Programas de Evaluación en Grupos

Pedro Villalba

NIDTEC
Facultad Politécnica

Esquema

1. Programas de evaluación con entradas
2. Computabilidad y costo de evaluación
3. Propiedades de los programas de evaluación
4. Complejidad de programas de evaluación

Programas de evaluación con entradas

Conceptos previos

$(G,*)$ es un grupo $\rightsquigarrow$

- $G \neq \emptyset$
- $* : G \times G \to G$
- $*$ es asociativa
- $\exists\, e \in G : a * e = e * a = a,\ \forall\, a \in G$
- $\forall\, a \in G,\ \exists\, a^{-1} \in G : a * a^{-1} = a^{-1} * a = e$

$G = \langle S \rangle \quad \rightsquigarrow \quad S$ conjunto generador de G

$\rightsquigarrow \quad \forall\ g \in G \Rightarrow g = a_1 * a_2 * \cdots * a_n,\ a_i \in S \cup S^{-1}$

Ejemplos

1. *Son grupos:* $(\mathbb{Z},+)$, $(\mathbb{R},+)$, $(\mathbb{R}^{\times},\cdot)$. *Además* $\mathbb{Z} = \langle 1 \rangle$
2. S_n := *grupo simétrico de* $[n]$:= *permutaciones de* n *elementos,*
3. D_{2n} := *grupo diedral* := *grupo de simetrías de un polígono regular de* n *lados.*

Programas de evaluación con entradas

La idea

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1, \qquad \Gamma_6 = \Gamma_4 - \Gamma_5,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1, \qquad \Gamma_6 = \Gamma_4 - \Gamma_5,$$

$$\Gamma_7 = \Gamma_1 - \Gamma_5,$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1, \qquad \Gamma_6 = \Gamma_4 - \Gamma_5,$$

$$\Gamma_7 = \Gamma_1 - \Gamma_5, \qquad \Gamma_8 = \frac{\Gamma_6}{\Gamma_7}$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1, \qquad \Gamma_6 = \Gamma_4 - \Gamma_5,$$

$$\Gamma_7 = \Gamma_1 - \Gamma_5, \qquad \Gamma_8 = \frac{\Gamma_6}{\Gamma_7}$$

$$\therefore \quad \Gamma = (\Gamma_1, \cdots, \Gamma_8)$$

Programas de evaluación con entradas

La idea

$$f(X) = \frac{X^5 - 1}{X - 1}$$

$$\Gamma_1 := X, \qquad \Gamma_2 = \Gamma_1 * \Gamma_1, \qquad \Gamma_3 = \Gamma_2 * \Gamma_2,$$

$$\Gamma_4 = \Gamma_3 * \Gamma_1, \qquad \Gamma_5 = 1, \qquad \Gamma_6 = \Gamma_4 - \Gamma_5,$$

$$\Gamma_7 = \Gamma_1 - \Gamma_5, \qquad \Gamma_8 = \frac{\Gamma_6}{\Gamma_7}$$

$$\therefore \quad \Gamma = \underbrace{(\Gamma_1, \cdots, \Gamma_8)}_{\text{instrucciones}}$$

Programas de evaluación con entradas

Formalmente

Programas de evaluación con entradas

Formalmente

Definición

Sea G un grupo y $\langle S \rangle = G$, $a = (a_1, a_2, \cdots, a_n) \in S^n$, $n \in \mathbb{N}$. Un *programa de evaluación* sobre G es un par $\Gamma_G = (\Gamma, G)$ donde $\Gamma = (\Gamma_1, \Gamma_2, \cdots, \Gamma_t)$ es una secuencia finita de instrucciones y $t \in \mathbb{N}$ es su longitud.

Programas de evaluación con entradas

Formalmente

Definición

Sea G un grupo y $\langle S \rangle = G$, $a = (a_1, a_2, \cdots, a_n) \in S^n$, $n \in \mathbb{N}$. Un *programa de evaluación* sobre G es un par $\Gamma_G = (\Gamma, G)$ donde $\Gamma = (\Gamma_1, \Gamma_2, \cdots, \Gamma_t)$ es una secuencia finita de instrucciones y $t \in \mathbb{N}$ es su longitud.
Cada Γ_i tiene la siguiente forma:

Programas de evaluación con entradas

Formalmente

Definición

Sea G un grupo y $\langle S \rangle = G$, $a = (a_1, a_2, \cdots, a_n) \in S^n$, $n \in \mathbb{N}$. Un *programa de evaluación* sobre G es un par $\Gamma_G = (\Gamma, G)$ donde $\Gamma = (\Gamma_1, \Gamma_2, \cdots, \Gamma_t)$ es una secuencia finita de instrucciones y $t \in \mathbb{N}$ es su longitud.

Cada Γ_i tiene la siguiente forma:

- $\Gamma_i = a_j$ para algún $j \in [t]$, o

Programas de evaluación con entradas

Formalmente

Definición

Sea G un grupo y $\langle S \rangle = G$, $a = (a_1, a_2, \cdots, a_n) \in S^n$, $n \in \mathbb{N}$. Un *programa de evaluación* sobre G es un par $\Gamma_G = (\Gamma, G)$ donde $\Gamma = (\Gamma_1, \Gamma_2, \cdots, \Gamma_t)$ es una secuencia finita de instrucciones y $t \in \mathbb{N}$ es su longitud.

Cada Γ_i tiene la siguiente forma:

- $\Gamma_i = a_j$ para algún $j \in [t]$, o
- $\Gamma_i = \Gamma_j^{-1}$ para algún $j < i$ en $[t]$, o

Programas de evaluación con entradas

Formalmente

Definición

Sea G un grupo y $\langle S \rangle = G$, $a = (a_1, a_2, \cdots, a_n) \in S^n$, $n \in \mathbb{N}$. Un *programa de evaluación* sobre G es un par $\Gamma_G = (\Gamma, G)$ donde $\Gamma = (\Gamma_1, \Gamma_2, \cdots, \Gamma_t)$ es una secuencia finita de instrucciones y $t \in \mathbb{N}$ es su longitud.

Cada Γ_i tiene la siguiente forma:

- $\Gamma_i = a_j$ para algún $j \in [t]$, o
- $\Gamma_i = \Gamma_j^{-1}$ para algún $j < i$ en $[t]$, o
- $\Gamma_i = \Gamma_j \cdot \Gamma_k$ para algún $j, k < i$ en $[t]$.

Programas de evaluación con entradas

Ejemplos

1. *Sea el grupo diedral* $D_{12} = \langle \rho, \lambda \rangle$*. El siguiente es un programa de evaluación* $\Gamma_{D_{12}} = (\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5)$ *que nos permite obtener el elemento* $\lambda\rho^3$ *de* D_{12}*:*

$$\begin{array}{lll} \Gamma_1 = \lambda & \Gamma_2 = \rho & \Gamma_3 = \Gamma_2 \cdot \Gamma_2 \\ \Gamma_4 = \Gamma_3 \cdot \Gamma_2 & \Gamma_5 = \Gamma_1 \cdot \Gamma_4 & \end{array}$$

Programas de evaluación con entradas

Ejemplos

1. *Sea el grupo diedral* $D_{12} = \langle \rho, \lambda \rangle$*. El siguiente es un programa de evaluación* $\Gamma_{D_{12}} = (\Gamma_1, \Gamma_2, \Gamma_3, \Gamma_4, \Gamma_5)$ *que nos permite obtener el elemento* $\lambda\rho^3$ *de* D_{12}*:*

$$\begin{array}{lll} \Gamma_1 = \lambda & \Gamma_2 = \rho & \Gamma_3 = \Gamma_2 \cdot \Gamma_2 \\ \Gamma_4 = \Gamma_3 \cdot \Gamma_2 & \Gamma_5 = \Gamma_1 \cdot \Gamma_4 & \end{array}$$

2. *En* S_6*, si* $\alpha = (1\ 2\ 3\ 4\ 5\ 6)$ *y* $\beta = (1\ 2)$*, entonces* $S_6 = \langle \alpha, \beta \rangle$*. Un programa de evaluación que computa el elemento* $(1\ 2\ 3)(4\ 5\ 6)$ *de* S_6 *es:* $\Lambda_{S_6} = (\Lambda_1, \Lambda_2, \Lambda_3, \Lambda_4, \Lambda_5, \Lambda_6, \Lambda_7)$ *con:*

$$\begin{array}{llll} \Lambda_1 = \alpha; & \Lambda_2 = \beta; & \Lambda_3 = \Lambda_1 \cdot \Lambda_1; & \Lambda_4 = \Lambda_3 \cdot \Lambda_2; \\ \Lambda_5 = \Lambda_4 \cdot \Lambda_1; & \Lambda_6 = \Lambda_5 \cdot \Lambda_2; & \Lambda_7 = \Lambda_6 \cdot \Lambda_6. & \end{array}$$

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S; a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$.

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S; a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$. $b = \Gamma_G(S; a)$

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S;a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$. $b = \Gamma_G(S;a)$

Un programa de evaluación Γ_G computa un conjunto $F \subset G$ en n si, y sólo si, existe una entrada $(S;a)$ para el cual $F \subset \Gamma_G(S;a)$.

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S; a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$. $b = \Gamma_G(S; a)$

Un programa de evaluación Γ_G computa un conjunto $F \subset G$ en n si, y sólo si, existe una entrada $(S; a)$ para el cual $F \subset \Gamma_G(S; a)$.

Costo de evaluación

El costo de evaluación de un conjunto F, $L_G(F, n)$, es la longitud mínima de un programa de evaluación Γ_G que computa F en n.

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S; a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$. $b = \Gamma_G(S; a)$

Un programa de evaluación Γ_G computa un conjunto $F \subset G$ en n si, y sólo si, existe una entrada $(S; a)$ para el cual $F \subset \Gamma_G(S; a)$.

Costo de evaluación

El costo de evaluación de un conjunto F, $L_G(F, n)$, es la longitud mínima de un programa de evaluación Γ_G que computa F en n.
Si $F \not\subseteq G$, entonces $L_G(F, n) = \infty$.

Computabilidad y costo de evaluación

Computabilidad

Un programa de evaluación Γ_G es ejecutable en la entrada $(S; a)$ con salida $b \in G^t$ si $b_i = \Gamma_i$ para cada $i \in [t]$. $b = \Gamma_G(S; a)$

Un programa de evaluación Γ_G computa un conjunto $F \subset G$ en n si, y sólo si, existe una entrada $(S; a)$ para el cual $F \subset \Gamma_G(S; a)$.

Costo de evaluación

El costo de evaluación de un conjunto F, $L_G(F, n)$, es la longitud mínima de un programa de evaluación Γ_G que computa F en n.
Si $F \not\subseteq G$, entonces $L_G(F, n) = \infty$.

Babai-Szemerédi

Dado un grupo G de orden n y un conjunto generador S, el costo de evaluación de cada elemento de G es $\leq (1 + \log n)^2$.

Computabilidad y costo de evaluación

Consideremos los dos ejemplos anteriores:

Computabilidad y costo de evaluación

Consideremos los dos ejemplos anteriores:

Ejemplos

1. Como $\begin{cases} \Gamma_1 = \lambda & \Gamma_3 = \rho^2 & \Gamma_5 = \lambda\rho^3 \\ \Gamma_2 = \rho & \Gamma_4 = \rho^3 \end{cases}$
   esto implica que $\Gamma_{D_{12}}$ computa cualquier subconjunto del conjunto $\left\{\lambda,\ \rho,\ \rho^2,\ \rho^3,\ \lambda\rho^3\right\}$.

Computabilidad y costo de evaluación

Consideremos los dos ejemplos anteriores:

Ejemplos

1. Como $\left\{ \begin{array}{lll} \Gamma_1 = \lambda & \Gamma_3 = \rho^2 & \Gamma_5 = \lambda\rho^3 \\ \Gamma_2 = \rho & \Gamma_4 = \rho^3 & \end{array} \right.$
   esto implica que $\Gamma_{D_{12}}$ computa cualquier subconjunto del conjunto $\left\{\lambda,\ \rho,\ \rho^2,\ \rho^3,\ \lambda\rho^3\right\}$.

2. De igual manera, como
   $\left\{ \begin{array}{llll} \Lambda_1 = \alpha & \Lambda_3 = \alpha^2 & \Lambda_5 = \alpha^2\beta\alpha & \Lambda_7 = \alpha^2\beta\alpha\beta\alpha^2\beta\alpha\beta \\ \Lambda_2 = \beta & \Lambda_4 = \alpha^2\beta & \Lambda_6 = \alpha^2\beta\alpha\beta & \end{array} \right.$
   tenemos que Λ_{S_6} computa cualquier subconjunto de $\left\{\alpha,\ \beta,\ \alpha^2,\ \alpha^2\beta,\ \alpha^2\beta\alpha,\ \alpha^2\beta\alpha\beta,\ \alpha^2\beta\alpha\beta\alpha^2\beta\alpha\beta\right\}$.

Propiedades de los programas de evaluación

Proposición

Si un programa de evaluación Γ_G *computa un conjunto generador de* G *para algún* $n \in \mathbb{Z}^+$*, entonces* G *es finitamente generado.*

Propiedades de los programas de evaluación

Proposición

Si un programa de evaluación Γ_G *computa un conjunto generador de* G *para algún* $n \in \mathbb{Z}^+$, *entonces* G *es finitamente generado.*

Proposición

Si Γ_G *y* Λ_G *son dos programas de evaluación sobre* G *que computan los conjuntos* F_1 *y* F_2, *respectivamente, entonces existe un programa de evaluación* Φ_G *que computa* $F_1 \cup F_2$ *en* $2n$ *cuya longitud es, a lo sumo, la suma de las longitudes de* Γ_G *y* Λ_G.

Propiedades de los programas de evaluación

Proposición

Si un programa de evaluación Γ_G *computa un conjunto generador de* G *para algún* $n \in \mathbb{Z}^+$*, entonces* G *es finitamente generado.*

Proposición

Si Γ_G *y* Λ_G *son dos programas de evaluación sobre* G *que computan los conjuntos* F_1 *y* F_2*, respectivamente, entonces existe un programa de evaluación* Φ_G *que computa* $F_1 \cup F_2$ *en* $2n$ *cuya longitud es, a lo sumo, la suma de las longitudes de* Γ_G *y* Λ_G*.*

Proposición

El programa Φ_G *de la Proposición anterior también computa* $F_1 \cap F_2$ *en* $2n$*.*

Propiedades de los programas de evaluación

Extensión isomorfa

Sean G y G' dos grupos, se dice que $F \subset G$ tiene una extensión isomorfa F' en G' si, y sólo si, existen $H \leq G$ que contiene a F, $H' \leq G'$ que contiene a F', y un isomorfismo $\phi : H \to H'$ tal que $\phi(F) = F'$.

Propiedades de los programas de evaluación

Extensión isomorfa

Sean G y G' dos grupos, se dice que $F \subset G$ tiene una extensión isomorfa F' en G' si, y sólo si, existen $H \leq G$ que contiene a F, $H' \leq G'$ que contiene a F', y un isomorfismo $\phi : H \to H'$ tal que $\phi(F) = F'$. $F' = \widehat{F}_{G'}$

Propiedades de los programas de evaluación

Extensión isomorfa

Sean G y G' dos grupos, se dice que $F \subset G$ tiene una extensión isomorfa F' en G' si, y sólo si, existen $H \leq G$ que contiene a F, $H' \leq G'$ que contiene a F', y un isomorfismo $\phi : H \to H'$ tal que $\phi(F) = F'$. $F' = \widehat{F}_{G'}$

Simulación de programas de evaluación

Sea Γ_G un programa de evaluación. Para todo entero positivo n, decimos que un programa de evaluación $\Lambda_{G'}$ simula Γ_G si, y sólo si, para todo conjunto F computado por Γ_G en n, existe una extensión isomorfa de F en G' que es computado por $\Lambda_{G'}$ en n.

Propiedades de los programas de evaluación

Extensión isomorfa

Sean G y G' dos grupos, se dice que $F \subset G$ tiene una extensión isomorfa F' en G' si, y sólo si, existen $H \leq G$ que contiene a F, $H' \leq G'$ que contiene a F', y un isomorfismo $\phi : H \to H'$ tal que $\phi(F) = F'$. $\quad F' = \widehat{F}_{G'}$

Simulación de programas de evaluación

Sea Γ_G un programa de evaluación. Para todo entero positivo n, decimos que un programa de evaluación $\Lambda_{G'}$ simula Γ_G si, y sólo si, para todo conjunto F computado por Γ_G en n, existe una extensión isomorfa de F en G' que es computado por $\Lambda_{G'}$ en n. $\quad \Gamma_G \prec \Lambda_{G'}$

Propiedades de los programas de evaluación

Propiedades

1. Si G es un grupo no conmutativo y H es un grupo conmutativo, entonces no existen programas de evaluación Γ_G y Λ_H tal que $\Gamma_G \prec \Lambda_H$.

Propiedades de los programas de evaluación

Propiedades

1. Si G es un grupo no conmutativo y H es un grupo conmutativo, entonces no existen programas de evaluación Γ_G y Λ_H tal que $\Gamma_G \prec \Lambda_H$.

2. $\prec$ es de una relación de preorden.

Propiedades de los programas de evaluación

Propiedades

1. Si G es un grupo no conmutativo y H es un grupo conmutativo, entonces no existen programas de evaluación Γ_G y Λ_H tal que $\Gamma_G \prec \Lambda_H$.

2. $\prec$ es de una relación de preorden.
    1. $\Gamma_G \prec \Gamma_G$.

Propiedades de los programas de evaluación

Propiedades

1. Si G es un grupo no conmutativo y H es un grupo conmutativo, entonces no existen programas de evaluación Γ_G y Λ_H tal que $\Gamma_G \prec \Lambda_H$.

2. $\prec$ es de una relación de preorden.
   1. $\Gamma_G \prec \Gamma_G$.
   2. Si $\Gamma_G \prec \Lambda_{G'}$ y $\Lambda_{G'} \prec \Phi_{G''}$, entonces $\Gamma_G \prec \Phi_{G''}$.

Propiedades de los programas de evaluación

Propiedades

1. Si G es un grupo no conmutativo y H es un grupo conmutativo, entonces no existen programas de evaluación Γ_G y Λ_H tal que $\Gamma_G \prec \Lambda_H$.
2. $\prec$ es de una relación de preorden.
   1. $\Gamma_G \prec \Gamma_G$.
   2. Si $\Gamma_G \prec \Lambda_{G'}$ y $\Lambda_{G'} \prec \Phi_{G''}$, entonces $\Gamma_G \prec \Phi_{G''}$.
3. $\prec$ no es una relación de orden, pues no es simétrica.

Complejidad de programas de evaluación

Definición:

Sea $\mathcal{G}$ una familia de grupos y F un conjunto tal que $F \subset G$ para cada $G \in \mathcal{G}$. La complejidad de evaluación de F con respecto a $\mathcal{G}$ se define como $\mathcal{L}_{\mathcal{G}}(F, n) = \min_{G \in \mathcal{G}} L_G(F, n)$.

Complejidad de programas de evaluación

Definición:

Sea $\mathcal{G}$ una familia de grupos y F un conjunto tal que $F \subset G$ para cada $G \in \mathcal{G}$. La complejidad de evaluación de F con respecto a $\mathcal{G}$ se define como $\mathcal{L}_{\mathcal{G}}(F, n) = \min\limits_{G \in \mathcal{G}} L_G(F, n)$.

$\mathcal{C}$:= familia de todos los grupos conmutativos

$$\mathcal{L}_c(F, n) = \min_{G \in \mathcal{C}} L_G(F, n) \rightsquigarrow \text{ complejidad conmutativa de } F$$

Complejidad de programas de evaluación

Definición:

Sea $\mathcal{G}$ una familia de grupos y F un conjunto tal que $F \subset G$ para cada $G \in \mathcal{G}$. La complejidad de evaluación de F con respecto a $\mathcal{G}$ se define como $\mathcal{L}_{\mathcal{G}}(F,n) = \min_{G \in \mathcal{G}} L_G(F,n)$.

$\mathcal{C}$:= familia de todos los grupos conmutativos

$$\mathcal{L}_c(F,n) = \min_{G \in \mathcal{C}} L_G(F,n) \rightsquigarrow \text{ complejidad conmutativa de } F$$

$\mathcal{N}$:= familia de todos los grupos no conmutativos

$$\mathcal{L}_{nc}(F,n) = \min_{G \in \mathcal{N}} L_G(F,n) \rightsquigarrow \text{ complejidad no conmutativa de } F$$

Complejidad de programas de evaluación

Definición:

Sea $\mathcal{G}$ una familia de grupos y F un conjunto tal que $F \subset G$ para cada $G \in \mathcal{G}$. La complejidad de evaluación de F con respecto a $\mathcal{G}$ se define como $\mathcal{L}_{\mathcal{G}}(F, n) = \min_{G \in \mathcal{G}} L_G(F, n)$.

$\mathcal{C}$:= familia de todos los grupos conmutativos

$$\mathcal{L}_c(F, n) = \min_{G \in \mathcal{C}} L_G(F, n) \rightsquigarrow \text{ complejidad conmutativa de } F$$

$\mathcal{N}$:= familia de todos los grupos no conmutativos

$$\mathcal{L}_{nc}(F, n) = \min_{G \in \mathcal{N}} L_G(F, n) \rightsquigarrow \text{ complejidad no conmutativa de } F$$

$\mathscr{A}$:= familia de todos los grupos definibles

$$\mathcal{L}(F, n) = \min_{G \in \mathscr{A}} L_G(F, n) \rightsquigarrow \text{ complejidad total de } F$$

Complejidad de programas de evaluación

Proposición

Si $\mathcal{G}' \subset \mathcal{G}$, *entonces* $\mathcal{L}_{\mathcal{G}}(F,n) \leq \mathcal{L}_{\mathcal{G}'}(F,n)$

Complejidad de programas de evaluación

Proposición

Si $\mathcal{G}' \subset \mathcal{G}$, *entonces* $\mathcal{L}_{\mathcal{G}}(F,n) \leq \mathcal{L}_{\mathcal{G}'}(F,n)$

Proposición

Para todo grupo conmutativo G *existe un grupo no conmutativo* G' *tal que* $\mathcal{L}_{G'}(\widehat{F}_{\mathcal{G}'},n) \leq \mathcal{L}_{G}(F,n)$

Complejidad de programas de evaluación

Proposición

Si $\mathcal{G}' \subset \mathcal{G}$, *entonces* $\mathcal{L}_{\mathcal{G}}(F, n) \leq \mathcal{L}_{\mathcal{G}'}(F, n)$

Proposición

Para todo grupo conmutativo G *existe un grupo no conmutativo* G' *tal que* $\mathcal{L}_{G'}(\widehat{F}_{\mathcal{G}'}, n) \leq \mathcal{L}_G(F, n)$

Teorema

$\mathcal{L}_{nc}(\widehat{F}_{\mathcal{N}}, n) \leq \mathcal{L}_c(F, n)$.

Complejidad de programas de evaluación

Proposición

Si $\mathcal{G}' \subset \mathcal{G}$, *entonces* $\mathcal{L}_{\mathcal{G}}(F,n) \leq \mathcal{L}_{\mathcal{G}'}(F,n)$

Proposición

Para todo grupo conmutativo G *existe un grupo no conmutativo* G' *tal que* $\mathcal{L}_{G'}(\widehat{F}_{\mathcal{G}'}, n) \leq \mathcal{L}_{G}(F,n)$

Teorema

$\mathcal{L}_{nc}(\widehat{F}_{\mathcal{N}}, n) \leq \mathcal{L}_{c}(F,n)$.

http://www.cc.pol.una.py/∼mvillagra/files/programa-eval.pdf

FIN !!
Gracias