Importante. Esta prova consta de 4 páginas. Se num qualquer problema efectuar uma redução (polinomial ou não) descreva claramente a função de transformação associada à redução. Abreviaturas: $\uparrow(x)$ – função não definida em nenhum ponto, NPC–“completo em NP”, e co-NPC–“completo em co-NP” r.e.–“recursivamente enumerável”, dec–“decidível”... Os problemas que pode utilizar como sendo NPC são: 3SAT, VC, Clique, HC (ciclo hamiltoniano), PART e 3DM. Sabe-se que problemas como 2SAT, 2COL e EC pertencem à classe P.

1. As seguintes afirmações são verdadeiras ou falsas?

Nota importante. Escreva V ou F dentro do quadrado respectivo. Cada quadrado: 100% se resposta certa, 0% se não responde, -100% se resposta errada.

- V Qualquer que seja o inteiro k positivo, a solução da recorrência $x_0 = 1, x_{n+1} = 3x_n + 1$ não é de ordem $O(n^k)$.
- F Para toda a função $f : \mathbb{N} \rightarrow \mathbb{R}^+ : f(n) \in O(n^2)$ implica $f(n) \in \Theta(n^2)$.
- V Para todas as funções $f, g : \mathbb{N} \rightarrow \mathbb{R}^+ : f(n) \in O(g(n))$ se e só se $g(n) \in \Omega(f(n))$.
- F $\{i : \{i\}(x) \text{ não pára para nenhum valor de } x\}$ é uma linguagem r.e.
- F $\{i : \{i\}(x) \text{ pára para todos os valores de } x\}$ é uma linguagem r.e.
- F Há funções cuja computação com dados 1 termina em menos de 100 passos e outras em que isso não acontece. Usando o Teorema de Rice, conclui-se que que é indecidível, dado i , determinar se a computação $\{i\}(1)$ termina em menos de 100 passos.
- V Com base num oráculo para o problema geral da paragem, qualquer problema semi-decidível ou complementar de semi-decidível tem algoritmo de decisão.
- V Para um problema de decisão Π descobriu-se um algoritmo cujo tempo de execução (no pior caso) é determinado pela seguinte sequência: $t(0) = 1, t(n+1) = t(n) + 2n + 1$ onde n é o comprimento dos dados. Como $t(n)$ é polinomial, conclui-se que Π pertence à classe P.
- V Toda a linguagem de Σ_2^P é recursiva.
- F O problema: “dado um grafo não dirigido G , existe um ciclo simples que passa exactamente por 10 vértices de G ?” é completo em NP (supõe-se $P \neq NP$).
- V A existência de uma redução polinomial $SAT \leq_p 2SAT$ implicaria $P=NP$.
- V Seja A um problema de decisão pertence à classe NPC; se se descobrisse um algoritmo polinomial para A , passariam a existir algoritmos polinomiais para todos os problemas de $NP \cup co-NP$.

2. Análise de algoritmos

- (a) Averigue se $3^n \in O(2^n)$. Sim ou não: NÃO Demonstração:

Se fosse, $\exists n_0 \in \mathbb{N}, \exists k \in \mathbb{R}^+, \forall n \geq n_0 : 3^n \leq k2^n$ ou seja, $(1.5)^n \leq k$ o que é absurdo porque, para $a > 1, \lim_{n \rightarrow +\infty} a^n = +\infty$.

- (b) O tempo de execução de um algoritmo de ordenação é determinado (para argumentos que sejam potências de 2) pela seguinte recorrência

$$\begin{cases} t(1) = 0 \\ t(2n) = 2t(n) + 4n \quad (\text{para } n \geq 1) \end{cases}$$

Resolva a recorrência pelo método "tabelar/suspeitar/demonstrar". Apresente apenas a suspeita e a demonstração.

Sugestão. Pode ter interesse tabelar também os valores de $\log n$.

Suspeita: $t(n) = 2n \log n$

Prova por indução (para valores de n que sejam potências de 2):

- Caso base: para $n = 0$ vem $t(0) = 0$ tanto da recorrência como da suspeita.
- Vamos mostrar que $[t(n) = 2n \log n] \Rightarrow [t(2n) = 4n \log(2n)]$.

$$\begin{aligned} t(2n) &= 2t(n) + 4n && \text{(da recorrência)} \\ &= 2(2n \log n) + 4n && \text{(da hipótese indutiva)} \\ &= 4n(\log n + 1) \\ &= 4n(\log n + \log 2) \\ &= 4n \log 2n \end{aligned}$$

3. Computabilidade para complexidade

- (a) Mostre que PAP (problema da auto-paragem) não é decidível. A demonstração deve ser directa (não se baseando em qualquer redução).

Se existisse um método de decisão para PAP, existia, pela tese de Church-Turing, uma máquina de Turing M tal que

$$M(i) = \begin{cases} 1 & \text{se } \{i\}(i) \downarrow \\ 0 & \text{se } \{i\}(i) \uparrow \end{cases}$$

A máquina M pode ser facilmente transformada (como?) numa máquina M' tal que

$$M'(i) = \begin{cases} \uparrow & \text{se } \{i\}(i) \downarrow \\ 0 & \text{se } \{i\}(i) \uparrow \end{cases}$$

Seja m o índice de M' ; $M'(x)$ representa o mesmo que $\{m\}(x)$. Assim, e considerando em particular os dados $i = m$, temos

$$M'(m) = \{m\}(m) = \begin{cases} \uparrow & \text{se } \{m\}(m) \downarrow & \text{(a)} \\ 0 & \text{se } \{m\}(m) \uparrow & \text{(b)} \end{cases}$$

Temos uma contradição! Se a computação $\{m\}(m)$ pára (caso (a)), pela definição de M' é $\{m\}(m) = \uparrow$, isto é, não pára. E, se a computação $\{m\}(m)$ não pára (caso (b)), pela definição de M' é $\{m\}(m) = 0$, isto é, pára.

Esta contradição só pode resultar de termos admitido a existência de um algoritmo de decisão para PAP; logo tal algoritmo não existe.

Nota. Vêem-se frequentemente argumentos da seguinte forma

se $\{i\}(i)$ pára, podemos sabê-lo se esperarmos o tempo suficiente; mas se $\{i\}(i)$ não pára, teríamos que esperar um tempo infinito, o que não é possível!

Estes argumentos nada provam!

- (b) " L_{PGP} , a linguagem associada ao problema geral da paragem é completa na classe das linguagens r.e. relativamente à redução $\leq$ ". Explique o significado desta afirmação.

Significa que

- L_{PGP} pertence à classe das linguagens r.e. e que...
- qualquer que seja a linguagem L da classe r.e. existe uma redução $L \leq L_{\text{PGP}}$.

- (c) Considere o seguinte problema de decisão Γ : dado $i \in \mathbb{N}$, a função $\{i\}(n)$ está definida para no máximo 10 valores de n ? (por exemplo, se $\{i\}(n) \equiv \uparrow(n)$, então $i \in L_\Gamma$). A que classe pertence este problema? Responda I, II, III ou IV, sendo I: “dec”, II: “semi-dec mas não dec”, III: “co-semi-dec mas não dec”, IV: “outra”? III Demonstração:

Consideremos a linguagem $\overline{L_\Gamma}$. É fácil verificar que esta linguagem pertence à classe r.e. Na verdade, é possível simular “em paralelo” pela técnica de “dovetailing” as computações $\{i\}(x)$ para todos os x ; quando e se houver 11 paragens das computações simuladas, para-se a simulação global com resultado 1 (se isso acontecer, $i \in \overline{L_\Gamma}$, $i \notin L_\Gamma$).

Logo $\overline{L_\Gamma}$ pertence ou à classe III ou à classe I.

Para mostrar agora que L_Γ pertence à classe III basta mostrar que L_Γ não é decidível, isto é, que não pertence à classe I. Para esse efeito podemos usar o Teorema de Rice ou uma redução apropriada. Vamos escolher esta última alternativa e descrever uma redução

$$\overline{L_{\text{PAP}}} \leq L_\Gamma$$

Seja i a instância de $\overline{L_{\text{PAP}}}$. A sua transformada é a máquina de Turing seguinte, de índice $f(i)$ e dados x com o comportamento seguinte

- Apaga a fita.
- Escreve i na fita.
- Simula $\{i\}(i)$
- Se $i \in \overline{L_{\text{PAP}}}$, $\{i\}(i) \uparrow$ de modo que $\{f(i)\}(x)$ não pára para qualquer valor de x , logo $\{f(i)\} \in L_\Gamma$ (está definida para 0 valores e $0 \leq 10$).
- Se $i \notin \overline{L_{\text{PAP}}}$, $\{i\}(i) \downarrow$ de modo que $\{f(i)\}(x)$ pára para qualquer valor de x , logo $\{f(i)\} \notin L_\Gamma$ (está definida para uma infinidade de valores), o que é falso.

Nota. É impossível uma redução $L_{\text{PAP}} \leq L_\Gamma$, pois $\Rightarrow L_{\text{PAP}} \in \text{co-r.e.} \Rightarrow L_{\text{PAP}} \in \text{rec.}$ (pois é sabido que $L_{\text{PAP}} \in \text{r.e.}$).

4. Teoria dos problemas completos em NP

(a) Verdadeiro ou falso?

- i. Considere a seguinte relação binária R entre problemas de decisão A e B : ARB sse existe uma redução polinomial $A \propto B$. “A relação binária R é anti-simétrica, isto é, ARB e BRA implica $A=B$ ”. Verdadeiro ou falso? falso Justificação:

Consideremos 2 problemas distintos da classe NPC, por exemplo 3SAT e HC. Por definição de “problema completo em NP” verifica-se: $3\text{SAT} \propto \text{HC}$ e $\text{HC} \propto 3\text{SAT}$, sendo 3SAT e HC problemas distintos.

- ii. "Se Π é um problema de decisão da classe NPC e A um problema de decisão da classe P, existe uma redução polinomial $A \leq \Pi$ ". Supõe-se que $P \neq NP$. Verdadeiro ou falso? verdadeiro Justificação:

Se $A \in P$, é também $A \in NP$. Como $\Pi \in NPC$, qualquer problema de NP se reduz a Π , em particular $A \leq \Pi$.

- (b) Considere o seguinte problema (Φ): INSTÂNCIA: (U, C) como em SAT (cada cláusula pode ter um número arbitrário de literais). PERGUNTA: Existe uma valorização lógica das variáveis de U que satisfaça todas as cláusulas excepto 2 (isto é, que satisfaça exactamente $|C| - 2$ cláusulas)? Mostre que o problema Φ pertence à classe NPC. Qual a redução que vai efectuar? $3SAT \leq \Phi$

É fácil verificar que $\Phi \in NP$. Na verdade, dada uma instância (U, C) do problema Φ e dada uma palavra y é possível determinar em tempo polinomial se y representa uma valorização lógica das variáveis de U e se há exactamente $|C| - 2$ cláusulas de C que ficam satisfeitas com essa valorização. A seguinte redução mostra que $\Phi \in NPC$.

$$\begin{aligned} 3SAT &\rightarrow \Phi \\ (U, C) &\rightarrow (U', C') \end{aligned}$$

onde $U' = U \cup \{u, v\}$ e $C' = C \cup \{(u), (\bar{u}), (v), (\bar{v})\}$, sendo u e v variáveis novas (não pertencentes a U). Note-se que, qualquer que seja a valorização lógica das variáveis de U' , de entre as cláusulas (u) , $(\bar{u})$, (v) e $(\bar{v})$ há exactamente 2 não satisfeitas.

É fácil ver que se trata de uma redução polinomial. Em primeiro lugar, é óbvio que a transformação pode ser efectuada em tempo polinomial.

→ Se $(U, C) \in L_{3SAT}$, existe uma valorização lógica das variáveis de U que torna todas as cláusulas de C satisfeitas; essa valorização, estendida arbitrariamente a $\{u, v\}$ (por exemplo, com os valores lógicos de u e de v ambos iguais a verdade) torna $|C'| - 2$ cláusulas de C' satisfeitas.

← Se $(U, C) \notin L_{3SAT}$, nenhuma valorização lógica das variáveis de U torna todas as cláusulas de C satisfeitas; isso implica que para qualquer valorização lógica das variáveis de U' , o número de cláusulas de C' satisfeitas é menor que $|C'| - 2$; logo, $(U', C') \notin \Phi$.