|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COGNOME:** |  | **NOME** |  | **B1** |
| **MATRICOLA:** |  | | |
| **DOCENTE:** |  | | |

|  |  |
| --- | --- |
| **Domanda 1** | Risultato |
| Siano dati i seguenti numeri rappresentati in codifica esadecimale:   * N1: A3 * N2: 3F   Dopo averli rappresentati in codifica binaria, eseguire la loro somma considerandoli due numeri in CA2 su 8 bit e verificando eventualmente la presenza di overflow | Risultato somma:  Overflow (si/no): |
| Si riportino TUTTI i passaggi | |

|  |  |
| --- | --- |
| **Domanda 2** | Risultato |
| Calcolare la tavola di verità della seguente funzione booleana (Z and (X and not(Y))) or (not(X) and Z) | |
| Si riportino tutti i passaggi significativi | |

|  |
| --- |
| **Domanda 3** |
| Quali sono le affinità e le differenze tra la memoria RAM e la memoria ROM? |
|  |

|  |
| --- |
| **Domanda 4 (PROGRAMMAZIONE)** |

Un algoritmo di classificazione è un algoritmo di machine learning in grado di assegnare un’etichetta a un oggetto sulla base della conoscenza di come sono fatti altri oggetti simili. Si scriva un programma C in grado di implementare l’algoritmo di classificazione 1-NN.

Il programma riceve in ingresso due file <train> e <test> i cui nomi sono passati come argomento sulla linea di comando:

<train> - è un file contenente un insieme di N oggetti (uno per riga) con la relativa etichetta. Questo file è utilizzato per insegnare al classificatore a riconoscere gli oggetti ignoti. Ogni riga del file contiene il nome dell’oggetto rappresentato come una stringa lunga al massimo 5 caratteri, un insieme di M numeri reali F1 F2 F3 … FM seguiti da un numero intero L. Il carattere spazio è usato per separare i vari elementi. Gli M numeri reali sono la rappresentazione di un oggetto mentre il numero intero L rappresenta l’etichetta dell’oggetto.

<test> - è un file contenente un insieme di oggetti da classificare. Il formato è analogo a quello del file <train> ma il numero di righe **NON E’ NOTO A PRIORI.**

Si assuma che i valori di M ed N siano noti a priori e definiti nel codice del programma tramite una direttiva define.

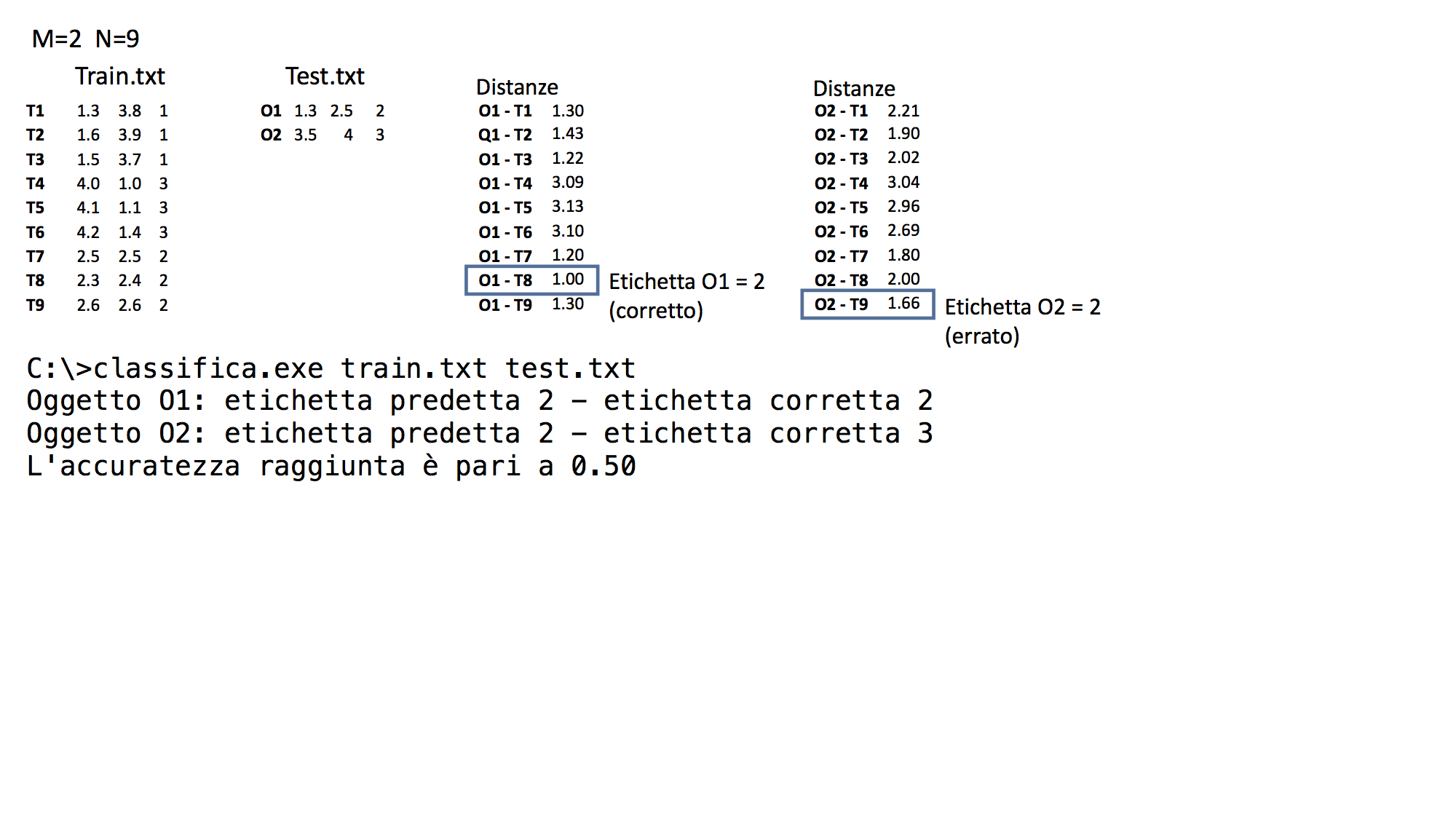
Il programma deve predire l’etichetta di ogni oggetto O contenuto nel file <test> utilizzando il seguente criterio:

* Calcolare la distanza euclidea tra O e ogni oggetto T contenuto nel file <train>.
* L’etichetta assegnata a O è l’etichetta dell’oggetto in T più vicino a O.
* La distanza euclidea tra due oggetti O e T si calcola come dove e sono i numeri reali utilizzati per rappresentare gli oggetti.

Per ogni classificazione effettuata il programma deve stampare sullo schermo un messaggio che indichi il nome dell’oggetto, l’etichetta predetta per l’oggetto e la relativa etichetta corretta (così come riportata nel file <test>).

Infine il programma deve stampare su schermo l’accuratezza del classificatore calcolata come il rapporto tra il numero di etichette predette correttamente diviso per il numero di oggetti classificati.

**Esempio:**



|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **COGNOME:** |  | **NOME** |  | **B2** |
| **MATRICOLA:** |  | | |
| **DOCENTE:** |  | | |

|  |  |
| --- | --- |
| **Domanda 1** | Risultato |
| Siano dati i seguenti numeri rappresentati in codifica esadecimale:   * N1: A3 * N2: 3F   Dopo averli rappresentati in codifica binaria, eseguire la loro somma considerandoli due numeri in binario puro su 8 bit verificando eventualmente la presenza di overflow | Risultato somma:  Overflow (si/no): |
| Si riportino TUTTI i passaggi | |

|  |  |
| --- | --- |
| **Domanda 2** | Risultato |
| Calcolare la tavola di verità della seguente funzione booleana: (not(Z) and (X and Y)) or (x and z) | |
| Si riportino tutti i passaggi significativi | |

|  |
| --- |
| **Domanda 3** |
| Descrivere brevemente la differenza tra memoria centrale e memoria di massa |
|  |

|  |
| --- |
| **Domanda 4 (PROGRAMMAZIONE)** |

Un algoritmo di classificazione è un algoritmo di machine learning in grado di assegnare un’etichetta a un oggetto sulla base della conoscenza di come sono fatti altri oggetti simili. Si scriva un programma C in grado di implementare l’algoritmo di classificazione 1-NN.

Il programma riceve in ingresso due file <train> e <test> i cui nomi sono passati come argomento sulla linea di comando:

<train> - è un file contenente un insieme di N oggetti (uno per riga) con la relativa etichetta. Questo file è utilizzato per insegnare al classificatore a riconoscere gli oggetti ignoti. Ogni riga del file contiene il nome dell’oggetto rappresentato come una stringa lunga al massimo 5 caratteri, un insieme di M numeri reali F1 F2 F3 … FM seguiti da un numero intero L. Il carattere spazio è usato per separare i vari elementi. Gli M numeri reali sono la rappresentazione di un oggetto mentre il numero intero L rappresenta l’etichetta dell’oggetto.

<test> - è un file contenente un insieme di oggetti da classificare. Il formato è analogo a quello del file <train> ma il numero di righe **NON E’ NOTO A PRIORI.**

Si assuma che i valori di M ed N siano noti a priori e definiti nel codice del programma tramite una direttiva define.

Il programma deve predire l’etichetta di ogni oggetto O contenuto nel file <test> utilizzando il seguente criterio:

* Calcolare la distanza di Manhattan tra O e ogni oggetto T contenuto nel file <train>.
* L’etichetta assegnata a O è l’etichetta dell’oggetto in T più vicino a O.
* La distanza di Manhattan tra due oggetti O e T si calcola come: dove e sono i numeri reali utilizzati per rappresentare gli oggetti.

Per ogni classificazione effettuata il programma deve stampare sullo schermo un messaggio che indichi l’etichetta predetta per l’oggetto e la relativa etichetta corretta (così come riportata nel file <test>).

Infine il programma deve stampare su schermo l’accuratezza del classificatore calcolata come il rapporto tra il numero di etichette predette correttamente diviso per il numero di oggetti classificati.

