Pontificia Universidad Católica de Chile
Departamento de Ciencia de la Computación
IIC1253 - Matemáticas Discretas

Examen

20 de Noviembre de 2017
Profesores: Gabriel Diéguez - Fernando Suárez

Instrucciones

- En cada parte del examen debe contestar al menos dos preguntas. Si contesta las tres, se considerarán las dos mejores en el cálculo de su nota.
- Use lápiz pasta. Por el uso de lápiz mina usted pierde el derecho a recorreción.
- Rellene sus datos en cada hoja de respuesta que utilice.
- Cada pregunta debe responderse en hojas separadas.
- Entregue al menos una hoja por pregunta.
    - Si entrega la pregunta **completamente en blanco**, tiene nota mínima 1.5 en vez de 1.0 en la pregunta entregada.

Parte A (50 %)

Pregunta 1

a) Considere los siguientes principios:

Principio de Buen Orden

Todo subconjunto no vacío de los naturales tiene un menor elemento.

Principio de Inducción Simple

Sea A un subconjunto de $\mathbb{N}$. Si se cumple que:

1. $0 \in A$
2. Si $n \in A$, entonces $n + 1 \in A$

entonces $A = \mathbb{N}$.

Demuestre que el Principio de Buen Orden implica al Principio de Inducción Simple.

b) Sea P un conjunto de variables proposicionales, $\Sigma \subseteq L(P)$ y $\alpha, \beta, \gamma \in L(P)$.

Demuestre que $\Sigma \cup \{\alpha\} \models \gamma$ y $\Sigma \cup \{\beta\} \models \gamma$ si y sólo si $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$.

Solución

a) Se quiere demostrar que PBO $\Rightarrow$ PIS. Supongamos que tenemos un conjunto $A \subseteq \mathbb{N}$ que cumple con las anteriores características. Demostraremos esto por contradicción. Supongamos que $A \neq \mathbb{N}$. Entonces el conjunto $B = \mathbb{N} - A$ cumple con $B \subseteq \mathbb{N}$ y con $B \neq \emptyset$. Por el principio de buen orden, B debe tener un menor elemento, digamos $b \in B$. Es claro que $b \neq 0$ (ya que $0 \in A$), luego $b - 1$ pertenece a $\mathbb{N}$ y no a B, por lo que se cumple que $b - 1 \in A$. Dado que estamos suponiendo que A cumple las características de PIS entonces $b \in A$ lo que contradice el hecho de que b sea el menor elemento de B. La contradicción ocurre por el hecho de suponer que $A \neq \mathbb{N}$, luego A es igual a $\mathbb{N}$.

b) Debemos realizar la demostración para ambos lados.

($\Rightarrow$) Dado que $\Sigma \cup \{\alpha\} \models \gamma$ y $\Sigma \cup \{\beta\} \models \gamma$, demostraremos que $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$. Sea σ una valuación tal que $\sigma(\Sigma \cup \{\alpha \vee \beta\}) = 1$. Esto implica que $\sigma(\Sigma) = 1$ y $\sigma(\alpha \vee \beta) = 1$. Sin pérdida de generalidad, asumamos que $\sigma(\alpha) = 1$. Luego, dado que $\Sigma \cup \{\alpha\} \models \gamma$ obtenemos que $\sigma(\gamma) = 1$ y como σ es arbitrario concluimos que $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$.

($\Leftarrow$) Dado que $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$, demostraremos que $\Sigma \cup \{\alpha\} \models \gamma$ y $\Sigma \cup \{\beta\} \models \gamma$. Por contradicción supongamos que $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$ pero que $\Sigma \cup \{\alpha\} \not\models \gamma$ o $\Sigma \cup \{\beta\} \not\models \gamma$. Sin pérdida de generalidad asumamos que $\Sigma \cup \{\alpha\} \not\models \gamma$, luego debe existir una valuación σ tal que $\sigma(\Sigma \cup \{\alpha\}) = 1$ y $\sigma(\gamma) = 0$. Esto implica que $\sigma(\Sigma) = 1$ y $\sigma(\alpha) = 1$. Por lo

tanto, $\sigma(\Sigma \cup \{\alpha \vee \beta\}) = 1$ y como $\Sigma \cup \{\alpha \vee \beta\} \models \gamma$ tenemos que $\sigma(\gamma) = 1$, lo cual es una contradicción.

Pauta

a) 3 pts.

b)
- 1.5 pts. por ($\Rightarrow$).
- 1.5 pts. por ($\Leftarrow$).

Puntajes intermedios a criterio del corrector.

Pregunta 2

a) Sean A un conjunto y R, S relaciones sobre A. Diremos que S es la *clausura refleja de* R si:

- S contiene a R; vale decir, $R \subseteq S$,
- S es refleja, y
- S es mínima; vale decir, para toda relación refleja T sobre A que contiene a R, se tiene que $S \subseteq T$.

Demuestre que la relación $S = R \cup D$ es la clausura refleja de R, donde D es la *relación diagonal* definida como $D = \{(a, a) \mid a \in A\}$.

b) Sea $\mathcal{F}$ el conjunto de todas las funciones inyectivas de $\mathbb{N}$ en $\mathbb{N}$. ¿Es el conjunto $\mathcal{F}$ numerable o no? Demuestre su afirmación.

Solución

a) Para demostrar que S es la clausura refleja de R, debemos demostrar las tres propiedades enunciadas:

- Es claro que $R \subseteq S$, puesto que $S = R \cup D$, y luego todo elemento de R está en S.
- Para demostrar que S es refleja, debemos mostrar que $(a, a) \in S$ para todo $a \in A$. Es claro que esto se cumple, pues $S = R \cup D$, y D contiene a todos esos pares, y luego están en S.
- Sea T una relación refleja tal que $R \subseteq T$. Demostraremos directamente que $S \subseteq T$.

  Sea $(s_1, s_2) \in S$ un par cualquiera de la relación S. Como $S = R \cup D$ tenemos que $(s_1, s_2) \in R \vee (s_1, s_2) \in D$. En el primer caso, sabemos que $R \subseteq T$, y luego $(s_1, s_2) \in T$. En el segundo caso, sabemos que (s_1, s_2) es de la forma (a, a) para algún $a \in A$. Como T es refleja, sabemos que contiene todos los pares de esa forma, y luego $(s_1, s_2) \in T$. Hemos demostrado que todo par en la relación S está en la relación T, y luego $S \subseteq T$.

Habiendo demostrado las 3 propiedades, queda demostrado que S es la clausura refleja de R.

b) Demostramos que es no numerable por contradicción utilizando diagonalización de Cantor. Supongamos que es numerable y que podemos enumerar cada función dentro del conjunto $\mathcal{F}$, es decir:

$$\mathcal{F} = \{f_1, f_2, f_3, ...\}$$

Las ponemos a todas las funciones en una lista y las evaluamos en cada número natural.

$$\begin{array}{cccccc} f_1(1) & f_1(2) & \cdots & f_1(n) & \cdots \\ f_2(1) & f_2(2) & \cdots & f_2(n) & \cdots \\ \vdots & \vdots & \ddots & \vdots & \cdots \\ f_n(1) & f_n(2) & \cdots & f_n(n) & \cdots \\ \vdots & \vdots & \ddots & \vdots & \cdots \end{array}$$

Analizamos la diagonal y generamos la función:

$$f_d(n) = \sum_{i=1}^{n} f_i(i) + 1$$

Esta función es inyectiva ya que es estrictamente creciente. Además se tiene que $f_d(n) \neq f_n(n)$ $\forall n \in \mathbb{N}$, lo que implica que f_d no aparece en ninguna lista definida previamente, por lo tanto no es parte de $\mathcal{F}$. Sin embargo f_d es inyectiva y solo toma valores naturales, es decir debe pertenecer al conjunto $\mathcal{F}$. **Contradicción**, por lo tanto $\mathcal{F}$ es no numerable.

Pauta

a)
- 0.5 ptos. por contención.
- 1 pto. por reflexividad.
- 1.5 ptos. por mostrar que es mínima.

b)
- 1.5 pts. por construcción de la diagonalización.
- 1.5 pts. por mostrar que la función es inyectiva y no aparece en la lista.

Puntajes intermedios a criterio del corrector.

Pregunta 3

a) Demuestre que $\log(n!) \in \Theta(n \log(n))$.

b) Demuestre que todo grafo con más de un vértice tiene al menos dos vértices con el mismo grado.

Solución

a) Se quiere demostrar que $\log(n!) \in \Theta(n\log(n))$. Esto es equivalente a demostrar que:

$$\log(n!) \in \mathcal{O}(n\log(n) \cap \Omega(n\log(n))$$

- $\log(n!) \in \mathcal{O}(n\log(n))$. Para todo $n > 0$ se tiene que:

$$\begin{aligned}\log(n!) &= \log(1 \cdot 2 \cdot ... \cdot n) \\ &= \sum_{i=1}^{n} \log(i) \\ &\leq \sum_{i=1}^{n} \log(n) \\ &= n\log(n)\end{aligned}$$

  y por lo tanto $\log(n!) \in \mathcal{O}(n\log(n))$.

- $\log(n!) \in \Omega(n\log(n))$. Esto quiere decir que debemos encontrar una constante $c > 0$ y un $n_0 \in \mathbb{N}$ tal que:

$$n\log(n) \leq c\log(n!) \qquad \forall n > n_0$$

  Si se toma $c = 2$, se tiene que:

$$\begin{aligned}2\log(n!) &= \log(n!^2) \\ &= \log(1 \cdot 1 \cdot ... \cdot n \cdot n) \\ &= \log(\underbrace{1 \cdot n}_{\geq n} \cdot \underbrace{2 \cdot (n-1)}_{\geq n} \cdot ... \cdot \underbrace{n \cdot 1}_{\geq n}) \\ &\geq \log(n \cdot ... \cdot n) \\ &= \log(n^n) \\ &= n\log(n)\end{aligned}$$

  Luego $n\log(n) \leq 2\log(n!)$ para $n \geq 1$, por lo tanto $\log(n!) \in \Omega(n\log(n))$.

b) Demostremos esto por contradicción. Supongamos que no, es decir supongamos que en un grafo de n vértices no existe ningún par de vértices con el mismo grado. Sabemos que el grado máximo de un vértice es $n-1$ y el grado mínimo es 0. Luego, el conjunto de grados para los vértices es $\{0, ..., n-1\}$, como tiene n elementos, cada vértice debería tener como grado un elemento de dicho conjunto. Sin embargo si un vértice tiene grado $n-1$ significa que está conectado con todos los otros vértices, lo que hace imposible que exista un vértice con grado 0, contradicción. Por lo tanto debe existir al menos dos vértices con el mismo grado.

Pauta

a)
- 1.5 pts. por $\mathcal{O}(n \log(n))$.
- 1.5 pts. por $\Omega(n \log(n))$.

b) 3 pts.

Puntajes intermedios a criterio del corrector.

Parte B (50 %)

Pregunta 4

Sean a, b, c, d y n enteros.

a) Demuestre que $a \equiv_n b$ si y sólo si $a \bmod n = b \bmod n$.

b) Demuestre que si $a \equiv_n b$ y $c \equiv_n d$, entonces:
- $(a + c) \equiv_n (b + d)$
- $(a \cdot c) \equiv_n (b \cdot d)$

Solución

a) Podemos expresar a y b en términos de n como:

$$a = \alpha \cdot n + \beta \qquad 0 \leq \beta < n \tag{1}$$
$$b = \gamma \cdot n + \delta \qquad 0 \leq \delta < n \tag{2}$$

donde $\beta = a \bmod n$ y $\delta = b \bmod n$. Mostraremos ahora ambas direcciones:

($\Rightarrow$) Por contrapositivo, supongamos que

$$a \bmod n \neq b \bmod n \tag{3}$$

Mostraremos que $a \not\equiv_n b$, o equivalentemente, $n \not| (b - a)$.

Por (3), tenemos que $\beta \neq \delta$ en (1) y (2). Sin pérdida de generalidad, supongamos que $\beta < \delta$. Restando (2) y (1), tenemos que $b - a = (\gamma - \alpha) \cdot n + (\delta - \beta)$. Como $\beta < \delta$, tenemos que $1 \leq (\delta - \beta) \leq \delta < n$, y por lo tanto $n \not| (b - a)$, pues el resto de dividir $(b - a)$ por n no es 0.

($\Leftarrow$) En este caso, tenemos que $\beta = \delta$. Restando (2) y (1):

$$b - a = (\gamma - \alpha) \cdot n \Leftrightarrow n \mid (b - a) \Leftrightarrow a \equiv_n b.$$

b) Por definición de equivalencia modular, $n \mid (b-a)$ y $n \mid (d-c)$.

Por definición de la relación divide a:

$$b - a = k_1 \cdot n \iff b = a + k_1 \cdot n \quad (4)$$
$$d - c = k_2 \cdot n \iff d = c + k_2 \cdot n \quad (5)$$

- Sumando (4) y (5):

$$\begin{aligned} & b + d = a + c + (k_1 + k_2) \cdot n \\ \Leftrightarrow \quad & (b+d) - (a+c) = (k_1 + k_2) \cdot n \\ \Leftrightarrow \quad & (b+d) - (a+c) = k_3 \cdot n \\ \Leftrightarrow \quad & n \mid (b+d) - (a+c) \\ \Leftrightarrow \quad & a + c \equiv_n b + d \end{aligned}$$

- Multiplicando (4) y (5):

$$\begin{aligned} & b \cdot d = (a + k_1 \cdot n)(c + k_2 \cdot n) \\ \Leftrightarrow \quad & bd = ac + a \cdot k_2 \cdot n + c \cdot k_1 \cdot n + k_1 \cdot k_2 \cdot n^2 \\ \Leftrightarrow \quad & bd = ac + (a \cdot k_2 + c \cdot k_1 + k_1 \cdot k_2 \cdot n) \cdot n \\ \Leftrightarrow \quad & bd = ac + k_4 \cdot n \\ \Leftrightarrow \quad & bd - ac = k_4 \cdot n \\ \Leftrightarrow \quad & n \mid bd - ac \\ \Leftrightarrow \quad & ac \equiv_n bd \end{aligned}$$

Pauta

a) 1.5 ptos. cada dirección.

b) 1.5 ptos. cada propiedad (suma y multiplicación).

Puntajes intermedios a criterio del corrector.

Pregunta 5

Decimos que un ciclo en un grafo es un *círculo* si no repite vértices. Considere el siguiente problema:

$$\text{Half-Circle} = \{\ G \mid G \text{ tiene un círculo de tamaño } \tfrac{n}{2}\ \}$$

En otras palabras, las instancias $I_{\text{HALF-CIRCLE}}$ son todos los grafos y el lenguaje $L_{\text{Half-Circle}}$ son todos los grafos que contienen un círculo de tamaño $\frac{n}{2}$. Demuestre que el problema Half-Circle es NP-completo.

Solución

- Half-Circle $\in$ NP: Es fácil notar que Half-Circle está en NP. El certificado c para el grafo $G(V, E)$ es la instancia del ciclo que recorre G, el cuál esta acotado polinomialmente por la cantidad de vértices. Para verificar el certificado, podemos utilizar el siguiente algoritmo:

  Algoritmo: Half-Circle$(G(V, E), c = (v_1, \ldots, v_m))$

  ```
  k ← 0
  S ← {}
  for i = 1 to m do
      if v_i, v_{i+1} ∉ E or v_i ∈ S then
          return False
      else
          k ← k + 1
          S ← S ∪ {v_i}
      end if
  end for
  if k ≠ n/2 then
      return False
  end if
  return True
  ```

- Half-Circle es NP-hard: La reducción la haremos desde Hamiltoniano. Dado un grafo $G(V, E)$ buscamos construir $G'(V', E')$ tal que G tiene un ciclo Hamiltoniano si y sólo si G' tiene un círculo de tamaño $\frac{V}{2}$. La construcción consiste en conservar G' idéntico a G a excepción de la adición de $|V|$ vértices aislados (un vértice aislado es un vértice que no está conectado ningún otro).

  Es claro que esta reducción es correcta. Por un lado, si G tiene un ciclo Hamiltoniano C, por construcción el grafo G' tiene un círculo que recorre exactamente la mitad de los vértices en V'. Por otro lado, si G' tiene un círculo de tamaño $\frac{V'}{2}$, este ciclo no incluye ninguno de los vértices aislados que añadimos. Dado que incluye la mitad de los vértices en V', debe incluir todos los vértices en V. Por lo tanto, el ciclo C' es un ciclo Hamiltoniano en el grafo G.

  Finalmente, notemos que la construcción de G a G' toma en tiempo polinomial, de lo que concluimos que Hamiltoniano $\propto$ Half-Circle.

Pauta

- 1.5 pts. por mostrar que Half-Circle está en NP.
- 3 pts. por dar reducción polinomial.
- 1.5 pts. por mostrar que la reducción es correcta.
- Puntajes intermedios a criterio del corrector.

Pregunta 6

Definimos 3-SatSearch como el problema de buscar y retornar una valuación que satisfaga a una fórmula φ en 3CNF. Demuestre que si 3-Sat puede ser resuelto en tiempo polinomial, entonces 3-SatSearch puede ser resuelto en tiempo polinomial.

Hint: Recuerde que usted puede forzar el valor de una variable utilizando variables auxiliares como en la reducción de Sat $\propto$ 3-Sat.

Solución

Supongamos que existe una función 3Sat(φ) que responde que sí en tiempo polinomial si φ en 3CNF es satisfacible y no en caso contrario. Lo que debemos buscar es un algoritmo que resuelve 3SatSearch en tiempo polinomial. Dicho algoritmo debe tomar como *input* una fórmula φ en 3CNF de n variables y k cláusulas, y debe retornar una asignación para cada variable $x_1, \ldots, x_n$ que satisfaga φ ó que φ es no satisfacible.

Notemos que es posible forzar el valor de una variable proposicional x_i de la fórmula φ, transformando φ en $\varphi' \wedge \varphi'_{i,v}$ de la siguiente manera:

- Para forzar $x_i = 1 \Rightarrow \varphi'_{i,1} = (x_i \vee y \vee z) \wedge (x_i \vee \bar{y} \vee \bar{z}) \wedge (x_i \vee \bar{y} \vee z) \wedge (x_i \vee y \vee \bar{z})$
- Para forzar $x_i = 0 \Rightarrow \varphi'_{i,0} = (\bar{x_i} \vee y \vee z) \wedge (\bar{x_i} \vee \bar{y} \vee \bar{z}) \wedge (\bar{x_i} \vee \bar{y} \vee z) \wedge (\bar{x_i} \vee y \vee \bar{z})$

donde z e y son variables auxiliares, luego el algoritmo que buscamos es el siguiente:

Algoritmo: 3SatSearch(φ)

if 3Sat(φ) = False **then**
    return φ no es satisfacible
else
    $\varphi' \leftarrow \varphi$
    for $i = 1$ to n **do**
        if 3Sat($\varphi' \wedge \varphi'_{i,1}$) = True **then**
            $\varphi' \leftarrow \varphi' \wedge \varphi'_{i,1}$
            $x_i \leftarrow 1$
        else
            $\varphi' \leftarrow \varphi' \wedge \varphi'_{i,0}$
            $x_i \leftarrow 0$
        end if
    end for
    return $(x_1, x_2, ..., x_n)$
end if

Justificación de ejecución polinomial: El algoritmo de arriba resuelve el *loop* n veces. El loop consiste en un número constante de operaciones y llamar al algoritmo 3Sat el cual corre en tiempo polinomial. Tras todas las iteraciones, el *input* φ crece de φ (con n variables y k cláusulas) a una

fórmula con $n+2$ variables y $k+4n$ cláusulas. Finalmente ya que el tamaño del input está acotado polinomialmente y el algoritmo corre en tiempo polinomial, concluimos que 3SatSearch puede ser resuelto en tiempo polinomial.

Pauta

- 4 pts. por función correcta.
- 2 pts. por argumentar complejidad de la solución.
- Puntajes intermedios a criterio del corrector.