

**Esercizio n. 1 (punti 7)**

Dato il seguente codice in linguaggio C, dove il tipo **list** rappresenta una lista di interi (non ordinata):

```
int f(int i, list L)
{ if (L!=NULL)
    {if (i==L->value) return 1+f(i,L->next);
    else return f(i, L->next);
    }
  else return 0;
}

int common(list A, list L)
{ int c=0;
  while (A!=NULL)
    {if (f(A->value, L)) c++;
    A=A->next;
    }
  return c;
}
```

- Si descriva cosa fanno la funzione **common** indicata e la funzione **f**.
- Si discuta la complessità del codice sapendo che la lista **A** contiene **M** elementi e la lista **L** contiene **N** elementi. Si individuino caso migliore e peggiore, se esistono, e si discuta dal punto di vista della complessità asintotica (in particolare andamento asintotico) del numero di esecuzioni complessive del test sottolineato nella funzione **f**.

**Esercizio n. 3 (punti 3)**

Descrivere il tipo di ereditarietà (singola, o multipla) supportato tra le diverse tipologie di componenti nel linguaggio Java, evidenziando le differenze e motivandole.