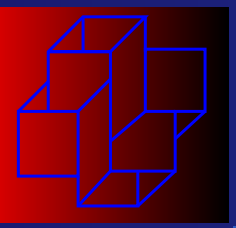
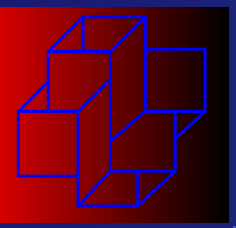


Um novo algoritmo probabilístico para fatoração de inteiros com primos relativamente distantes

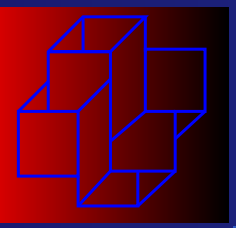
VIII SBSeg - Set/08

Fábio Borges - LNCC



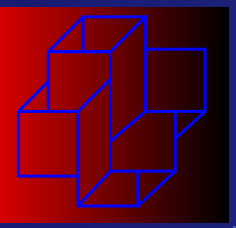
Introdução

- Problema da Fatoração de Inteiros
  - $n = p_1^{e_1} \cdots p_i^{e_i}$
  - $n = pq$
- Complexidade subexponencial
  - Computação Clássica
- Base da segurança de muitos sistemas computacionais
- Depende da escolha de p e q

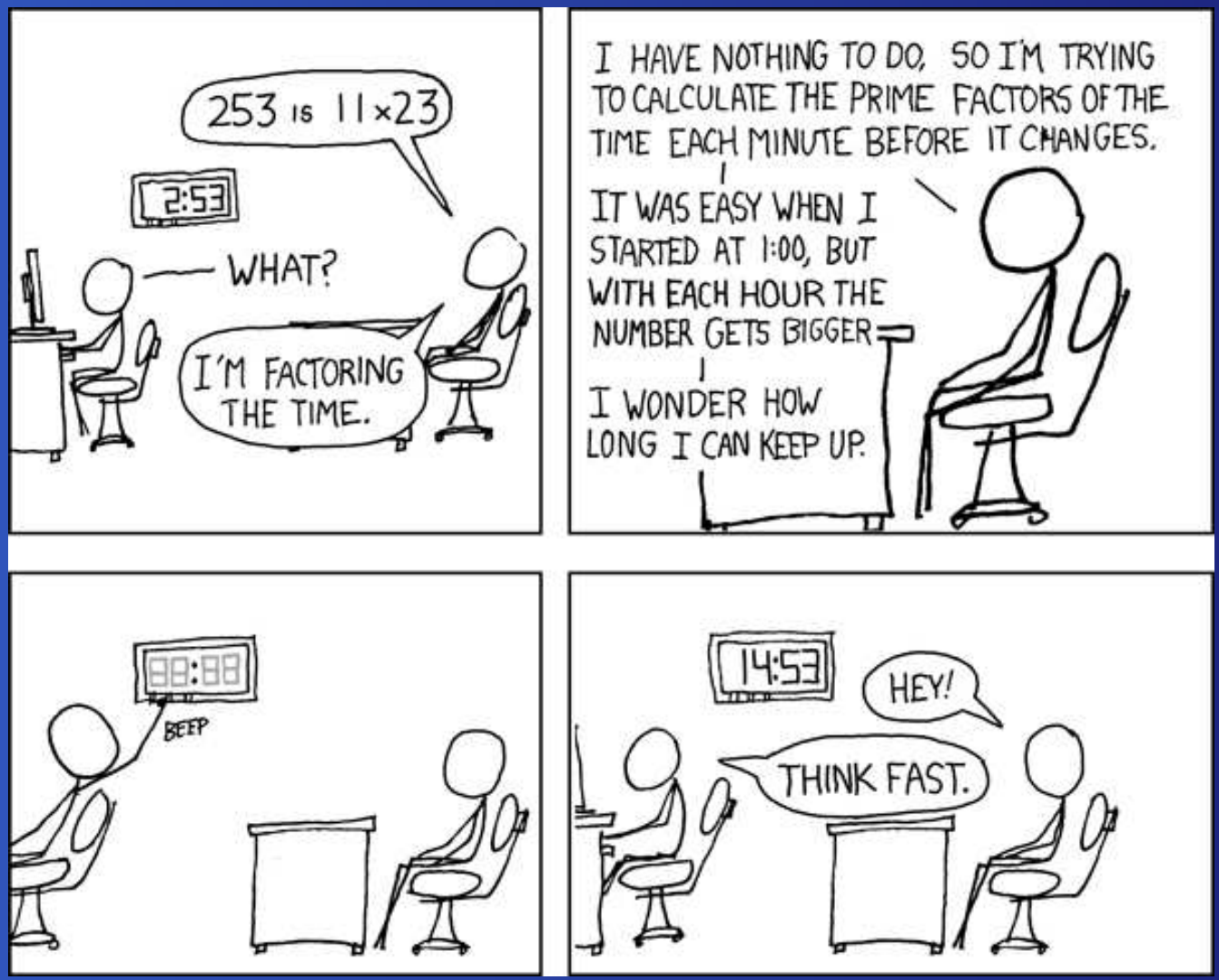


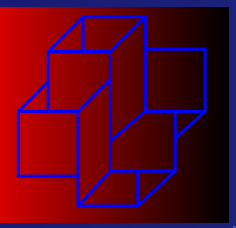
Trabalhos Relacionados

- método de Fermat
- $p_i - 1$ de Pollard
- método de Lenstra
- *multiple polynomial quadratic sieve* (MPQS)
- *general number field sieve* (GNFS)



PhD Comics


253 is 11×23
2:53
WHAT?
I'M FACTORING THE TIME.
I HAVE NOTHING TO DO, SO I'M TRYING TO CALCULATE THE PRIME FACTORS OF THE TIME EACH MINUTE BEFORE IT CHANGES.
IT WAS EASY WHEN I STARTED AT 1:00, BUT WITH EACH HOUR THE NUMBER GETS BIGGER
I WONDER HOW LONG I CAN KEEP UP.
BEEP
14:53
HEY!
THINK FAST.



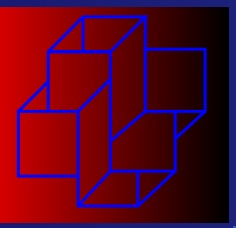
Idéia Central

A idéia central do algoritmo é encontrar m tal que

$$\text{mdc}\ (pq \mod m, pq) > 1$$

Observe que

$$pq \mod m \neq ((p \mod m) \cdot (q \mod m)) \mod m$$

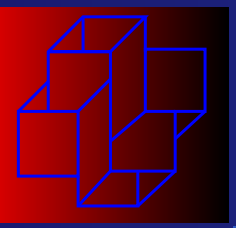


Princípio

$$pq \mod q-1 = p(q-1)+p \mod q-1 = p$$

$$p(q-s)+sp \mod q-s = sp$$

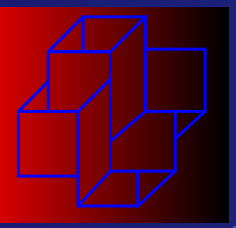
sempre que $sp < q-s$



Exemplo

$973 = 7 \cdot 139$ tem:

- 17 soluções consecutivas entre 121 e 139
- 166 soluções no total



Fatorando

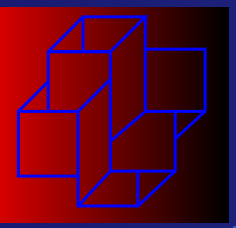
Podemos fatorar procurando um número

$$m = q - s$$

tal que

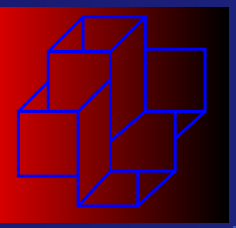
$$\text{mdc}\ (pq \mod m, pq) > 1$$

se $m = 139$ então $\text{mdc}\ (0, 7 \cdot 139) = 973$, mas se $m = 138$ então $\text{mdc}\ (7 \cdot 139 \mod 138, 7 \cdot 139) = 7$.



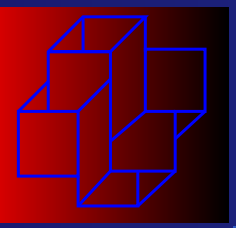
Encontrando fatores distantes

1. while true do
2. $m = \mathrm{PRNG}()$
3. $v = n \mod m$;
4. if $(\mathrm{mdc}\ (v, n) > 1)$ then
5. Return$(\mathrm{mdc}\ (v, n))$;



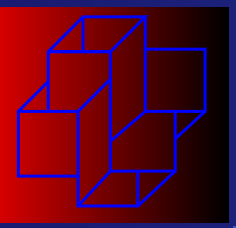
Encontrando fatores distantes

1. $i = 1$; $v = 1$; $r = [\sqrt{n}]$;
2. while $\mathrm{mdc}\ (v, n) = 1$ and $m < 3r$ do
3. $\quad i = \mathrm{nextprime}(i)$;
4. $\quad$ for m from r to $3r$ by $[3r/i]$ do
5. $\quad\quad v = n \mod m$;
6. $\quad\quad$ if $(\mathrm{mdc}\ (v, n) > 1)$ then
7. $\quad\quad\quad$ Return$(\mathrm{mdc}\ (v, n))$;



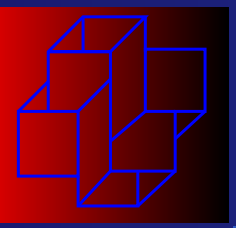
Aumentando a probabilidade

- $p < q$



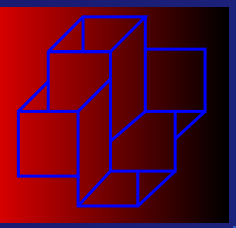
Aumentando a probabilidade

- $p < q$
- $\sqrt{pq} < q < pq$ único divisor



Aumentando a probabilidade

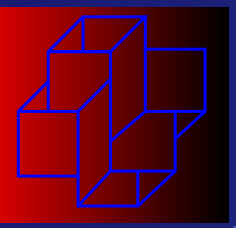
- $p < q$
- $\sqrt{pq} < q < pq$ único divisor
- $\sqrt{pq} < \text{mdc}\,(X_1, pq) < \cdots < \text{mdc}\,(X_i, pq) < pq$



Aumentando a probabilidade

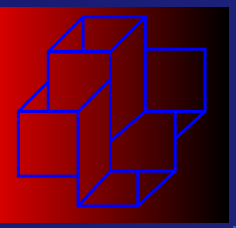
- $p < q$
- $\sqrt{pq} < q < pq$ único divisor
- $\sqrt{pq} < \text{mdc}\,(X_1, pq) < \cdots < \text{mdc}\,(X_i, pq) < pq$
-

$$\begin{aligned} \sqrt{pq} &< \text{mdc}\,(pq \mod X_1, pq) \\ \cdots &< \cdots \\ &< \text{mdc}\,(pq \mod X_j, pq) < pq \end{aligned}$$



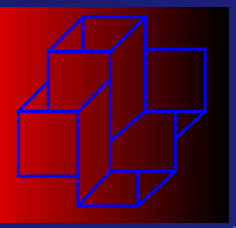
Aumentando a probabilidade

- $p < q$
- $\sqrt{pq} < q < pq$ único divisor
- $\sqrt{pq} < \text{mdc}\,(X_1, pq) < \cdots < \text{mdc}\,(X_i, pq) < pq$
- $$\begin{aligned} \sqrt{pq} &< \text{mdc}\,(pq \mod X_1, pq) \\ \cdots &< \cdots \\ &< \text{mdc}\,(pq \mod X_j, pq) < pq \end{aligned}$$
- $i < j$, pois $pq \mod pX = pq - kpX = pY$



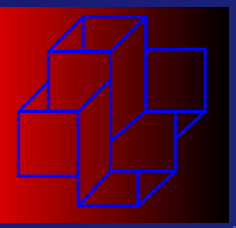
Teste

n	=	1 132 071 599 596 903 309
passos	=	149 095 241
p	=	997 845 647
q	=	1 134 515 747



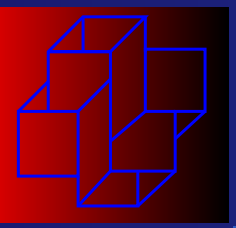
Relação entre s e d

- $p(s-1)+s<d$
- d cresce muito mais rápido que s
- $s<\frac{q}{p+1}$



Conclusão

Quanto maior for s, maior a distância d e melhor será o algoritmo.



Último Slide

- Obrigado.
- Quaisquer sugestões serão bem-vindas.

www.lncc.br/∼borges