqrt{5}}{2}\right)^n$
- 注意：
  - Fibonacci数列的通项公式对于计算Fibonacci数帮助不大，因为开平方需要复杂的算法，而且用通项公式进行数值计算无法得到整数解。

Fibonacci数—指数式增长

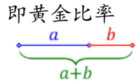
- 显然，$\varphi > 1, -1 < \psi < 0$
- 当n→∞时，$F_n \to 0.447 \times 2^{0.694n} \approx 2^{0.694n-1}$
- 所以Fn是指数式增长的
- 令0.694n-1=32，得n≈47.55，即第48个Fibonacci数就超过了32位整数的容量
- 令0.694n-1=64，得n≈93.65，即第94个Fibonacci数就超过了64位整数的容量
- 令0.694n-1=50/lg2，得n≈237.91，即第238个Fibonacci数就超过了50位十进制数
- 令0.694n-1=100/lg2，得n≈477.27，即第478个Fibonacci数就超过了100位十进制数

Fibonacci数—目录

- 算法与数学
  - 理想兔模型
  - 达芬奇密码
  - 递归算法
  - 动态规划算法
  - 幂函数—Θ(logn)算法
  - 指数运算公式
  - Θ(logn)算法
  - 通项公式
  - 指数式增长
- 普适性
  - 黄金分割
  - 杨辉三角
  - 黄金螺旋
  - 自然的法则
  - 建筑与艺术设计
  - 光伏阵列设计
  - 五角星

Fibonacci数—黄金分割

- 所谓的黄金分割(golden section)是指将线段分成a,b两部分，较长的一段a与较短的一段b的比值等于整个线段a+b与较长的一段a的比值，即黄金比率
  $\frac{a}{b} = \frac{a+b}{a} \equiv \varphi$



- 显然φ满足方程：$\varphi^2 - \varphi - 1 = 0$
- 其正数解为：$\varphi = \frac{1+\sqrt{5}}{2} \approx 1.6180339887\cdots$

Fibonacci数—黄金分割

- 可以看出，Fibonacci数的特征方程正是黄金分割满足的方程，根据韦达定理，我们有

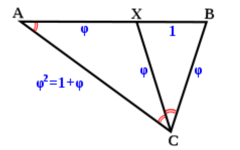
$$\varphi + \psi = 1, \varphi \cdot \psi = -1$$

- 相邻Fibonacci数的渐近比为黄金分割 $\lim_{n\to\infty} \frac{F_{n+1}}{F_n} = \varphi$
- 而相邻Fibonacci数的比值是黄金分割最接近的有理数比值：2/1, 3/2, 5/3, 8/5, …
- 黄金分割方程决定了φ很有趣的数值特点

$$\frac{1}{\varphi} = \varphi - 1 \Rightarrow \frac{1}{1.6180339887\cdots} = 0.6180339887\cdots = 1.6180339887\cdots - 1$$

Fibonacci数—黄金三角形

- 黄金三角形指的是如下性质的等腰三角形
  - 平分其底角得到一个与原三角形相似的三角形

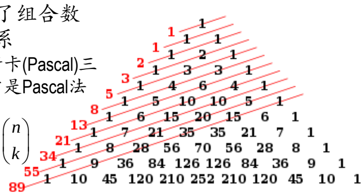


$$\angle BCX = \angle ACX = \angle A$$
$$5\angle A = 180° \Rightarrow \angle A = 36°$$
$$\frac{BC}{BX} = \frac{AB}{BC}$$
$$BC = CX = AX$$
$$\frac{AX}{BX} = \frac{AB}{AX} = \varphi$$

Fibonacci数—杨辉三角

- 杨辉三角表达了组合数的一种奇特关系
  - 西方称为帕斯卡(Pascal)三角形，因为它是Pascal法则的体现

$$\binom{n-1}{k} + \binom{n-1}{k-1} = \binom{n}{k}$$

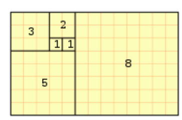


- 杨辉三角各"浅"斜线上数的和正好是Fibonacci数 $F_n = \sum_{k=0}^{\lfloor \frac{n-1}{2} \rfloor} \binom{n-k-1}{k}$

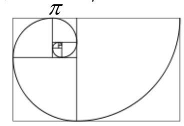
Fibonacci数—黄金螺旋

- 黄金螺旋(golden spiral)指的是增长因子为黄金比率φ的螺旋线，即当角度变化一个直角时，半径扩大到黄金比率φ，

$$r = ae^{b\theta}, e^{b\frac{\pi}{2}} = \varphi, b = \frac{2}{\pi}\ln\varphi \approx 0.306349\cdots$$



使用边长为Fibonacci数的正方形可以长方形方式平铺平面

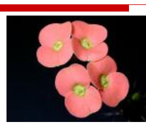


Fibonacci平铺中依次连接各正方形的相对顶点的圆弧是黄金螺旋非常好的近似

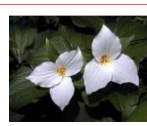
维基百科：http://www.wikipedia.org

Fibonacci数—自然的法则？

马蹄莲有1个花瓣



大戟属植物有2个花瓣



延龄草有3个花瓣

耧斗菜有5个花瓣

罂粟科植物有8个花瓣

多毛金光菊有13个花瓣

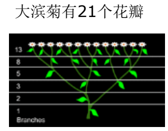
维基百科：http://www.wikipedia.org

Fibonacci数—自然的法则？

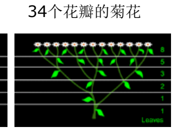
大滨菊有21个花瓣

34个花瓣的菊花

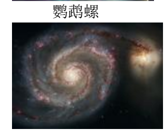
鹦鹉螺



珠蓍的分叉



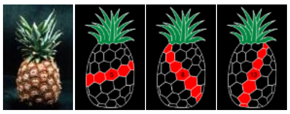
珠蓍的叶片



螺旋星云

维基百科：http://www.wikipedia.org

Fibonacci数—自然的法则？



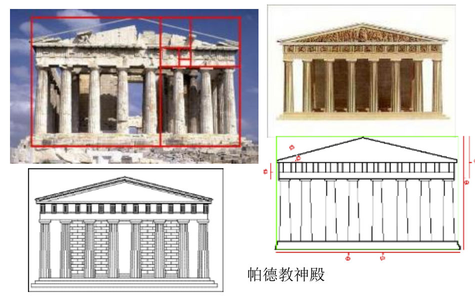
菠萝上的Fibonacci数

金花菊　向日葵仔的排列　向日葵叶子的排列

叶子绕茎的排列

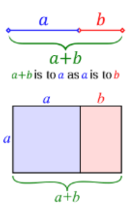
维基百科：http://www.wikipedia.org

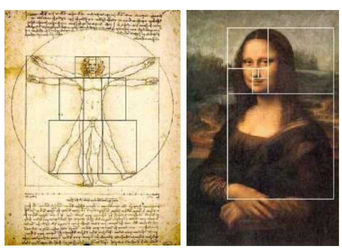
Fibonacci数—建筑设计



帕德教神殿

Fibonacci数—艺术设计





达·芬奇的著名设计

Fibonacci数—光伏阵列设计

Fibonacci数—光伏阵列设计

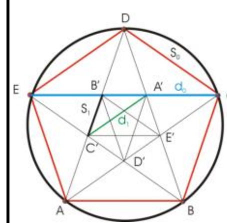
The Secret of the Fibonacci Sequence in Trees

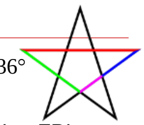
Aidan Dwyer；
美国长岛，
13岁，小学7年级；
获2011年美国自然历史博物馆7-12年级段少年博物学家奖，获奖作品是基于植物叶子沿茎排列的Fibonacci螺旋形态是为了获得充分的光照量的原理设计的光伏阵列

Fibonacci数—红星与希望

Fibonacci数—国家象征的设计

Fibonacci数—五角星





$$\angle ADB = \angle DEC = 36^\circ$$

$$\Delta EA'D \sim \Delta DA'B'$$

$$\frac{EA'}{DB'} = \frac{DB'}{B'A'} \Rightarrow \frac{EA'}{EB'} = \frac{EB'}{B'A'} = \varphi$$

$$\frac{EB'}{EA'} + 1 = \frac{B'A'}{EB'} + 1 \Rightarrow \frac{EC}{EA'} = \frac{EA'}{EB'} = \varphi$$

B'点既是EA'的黄金分割点，也是EC的黄金分割点

五角星丰富地体现了黄金分割

Fibonacci数—总结

- 算法与数学
  - 理想兔模型
  - 达芬奇密码
  - 递归算法
  - 动态规划算法
  - 幂函数—Θ(logn)算法
  - 指数运算公式
  - Θ(logn)算法
  - 通项公式
  - 指数式增长
- 普适性
  - 黄金分割
  - 杨辉三角
  - 黄金螺旋
  - 自然的法则
  - 建筑设计
  - 艺术设计
  - 光伏阵列设计
  - 国家象征的设计
  - 五角星

The End

Thanks