Iniziato	Friday, 18 June 2021, 13:22
Stato	Completato
Terminato	Friday, 18 June 2021, 13:37
Tempo impiegato	15 min. 1 secondo

Domanda **1**

Risposta corretta

Punteggio max.: 2,00

Let f, g be the functions defined as $f(n) = 10^{-5}n^{10}\log n$ and $g(n) = \frac{n^{10}}{\log n}$.

Scegli una o più alternative:

- [ ] $f \in O(g)$
- [ ] $f \in \Theta(g)$
- [x] $f \in \Omega(g)$ ✔

Your answer is correct.

Domanda **2**

Risposta corretta

Punteggio max.: 2,00

Turing Machines:

Scegli una o più alternative:

- [x] Are such that certain problems that can be computed in time 2^n cannot be computed in time n^2 ✔
- [x] Can work on alphabets of arbitrary but fixed size ✔
- [ ] All work in time bounded by a computable function
- [ ] All work in time bounded by a polynomial function

Your answer is correct.

Domanda **3**

Parzialmente corretta

Punteggio max.: 2,00

The problem of deciding if a 3CNF is satisfiable is

Scegli una o più alternative:

- [ ] In the class $\mathbf{P}$
- [x] In the class $\mathbf{EXP}$ ✔
- [x] Undecidable, but $\mathbf{NP}$-hard. ✘
- [x] $\mathbf{NP}$-complete ✔

Your answer is partially correct.

Hai selezionato troppe opzioni.

Domanda **4**

Risposta corretta

Punteggio max.: 2,00

Suppose a language $\mathcal{L}$ is both undecidable and not in $\mathbf{NP}$. Then:

Scegli una o più alternative:

- [ ] The classes $\mathbf{P}$ and $\mathbf{NP}$ are necessarily equal.
- [ ] The classes $\mathbf{P}$ and $\mathbf{NP}$ are necessarily different.
- [x] The language $\mathcal{L}$ is strictly harder to decide than $\mathtt{3SAT}$ ✔
- [x] There cannot be any deterministic Turing machine deciding $\mathcal{L}$ ✔

Your answer is correct.

Domanda **5**

Risposta corretta

Punteggio max.: 2,00

The notion of PAC-learnable concept class:

Scegli una o più alternative:

- [ ] Cannot be reached when the underlying concept class is a conjunctions of literals.
- [x] Is such that the accuracy and confidence errors are taken into account independently. ✔
- [x] Implicitly refers to the underlying representation class. ✔
- [ ] Is such that the output concept needs to have probability of error ε, in all possible cases

Your answer is correct.

Domanda **1**

Completo

Punteggio max.: 6,00

Construct a deterministic TM of the kind you prefer, which decides the following language:

$$\mathcal{L} = \{w \in \{0,1\}^* \mid \text{between any pair of occurrences of } 0 \text{ in } w \text{ there are an o}$$

Study the complexity of TM you have defined.

1 tape TM

The alphabet: A ={start, 0, 1, blank}
The set od states: Q ={qinit, q1, q2, q3, q4, qf, qhalt}
The transition function: delta=
(qinit, start) -> (q1, start, S)
(q1, start) -> (q1, start, R)
(q1, 0) -> (q2, 0, R)
(q1, 1) -> (q1, 1, R)
(q2, 0) -> (q2, 0, R)
(q2, 1) -> (q3, 1 ,R)
(q3, 0) -> (q2, 0, R)
(q3, 1) -> (q4, 1, R)
(q4, 1) -> (q3, 1, R)
(q1, blank) -> (qf, S)
(q2, blank) -> (qf, S)
(q3, blank) -> (qf, S)
(q4, blank) -> (qf, S)
(qf, blank) -> (qf, L)
(qf, 0) -> (qf, 0, L)
(qf, 1) -> (qf, 1, L)
(qf, start) -> (qhalt, start, S)

TM=(Q, A, delta)

We need to pass through the input string from left to right only once. This TM has linear time complexity.

Domanda **2**

Completo

Punteggio max.: 7,00

You are required to prove that the following problem $\mathcal{L}$ is in $\mathbf{NP}$. To do that, you can give a TM or define some pseudocode. The language $\mathcal{L}$ includes precisely those binary strings which are encodings of pairs in the form (G, k) where G is a graph and k is a natural number, such that the nodes of G can be assigned a natural number between 1 and k in such a way that nodes which are linked by an edge are assigned distinct natural numbers.

L = {(G, k) | Exist C.(number of differents values in C < k and (for every (v, u) that are edges C[v] != C[u]))}
Given that this is in the form 'certificate-verifier' what we need to do to prove that this problem is in NP is to show that the certificate has polinomial size and that M runs in polynomial time.
1. The certificate C has size bounded by the graph G and thus is polinomial
2. It's enough to check whether any pair have different value (given that they're connected by an edge). This can be done in quadratic time and thus M runs in polynomial time.

Domanda **3**

Completo

Punteggio max.: 7,00

Consider the following problem:

$$\texttt{1SAT} = \{A \mid A \text{ is a satisfiable 1CNF}\}.$$

To which complexity class does $\texttt{1SAT}$ belong? Prove your claim.

1SAT is in P.

Pseudocode:

```
l = lenght(A)
for i in 0..l
    for v in 0..l
        if (A[i][0] == A[v][0] and A[i][1] =! A[v][1])
           return false
return true
```

This pseudocode returns true only if A is satisfiable (not the result, but given whether A is satisfiable the solution is trivial).
Proof that 1SAT is in P, given the pseudocode.

1. The input can be easily encoded in a binary string. Each element of A can be represented as a pair of the variable and whether it's negated (variable#negation). With n differents variable log n + 1 bits are required, the negation needs 1 bit, we also need to add @ to separate the pairs. We map 0 to 00, 1 to 11, # to 10 and @ to 01. Is then easy to confirm that the total number of bits needed is polinomial.
2. The total number of executed instruction is 1 + 2*O(n^2) = O(n^2)
3. Each instruction can be simulated by a TM in polynomial time (for example a TM to do an equality check needs to go through the tapes only one time).
4. i and v can be at most big as the lenght of A and are thus bounded by a polynomial in l.