Fondamenti di Informatica II – CdL in Ingegneria Informatica VV. O.

Compito, 15/9/2009

Nome, Cognome, Matricola: ______________________________

Corso di studio:______________________________

Prima Parte: domande teoriche

Si forniscano risposte il più possibile **rigorose e succinte** ai seguenti quesiti riguardanti argomenti trattati nel corso. Risposte approssimative, prolisse o in cattivo italiano saranno **fortemente penalizzate**. Ai fini del superamento dell'esame, è necessario conseguire una valutazione sufficiente alla prima parte.

T1 Data una generica contrazione $f(n)$, si dia la definizione formale della relativa funzione $f^\star(n, n_0)$, per $n > n_0$. Si determini inoltre $f^\star(n, 2)$ quando $f(n) = \sqrt{n}$ e $n = 2^{2^i}$, per $i > 0$.

Solution: Define $f^{(0)}(n) = n$, and, for $k > 0$, $f^{(k)}(n) = f(f^{(k-1)}(n))$. Then

$$f^\star(n, n_0) = \max\{k \geq 0 : f^{(k)}(n) > n_0\}.$$

When $f(n) = \sqrt{n} = n^{1/2}$, clearly we have $f^{(k)}(n) = n^{1/2^k}$, for $k \geq 0$. Therefore

$$f^{(k)}(n) > 2 \Leftrightarrow n^{1/2^k} > 2 \Leftrightarrow (\log_2 n)/2^k > 1 \Leftrightarrow 2^k < \log_2 n \Leftrightarrow k < \log_2 \log_2 n$$

Observe that $\log_2 \log_2 n$ is an integer when $n = 2^{2^i}$, hence $f^\star(n, 2) = \log_2 \log_2 n - 1$.

T2 Si enunci e si provi il lemma di cancellazione per le radici n-esime dell'unità nel campo complesso.

Solution: The cancellation lemma states that for any integers $n, d \geq 1$ and $k \geq 0$,

$$\omega_{dn}^{dk} = \omega_n^k$$

Proof.

$$\begin{aligned} \omega_{dn}^{dk} &= \left(e^{2\pi i/dn}\right)^{dk} \\ &= \left(e^{2\pi i/n}\right)^{k} \\ &= \omega_n^k. \end{aligned}$$

T3 Dimostrare che:

$\langle G = (V, E), k \rangle \in$ VERTEX COVER $\Leftrightarrow \langle G^c = (V, E^c), |V| - k \rangle \in$ CLIQUE.

Solution:

$\langle G = (V, E), k \rangle \in$ VERTEX COVER

$\Leftrightarrow \quad \exists V' \subseteq V, |V'| = k : \forall u, v \in V, u \neq v : \{u, v\} \in E \Rightarrow [(u \in V') \vee (v \in V')]$

$\Leftrightarrow \quad \exists V'' \subseteq V, |V''| = |V| - k : \forall u, v \in V, u \neq v : \{u, v\} \notin E^c \Rightarrow [(u \notin V'') \vee (v \notin V'')]$

$\Leftrightarrow \quad \exists V'' \subseteq V, |V''| = |V| - k : \forall u, v \in V, u \neq v : [(u \in V'') \wedge (v \in V'')] \Rightarrow \{u, v\} \in E^c$

$\langle G = (V, E), |V| - k \rangle \in$ CLIQUE

The above derivation is obtained by introducing the new quantified variable $V'' = V - V'$, and by observing that $|V''| = |V| - |V'|$, $\{u, v\} \in E \Leftrightarrow \{u, v\} \notin E^c$ and $(u \in V') \Leftrightarrow (u \notin V'')$:

Seconda Parte: risoluzione di problemi

Si forniscano soluzioni esaurienti e rigorose ai tre problemi seguenti. Gli algoritmi vanno codificati utilizzando lo **pseudocodice** usato in classe. **Attenzione:** Risposte immotivate e prive di prova di correttezza non saranno considerate.

Esercizio 1 [10 punti] Si consideri la seguente ricorrenza, definita per tutti i valori del parametro intero $n > 0$.

$$T(n) = \begin{cases} 18 & 0 < n < 20 \\ T\left(\left\lceil \frac{3n}{10} + 3 \right\rceil\right) + T\left(\left\lfloor \frac{2n}{5} - 1 \right\rfloor\right) + 3n & n \geq 20 \end{cases}$$

Utilizzando l'induzione parametrica, si determini una opportuna costante $c > 0$ tale che $T(n) \leq cn$ per $n > 0$.

Answer: Parametric induction lets us collect constraints on feasible values $c > 0$ by "simulating" an inductive argument. For the base cases $0 < n < 20$, observe that it must be $18 = T(n) \leq cn$, whence $c \geq 18/n$. The strictest constraint is the one for $n = 1$, which yields $c \geq 18$. Assume now that the statement is true for $T(k)$, with $0 < k < n$ and $n \geq 20$. We have:

$$\begin{aligned} T(n) &= T(\lceil (3n/10) + 3 \rceil) + T(\lfloor (2n/5) - 1 \rfloor) + 3n \\ &\leq c\lceil (3n/10) + 3 \rceil + c\lfloor (2n/5) - 1 \rfloor + 3n \\ &\leq c(3n/10 + 4) + c(2n/5 - 1) + 3n \\ &\leq c(7/10)n + 3c + 3n. \end{aligned}$$

In order to obtain the inductive thesis, it suffices to pick a single value $c > 0$ such that $c(7/10)n+3c+3n \leq cn$, for all values $n \geq 20$, or, equivalently, $c(3/10)n-3c \geq 3n$. Consider the family of straight lines $y_c = c(3/10)x - 3c$ and the straight line $y = 3x$. For the inequality to be true for all values $n \geq 20$, it suffices to pick any value c corresponding to a straight line y_c in the family which intersects $y = 3x$ at an abscyssa $x_{\text{in}} \leq 20$. This yields the simple inequality $3c \geq 60$, whence $c \geq 20$.

In conclusion, the whole inductive argument carries through by picking $c = \max\{18, 20\} = 20$, whence $T(n) \leq 20n$, for $n > 0$. □

Esercizio 2 [12 punti] Dato un array $A[1..n]$ di numeri interi *arbitrari* (sia positivi che negativi), si consideri il problema di determinare il valore

$$m^\star = \max \left\{ \sum_{k=i}^{j} A[k] : 1 \leq i \leq j \leq n \right\}.$$

Descrivere e analizzare algoritmo di programmazione dinamica FIND_MAX(A) che ritorni il valore $m^\star$ in tempo $O(n)$.

Suggerimento: Dimostrare che la soluzione è ottenibile calcolando, *per ogni* i, $1 \leq i \leq n$, il valore massimo $m(i)$ ottenibile sommando gli elementi dell'array in un qualche intervallo di indici che abbia i come estremo inferiore. La programmazione dinamica va applicata al calcolo di tali valori $m(i)$...

Answer: For $1 \leq i \leq n$, define

$$m(i) = \max \left\{ \sum_{k=i}^{j} A[k] : i \leq j \leq n \right\}.$$

Clearly, $m(i)$ represents the largest sum obtainable by summing elements of A with indices starting from i. Define now

$$\hat{m} = \max \{ m(i) : 1 \leq i \leq n \}.$$

We have that $\hat{m} \leq m^\star$, since the set of values $\{m(i) : 1 \leq i \leq n\}$ is contained in the set $\left\{ \sum_{k=i}^{j} A[k] : 1 \leq i \leq j \leq n \right\}$. But it is also $m^\star \leq \hat{m}$, since for each $1 \leq i \leq j \leq n$, we have $\sum_{k=i}^{j} A[k] \leq m(i)$. Therefore we have that $m^\star = \hat{m}$.

The values $m(i)$ can be computed according to the following, simple recurrence:

$$m(i) = \begin{cases} A[n] & i = n. \\ \max\{A[i], A[i] + m(i+1)\} & \text{otherwise.} \end{cases}$$

Let us show that the recurrence is valid using (backward) induction. The statement clearly holds for $i = n$, since $[n, n]$ is the only interval of indices starting from n. Assume the statement holds down to $i + 1$. Consider now the largest sum obtainable with indices starting from i. Such sum is either $A[i]$ or, otherwise, it must be $A[i] + m[i + 1]$, since it can be neither smaller ($A[i] + m[i + 1]$ is a sum of indices starting from i) or larger (or otherwise $m[i + i]$ would not be maximum). A simple, linear algorithm to solve the problem follows. In the algorithm, subroutine MAX(a, b) returns $\max\{a, b\}$.

```
FIND_MAX(A)
n ← length(A)
m[n] ← m̂ ← A[n]
for i ← n − 1 downto 1 do
    m[i] ← MAX(A[i], A[i] + m[i + 1])
    m̂ ← MAX(m̂, m[i])
return m̂
```

$\square$

Esercizio 3 [10 punti] Si consideri il seguente problema decisionale:

PARTITION:

INSTANCE: $\langle S \rangle$, $S \subseteq \boldsymbol{N}$, insieme finito

QUESTION: Esistono $S_1, S_2 \subseteq S$, con $S_1 \cup S_2 = S$ e $S_1 \cap S_2 = \emptyset$ tali che

$$\sum_{s_1 \in S_1} s_1 = \sum_{s_2 \in S_2} s_2 \quad ?$$

Si dimostri che PARTITION $<_P$ SUBSET SUM.

Answer: Recall that an instance of SUBSET SUM is $\langle S, t \rangle$, with $S \subset \boldsymbol{N}$ and $t \in \boldsymbol{N}$. Given an instance $\langle S \rangle$ of PARTITION, we set $a = \sum_{s \in S} s$ and define our reduction function as follows:q

$$f(\langle S \rangle) = \begin{cases} \langle \emptyset, 1 \rangle & \text{if } a \text{ is odd,} \\ \langle S, a/2 \rangle & \text{otherwise.} \end{cases}$$

Function f is clearly computable in polynomial time. Let us now show that f indeed reduces PARTITION to SUBSET SUM.

Let $x = \langle S \rangle \in$ PARTITION. Then, there exists a partition of S into subsets S_1 and S_2 such that $\sum_{s_1 \in S_1} s_1 = \sum_{s_2 \in S_2}$. Let $b = \sum_{s_1 \in S_1} s_1$. Then $a = \sum_{s \in S} s = 2b$ is an even number and $f(x) = \langle S, a/2 \rangle \in$ SUBSET SUM, since there exists a subset of S (either S_1 or S_2) whose elements sum to $a/2$.

Viceversa, let $f(x) \in$ SUBSET SUM. Then, it must be $f(x) = \langle S, a/2 \rangle$ necessarily (since $f(x)$ is a positive instance). Also, there is a subset $S_1 \subset S$ such that $\sum_{s_1 \in S_1} s_1 = a/2$, whence $\sum_{s_2 \in S - S_1} s_2 = a - a/2 = a/2$, which implies that $x = \langle S \rangle \in$ PARTITION. $\square$