

Implemente, ou use, um algoritmo qualquer disponível online (<https://www.alpertron.com.ar/ECM.HTM>) para verificar até que ponto a fatoração do parâmetro  $n$  do algoritmo RSA é possível. Use como entrada o parâmetro  $n$  da sua chave pública (exercício 1). Por exemplo, selecione os 10 primeiros dígitos de  $n$ , e tente fatorar. Selecione os 20 primeiros e assim por diante. Qual foi o último tamanho possível de fatoração? Qual a relação disso com a segurança do RSA?

### parâmetro $n$ da chave pública:

```
C:\Users\jotam\Documents\GitHub\Information-security\pratica\TP6>openssl rsa -pubin -in chave_publica.pem -text -noout
Public-Key: (2048 bit)
Modulus:
 00:c7:fd:09:e3:c0:89:56:a0:5d:bd:31:f4:42:02:
 2b:ae:a3:22:81:f4:3a:60:2a:b4:a4:59:c8:66:e4:
 e4:49:64:3d:3a:2a:89:1c:19:cf:63:e8:7f:ea:92:
 43:f7:5b:f7:1f:e6:3c:98:8d:bd:32:d8:f9:31:bb:
 fd:e5:26:da:cf:a9:a1:72:82:2f:84:8c:6d:9a:e3:
 44:34:44:ec:65:d7:63:a9:80:5f:de:b7:1b:c2:e6:
 4f:ab:18:a2:4f:a9:09:31:d9:5e:6c:39:2d:ca:74:
 88:bc:12:5d:03:49:83:c7:fd:9b:d2:55:d5:b1:95:
 1a:35:14:07:c6:72:fb:8e:5e:cb:d5:b4:1b:9c:ba:
 44:a2:49:6b:11:39:e3:0a:a1:45:49:5e:1d:46:b6:
 90:e8:65:7c:c3:c1:a4:84:c0:57:94:98:62:f0:57:
 29:a8:00:23:6f:02:ef:bd:3d:16:9d:77:33:ad:1d:
 01:97:cd:6f:d0:9a:94:bb:70:a0:4a:9b:f4:b0:d6:
 6f:96:ff:3f:c1:b9:b9:d6:bd:ca:87:ea:f9:7e:26:
 d2:43:4e:65:95:c8:61:cc:2c:29:da:0d:62:dc:9a:
 4f:13:16:21:f3:4e:ae:35:eb:90:88:b7:31:10:98:
 c0:59:df:82:8b:b0:58:82:05:71:76:c3:08:f0:67:
 86:15
Exponent: 65537 (0x10001)
```

Usei

<https://www.rapidtables.com/convert/number/hex-to-decimal.html?x=C7FD09E3C0>

para transformar Hexadecimal para Decimal para poder fatorar  
10 dígitos:

**c7fd09e3c0**

em decimal: 858943775680

← → ↻ 🌐 alpertron.com.ar/ECM.HTM

Alpertron > Web applications > Integer factorization calculator

Value

858943775680

Actions

Only evaluate

Is prime?

Factor

Help

Config

Open wizard

From file

Blockly mode

Clear input

Type one nu

Press the **Help** button to get help about this application. Press it again to return to the factorization. You c

- 858943 775680 = 2<sup>6</sup> × 5 × 37 × 2287 × 31721

Number of divisors: 112

Sum of divisors: 2 101624 626816

Euler's totient: 334134 927360

Möbius: 0

$n = a^2 + b^2 + c^2 + d^2$

a = 566600

b = 488440

c = 393872

d = 379736

Show divisors

Time elapsed: 0d 0h 0m 0.0s

Modular multiplications:

- Sum of squares: 194

Timings:


- Probable prime test of 5 numbers: 0d 0h 0m 0.0s

Written by Dario Alpern. Last updated on 26 March 2025.

20 dígitos:

c7fd09e3c08956a05dbd

em decimal: 944418668966570065157565

← → ↻  alpertron.com.ar/ECM.HTM

Value

Actions

Only evaluate Is prime? Factor Help Config

Open wizard From file Blockly mode Clear input

Type one number

Press the **Help** button to get help about this application. Press it again to return to the factorization. You can also

- 944418 668966 570065 157565 = 3 × 5 × 13 × 89 × 274 062827 × 198559 107589

Number of divisors: 64

Sum of divisors: 1 645590 357470 455135 084800

Euler's totient: 459720 477457 657774 516224

Möbius: 1

$n = a^2 + b^2 + c^2 + d^2$

a = 804082 476754

b = 534503 430873

c = 109786 705954

d = 11090 587202

Show divisors

Time elapsed: 0d 0h 0m 0.0s

Modular multiplications:

- ECM: 25675
- Probable prime checking: 265
- Sum of squares: 305

Timings:

- Probable prime test of 11 numbers: 0d 0h 0m 0.0s
- Factoring 1 number using ECM: 0d 0h 0m 0.0s

Written by Dario Alpern. Last updated on 26 March 2025

30 dígitos:

**c7fd09e3c08956a05dbd31f442022b**

em decimal: 1038399308017476748068948889621889579

The screenshot shows a web browser window with the address bar displaying "alpertron.com.ar/ECM.HTM". The main interface has a "Value" input field containing the number "1038399308017476748068948889621889579". Below the input field is a "Actions" menu with buttons for "Only evaluate", "Is prime?", "Factor", "Help", "Config", "Open wizard", "From file", "Blockly mode", and "Clear input". To the right of the "Actions" menu, the text "Type one" is partially visible.

Press the **Help** button to get help about this application. Press it again to return to the factorization. You

- 1 038399 308017 476748 068948 889621 889579 (37 digits) =  $41 \times 22281\ 419551 \times 1\ 136678\ 5$

Number of divisors: 8

Sum of divisors: 1 063726 120455 887413 119775 324154 398080 (37 digits)

Euler's totient: 1 013072 495581 339440 183064 307500 776000 (37 digits)

Möbius: -1

$n = a^2 + b^2 + c^2 + d^2$

$a = 722240\ 269929\ 178633$

$b = 530496\ 538466\ 466300$

$c = 476653\ 302831\ 613353$

$d = 90240\ 523518\ 623659$

Show divisors

Time elapsed: 0d 0h 0m 0.0s

Modular multiplications:

- ECM: 45076
- Probable prime checking: 608
- Sum of squares: 366

Timings:

- Probable prime test of 5 numbers: 0d 0h 0m 0.0s
- Factoring 1 number using ECM: 0d 0h 0m 0.0s

Written by Dario Alpern. Last updated on 26 March 2025.

40 dígitos:

c7fd09e3c08956a05dbd31f442022baea32281f4

**em decimal:**

**1141732113439767866725521058309511564917282603508**

alpertron.com.ar/ECM.HTM

Value

1141732113439767866725521058309511564917282603508

Actions

Only evaluate

Is prime?

Factor

Help

Config

Open wizard

From file

Blockly mode

Clear input

Press the **Help** button to get help about this application. Press it again to return to the factor

- 1 141732 113439 767866 725521 058309 511564 917282 603508 (49 digits) =  $2^2 \times 15$

Number of divisors: 48

Sum of divisors: 2 012698 875387 429038 759552 041184 798479 145810 784000 (49 digit

Euler's totient: 566680 692498 191305 067735 571652 818554 530686 259200 (48 digits)

Möbius: 0

$n = a^2 + b^2 + c^2 + d^2$

$a = 760519\ 139391\ 172598\ 134492$

$b = 551919\ 714676\ 186153\ 717260$

$c = 464477\ 564623\ 051578\ 660462$

$d = 207335\ 410803\ 543786\ 534620$

Show divisors

Time elapsed: 0d 0h 0m 0.0s

Modular multiplications:

- ECM: 45076
- Probable prime checking: 644
- Sum of squares: 302

Timings:

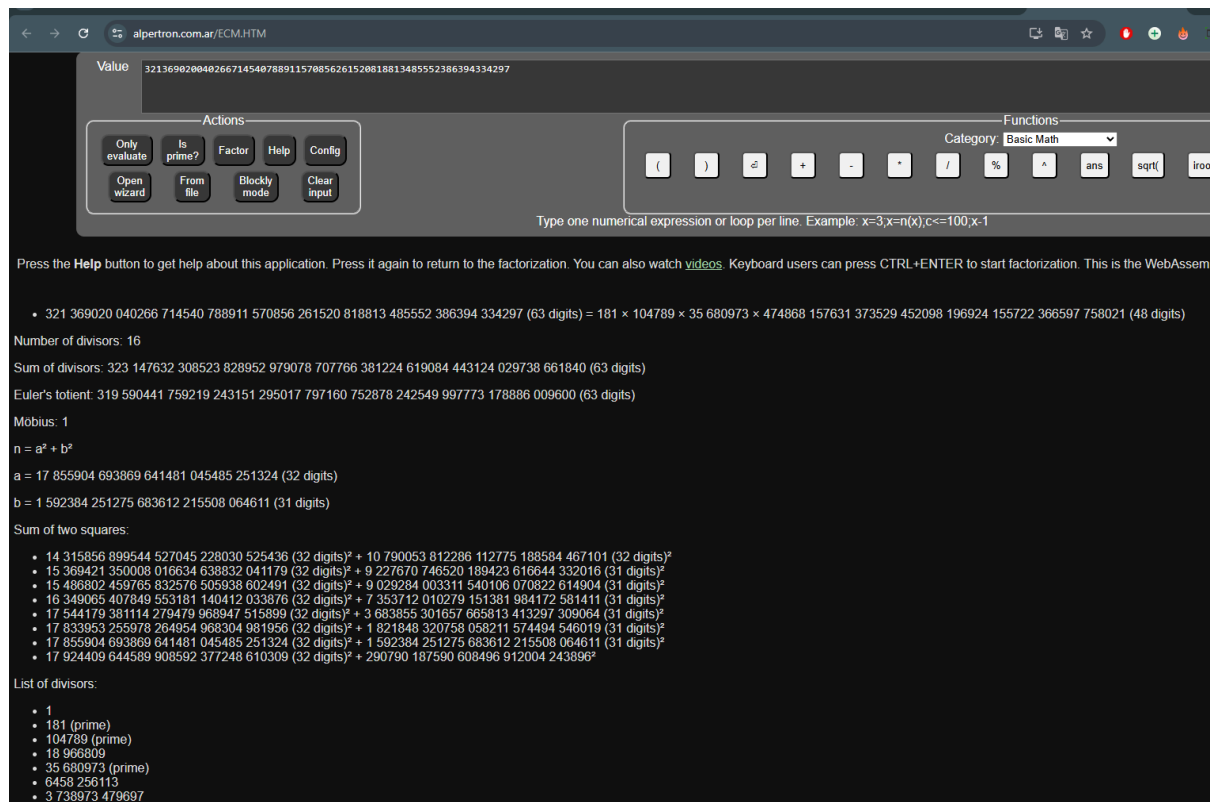
- Probable prime test of 9 numbers: 0d 0h 0m 0.0s
- Factoring 1 number using ECM: 0d 0h 0m 0.0s

50 dígitos:

**c7fd09e3c08956a05dbd31f442022baea32281f43a602ab4a459**

em decimal:

**32136902004026671454078891157085626152081881348555238  
6394334297**



COMPLETO:

**c7fd09e3c08956a05dbd31f442022baea32281f43a602ab4a459c866e4  
e449643d3a2a891c19cf63e87fea9243f75bf71fe63c988dbd32d8f931b  
bfde526dacf**

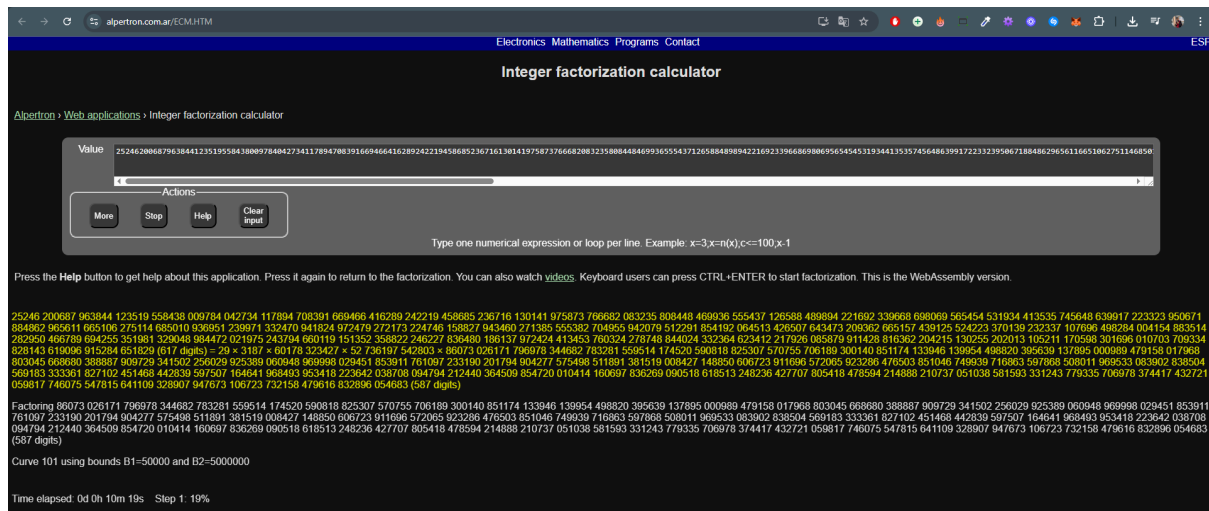
**a9a172822f848c6d9ae3443444ec65d763a9805fdeb71bc2e64fab18a2  
4fa90931d95e6c392dca7488bc125d034983c7fd9bd255d5b1951a351  
407c672fb8e5e**

**cbd5b41b9cba44a2496b1139e30aa145495e1d46b690e8657cc3c1a48  
4c057949862f05729a800236f02efbd3d169d7733ad1d0197cd6fd09a9  
4bb70a04a9bf4**

**b0d66f96ff3fc1b9b9d6bdca87eaf97e26d2434e6595c861cc2c29da0d6  
2dc9a4f131621f34eae35eb9088b7311098c059df828bb05882057176c  
308f0678615**

em decimal:

2524620068796384412351955843800978404273411789470839166946  
641628924221945868523671613014197587376668208323580844846  
993655543712658848989422169233966869806956545453193441353  
5745648639917223323950671884862965611665106275114685010936  
951239971332470941824972479272173224746158827943460271385  
555382704955942079512291854192064513426507643473209362665  
157439125524223370139232337107696498284004154883514282950  
4667896942553519813290489844720219752437946601191513523588  
222462278364801861379724244134537603242787488440243323646  
2341221792608587991142881636220421513025520201310521117059  
8301696010703709334828143619096915284651829



## Qual foi o último tamanho possível de fatoração? Qual a relação disso com a segurança do RSA?

O último tamanho possível de fatoração foi com os primeiros 50 dígitos hexadecimais de  $n$ , que equivalem a um número decimal ainda pequeno o suficiente para ser fatorado rapidamente pela ferramenta online. A partir do momento em que o valor de  $n$  completo (2048 bits) foi utilizado, a fatoração se tornou impraticável devido ao tamanho extremamente elevado do número.

Essa dificuldade de fatorar números grandes está diretamente relacionada à segurança do RSA: o algoritmo baseia sua robustez na complexidade de fatorar o produto de dois números primos muito grandes. Se a fatoração de



n fosse viável, a chave privada poderia ser descoberta, comprometendo a criptografia. Como isso é computacionalmente inviável com o  $n$  completo, a segurança do RSA é mantida.