ALGORITMOS ALEATORIZADOS: CORRETUDE

Análise de Algoritmos

Prof. André Vignatti

Teorema Fundamental da Álgebra (informalmente)**:**
um polinômio de grau n tem exatamente n raízes

$3x + 6$ (grau 1, 1 raiz)

$x^2 - 3x + 2$ (grau 2, raizes $x = 1$ e $x = 2$)

$(x-1)(x-1)(x-2)$ (grau 3, 3 raizes)

$\dots$

NOSSO PRIMEIRO PROBLEMA

Dados $F(x)$ e $G(x)$ dois polinômios de grau d, por exemplo:

$$F(x) = (x+1)(x-2)(x+3)(x-4)(x+5)(x-6)$$
$$G(x) = x^6 - 7x^3 + 25$$

NOSSO PRIMEIRO PROBLEMA

Dados $F(x)$ e $G(x)$ dois polinômios de grau d, por exemplo:

$$F(x) = (x+1)(x-2)(x+3)(x-4)(x+5)(x-6)$$
$$G(x) = x^6 - 7x^3 + 25$$

Como saber se $F(x) \equiv G(x)$?

NOSSO PRIMEIRO PROBLEMA

Dados $F(x)$ e $G(x)$ dois polinômios de grau d, por exemplo:

$$F(x) = (x+1)(x-2)(x+3)(x-4)(x+5)(x-6)$$
$$G(x) = x^6 - 7x^3 + 25$$

Como saber se $F(x) \equiv G(x)$?

- Solução natural em $O(d^2)$

Considere o seguinte algoritmo aleatorizado:

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

Considere o seguinte algoritmo aleatorizado:

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme. **(tempo constante - fim da aula)**

2. verifique se $F(r)$ é igual a $G(r)$. **(tempo linear)**

3. se $F(r) = G(r)$, devolva SIM; **(tempo constante)**

4. caso contrário, devolva NÃO. **(tempo constante)**

VP executa em tempo $O(d)$

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

O algoritmo pode dar resposta errada?

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

O algoritmo pode dar resposta errada?

Se ele devolve NÃO:

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.

2. verifique se $F(r)$ é igual a $G(r)$.

3. se $F(r) = G(r)$, devolva SIM;

4. caso contrário, devolva NÃO.

$\boxed{\text{O algoritmo pode dar resposta errada?}}$

Se ele devolve NÃO:

- então $F(r) \neq G(r)$

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

O algoritmo pode dar resposta errada?

Se ele devolve NÃO:

- então $F(r) \neq G(r)$
- as funções são diferentes em pelo menos um valor ("r")

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

$\boxed{\text{O algoritmo pode dar resposta errada?}}$

Se ele devolve NÃO:

- então $F(r) \neq G(r)$
- as funções são diferentes em pelo menos um valor ("r")
- então, 100% certeza que $F \neq G$

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO. `(100% correto)`

O algoritmo pode dar resposta errada?

Se ele devolve NÃO:

- então $F(r) \neq G(r)$
- as funções são diferentes em pelo menos um valor ("r")
- então, 100% certeza que $F \neq G$
- a resposta NÃO no passo 4 é 100% correta!

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

Um problema: Seja $F(x) = x^2$ e $G(x) = x^2 - 2x + 4$. Claramente $F(x) \neq G(x)$.

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM;
4. caso contrário, devolva NÃO.

Um problema: Seja $F(x) = x^2$ e $G(x) = x^2 - 2x + 4$. Claramente $F(x) \neq G(x)$.

- se amostrar $r = 2$, ele devolve SIM $\Longrightarrow$ o algoritmo **falha!**
- se amostrar $r = 3$, ele devolve NÃO $\Longrightarrow$ o algoritmo **acerta!**

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

$$\boxed{\text{Quando o algoritmo dá a resposta errada?}}$$

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$
- dois casos onde $H(r) = 0$

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$
- dois casos onde $H(r) = 0$
    1. H é zero em **todo** valor: então $H = 0 \Rightarrow F - G = 0 \Rightarrow F = G$

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$
- dois casos onde $H(r) = 0$
    1. H é zero em **todo** valor: então $H = 0 \Rightarrow F - G = 0 \Rightarrow F = G$
    2. H é zero em **alguns** valores:

Algoritmo VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$
- dois casos onde $H(r) = 0$
  1. H é zero em **todo** valor: então $H = 0 \Rightarrow F - G = 0 \Rightarrow F = G$
  2. H é zero em **alguns** valores:
     - então $H \neq 0$, mas H é zero em suas **raízes**

ALGORITMO VP

1. escolha $r \in \{1, \dots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$.
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Quando o algoritmo dá a resposta errada?

Se ele devolve SIM:

- então $F(r) = G(r)$
- então $F(r) - G(r) = 0$ (seja $H(x) = F(x) - G(x)$)
- então $H(r) = 0$
- dois casos onde $H(r) = 0$
  1. H é zero em **todo** valor: então $H = 0 \Rightarrow F - G = 0 \Rightarrow F = G$
  2. H é zero em **alguns** valores:
     - então $H \neq 0$, mas H é zero em suas **raízes**
     - então $F \neq G$, e o algoritmo **falha quando** r **é raíz de** H

Somos “azarados” quando selecionamos raízes de H. Quantas vezes somos azarados?

Somos “azarados” quando selecionamos raízes de H. Quantas vezes somos azarados?

$H(x)$ tem grau $\leq d \Longrightarrow H(x)$ tem $\leq d$ raízes. Assim,

$$\Pr(VP\ falhar) \leq \frac{d}{100d} = \frac{1}{100}$$

Somos "azarados" quando selecionamos raízes de H. Quantas vezes somos azarados?

$$H(x) \text{ tem grau } \leq d \implies H(x) \text{ tem } \leq d \text{ raízes. Assim,}$$

$$\Pr(VP\ falhar) \leq \frac{d}{100d} = \frac{1}{100}$$

formalmente, o que "VP falhar" significa?

Somos “azarados” quando selecionamos raízes de H. Quantas vezes somos azarados?

$H(x)$ tem grau $\leq d \implies H(x)$ tem $\leq d$ raízes. Assim,

$$\Pr(VP\ falhar) \leq \frac{d}{100d} = \frac{1}{100}$$

formalmente, o que “VP falhar” significa?

o que exatamente é uma probabilidade?

Somos "azarados" quando selecionamos raízes de H. Quantas vezes somos azarados?

$$H(x) \text{ tem grau} \leq d \Longrightarrow H(x) \text{ tem} \leq d \text{ raízes. Assim,}$$

$$\Pr(VP\ falhar) \leq \frac{d}{100d} = \frac{1}{100}$$

o que exatamente é uma probabilidade?

formalmente, o que "VP falhar" significa?

nós usamos nossa intuição! Mas essa é a forma correta de calcular a probabilidade?

O QUE ACONTECE SE VOCÊ NÃO FORMALIZAR?

Uma mulher está grávida de gêmeos

Depois do exame, ela descobriu que um bebê é menina

Qual é a probabilidade dela ter duas meninas?

O QUE ACONTECE SE VOCÊ NÃO FORMALIZAR?

Uma mulher está grávida de gêmeos

Depois do exame, ela descobriu que um bebê é menina

Qual é a probabilidade dela ter duas meninas?

probabilidade é traiçoeira! Devemos formalizar nossa conversa!!

AXIOMAS DE PROBABILIDADE

Definição. *Um **espaço de probabilidade discreto** tem 3 componentes:*

- conjunto Ω, chamado de *espaço amostral*
- conjunto $\mathcal{F}$ de todos subconjuntos de Ω, cada $E \in \mathcal{F}$ é chamado de *evento*.
- função de probabilidade $\Pr : \mathcal{F} \to \mathbb{R}^+$

AXIOMAS DE PROBABILIDADE

Definição. *Um **espaço de probabilidade discreto** tem 3 componentes:*

- conjunto Ω, chamado de *espaço amostral*
- conjunto $\mathcal{F}$ de todos subconjuntos de Ω, cada $E \in \mathcal{F}$ é chamado de *evento.*
- função de probabilidade $\Pr : \mathcal{F} \to \mathbb{R}^+$

Exemplo. Se $\Omega = \{\bigstar, \blacksquare, \spadesuit\}$ então

$$\mathcal{F} = \Big\{\emptyset, \{\bigstar\}, \{\blacksquare\}, \{\spadesuit\}, \{\bigstar, \blacksquare\}, \{\bigstar, \spadesuit\}, \{\blacksquare, \spadesuit\}, \{\bigstar, \blacksquare, \spadesuit\}\Big\}$$

E $\{\bigstar, \blacksquare\}$ é exemplo de um evento.

Definição (Axiomas de Kolmogorov). *Uma* função de probabilidade *é uma função* $\Pr : \mathcal{F} \to \mathbb{R}^+$ *tal que*

- $0 \leq \Pr(E) \leq 1, \ \forall E \in \mathcal{F}$
- $\Pr(\Omega) = 1$
- Seja $E_1, E_2, \dots,$ eventos **disjuntos**. Então

$$\Pr(E_1 \cup E_2 \dots) = \Pr(E_1) + \Pr(E_2) + \dots$$

Definição (Axiomas de Kolmogorov). *Uma* função de probabilidade *é uma função* $\Pr : \mathcal{F} \to \mathbb{R}^+$ *tal que*

- $0 \le \Pr(E) \le 1,\ \forall E \in \mathcal{F}$
- $\Pr(\Omega) = 1$
- Seja $E_1, E_2, \ldots,$ eventos **disjuntos**. Então

$$\Pr(E_1 \cup E_2 \ldots) = \Pr(E_1) + \Pr(E_2) + \ldots$$

Exemplo. Na verificação de polinômios

- $\Omega = \{1, \ldots, 100d\}$
- Cada escolha de $r = i$ é o evento simples $E_i = \{i\}$
- r é escolhido uniformemente $\implies \Pr(E_i) = \Pr(E_j),\ \forall i, j.$
- $\Pr(\Omega) = 1 \Rightarrow \Pr(E_i) = \frac{1}{100d}$ `(pois` $\bigcup_{i \ge 1} E_i = \Omega$`).`

Exemplo. Considere o lance de um dado de 6 lados.

- $\Omega = \{\epsdice{1}, \epsdice{2}, \epsdice{3}, \epsdice{4}, \epsdice{5}, \epsdice{6}\}$.

Exemplo de eventos que podemos considerar

Exemplo. Considere o lance de um dado de 6 lados.

- $\Omega = \{$⚀, ⚁, ⚂, ⚃, ⚄, ⚅$\}$.

Exemplo de eventos que podemos considerar

- E' = evento do dado mostrar número par = $\{$⚁, ⚃, ⚅$\}$.

Exemplo. Considere o lance de um dado de 6 lados.

- $\Omega = \{$⚀, ⚁, ⚂, ⚃, ⚄, ⚅$\}$.

Exemplo de eventos que podemos considerar

- $E' =$ evento do dado mostrar número par $= \{$⚁, ⚃, ⚅$\}$.
- $E'' =$ evento do dado mostrar número menor que 3 $= \{$⚀, ⚁$\}$.

Exemplo. Considere o lance de um dado de 6 lados.

- $\Omega = \{\epsdice{1}, \epsdice{2}, \epsdice{3}, \epsdice{4}, \epsdice{5}, \epsdice{6}\}$.

Exemplo de eventos que podemos considerar

- E' = evento do dado mostrar número par $= \{\epsdice{2}, \epsdice{4}, \epsdice{6}\}$.
- E'' = evento do dado mostrar número menor que $3 = \{\epsdice{1}, \epsdice{2}\}$.
- E''' = evento do dado mostrar número primo$= \{\epsdice{2}, \epsdice{3}, \epsdice{5}\}$.

Como diminuir a probabilidade de erro para $\frac{1}{1\text{bilhão}}$?

Lema. *Para eventos E_1 e E_2,*

$$\Pr(E_1 \cup E_2) = \Pr(E_1) + \Pr(E_2) - \Pr(E_1 \cap E_2)$$

Corolário. *Para eventos E_1 e E_2,*

$$\Pr(E_1 \cup E_2) \leq \Pr(E_1) + \Pr(E_2)$$

Lema (Limitante da União). *Dado eventos $E_1, E_2, \ldots,$*

$$\Pr\left(\bigcup_{i \geq 1} E_i\right) \leq \sum_{i \geq 1} \Pr(E_i)$$

Lema (Princípio da Inclusão-Exclusão). *Dado eventos* $E_1, E_2, \ldots,$

$$
\begin{aligned}
\Pr\left(\bigcup_{i\geq 1} E_i\right) &= \sum_{i\geq 1}\Pr(E_i) \\
&\quad - \sum_{i<j}\Pr(E_i\cap E_j) \\
&\quad + \sum_{i<j<k}\Pr(E_i\cap E_j\cap E_k) \\
&\quad \vdots \\
&\quad (-1)^{l+1}\sum_{i_1<i_2<\ldots<i_l}\Pr(E_{i_1}\cap E_{i_2}\cap\ldots\cap E_{i_l})
\end{aligned}
$$

INDEPENDÊNCIA

Definição. *Dois eventos E e F são ditos* ***independentes*** *se*

$$\Pr(E \cap F) = \Pr(E) \cdot \Pr(F)$$

e eventos $E_1, \ldots, E_k$ são ***mutuamente independentes*** *se $\forall I \subseteq \{1, \ldots, k\}$ temos*

$$\Pr\left(\bigcap_{i \in I} E_i\right) = \prod_{i \in I} \Pr(E_i).$$

ALGORITMO VP

1. escolha $r \in \{1, \ldots, 100d\}$ aleatoriamente de maneira uniforme.
2. verifique se $F(r)$ é igual a $G(r)$. `(tempo` $O(d)$`)`
3. se $F(r) = G(r)$, devolva SIM; `(nem sempre correto)`
4. caso contrário, devolva NÃO. `(100% correto)`

Como diminuir a probabilidade de erro para $\frac{1}{1\text{bilhão}}$?

- 1ª tentativa: aumentar o espaço amostral
  - faixa de valores limitada pela precisão da máquina
  - sorteio do r pode não levar tempo constante!

- 2ª tentativa: executar várias vezes o algoritmo

ALGORITMO VP_k

1. execute o algoritmo VP k vezes (com reposição).
2. devolve NÃO se em uma das k execuções o VP devolve não;
3. caso contrário, devolve SIM.

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Os eventos E_i são mutuamente independentes.

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Os eventos E_i são mutuamente independentes.

- A probabilidade do algoritmo falhar é:

$$\Pr(E_1 \cap E_2 \cap \ldots \cap E_k)$$

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Os eventos E_i são mutuamente independentes.

- A probabilidade do algoritmo falhar é:

$$\Pr(E_1 \cap E_2 \cap \ldots \cap E_k) = \prod_{i=1}^{k} \Pr(E_i)$$

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Os eventos E_i são mutuamente independentes.

- A probabilidade do algoritmo falhar é:

$$\Pr(E_1 \cap E_2 \cap \ldots \cap E_k) = \prod_{i=1}^{k} \Pr(E_i) \leq \prod_{i=1}^{k} \frac{d}{100d}$$

- Seja E_i o evento do algoritmo escolher raiz de $F(x) - G(x) = 0$ na i-ésima execução de VP.

- Os eventos E_i são mutuamente independentes.

- A probabilidade do algoritmo falhar é:

$$\Pr(E_1 \cap E_2 \cap \ldots \cap E_k) = \prod_{i=1}^{k} \Pr(E_i) \leq \prod_{i=1}^{k} \frac{d}{100d} \leq \left(\frac{1}{100}\right)^k$$

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

Método popular: **Gerador de Congruência Linear**

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

Método popular: **Gerador de Congruência Linear**

- é uma recorrência: $f(n) = (af(n-1)+c) \mod m$,

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

Método popular: **Gerador de Congruência Linear**

- é uma recorrência: $f(n) = (af(n-1) + c) \mod m$,
- $f(0)$ é chamado de *semente* (seed)

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

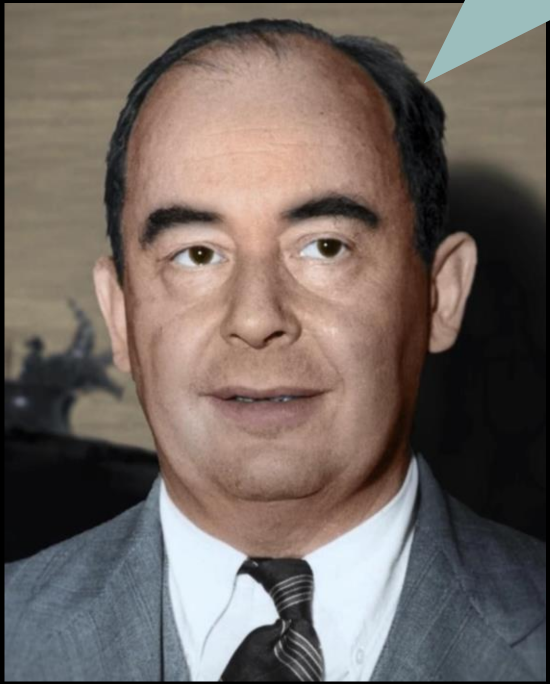
Método popular: **Gerador de Congruência Linear**

- é uma recorrência: $f(n) = (af(n-1) + c) \mod m$,
- $f(0)$ é chamado de *semente* (seed)
- em linguagem `C`: $a = 1103515245, c = 12345, m = 2^{31}$.

Suposição computacional: obter um número (pseudo) aleatório leva tempo $\Theta(1)$.

Método popular: **Gerador de Congruência Linear**

- é uma recorrência: $f(n) = (af(n-1) + c) \mod m$,
- $f(0)$ é chamado de *semente* (seed)
- em linguagem `C`: $a = 1103515245, c = 12345, m = 2^{31}$.
- em linguagem `Java`: $a = 25214903917, c = 11, m = 2^{48}$.



Anyone who considers arithmetical methods of producing random digits is, of course, in a state of sin.