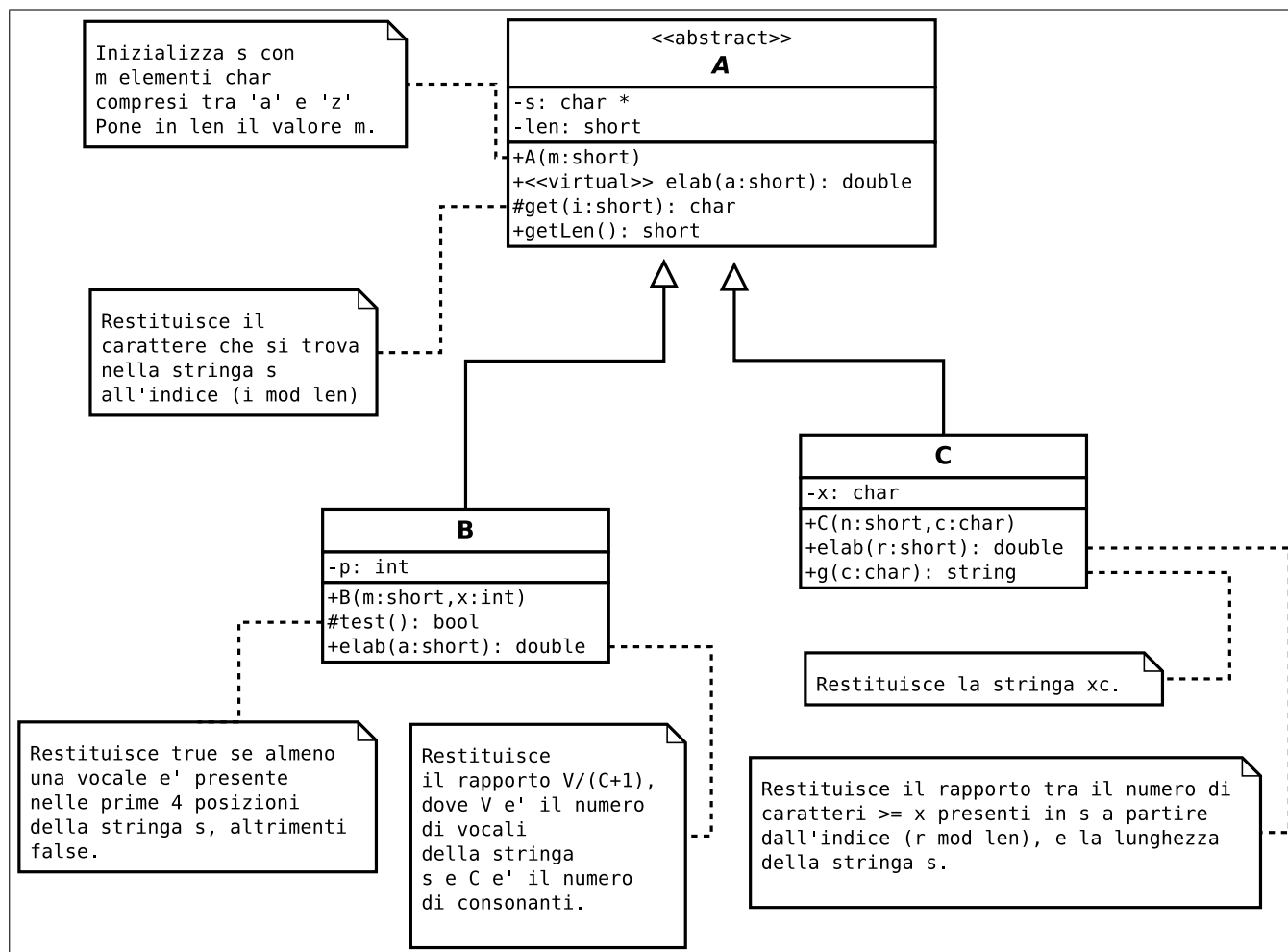


Si implementi in C++ la gerarchia ereditaria descritta dal seguente diagramma UML delle classi. NB: È necessario implementare tutti e soli i metodi indicati nel diagramma.



In un main, si generi una collezione di 50 oggetti utilizzando la sequenza di valori casuali riportata sul retro del foglio. **NB:** È inoltre possibile (e consigliato) prelevare il frammento di codice da inserire nel main a partire dalla URL indicata sul retro del foglio

Successivamente:

1. si visualizzi la collezione mediante l'overloading dell'operatore `<<`, ad esempio:

```

35)1C, s= mghbepovgc, x=f, elab(3)=0.4
36)1B, s= qrgqgf, p=4, elab(3)=0
37)1B, s= bw, p=5, elab(3)=0

```

2. si calcoli la media dei valori `elab(3)` per tutti gli oggetti della collezione e si visualizzi la stringa concatenazione di tutte le stringhe restituite da `g('h')` per tutti gli oggetti di tipo C;
3. si implementi l'overloading dello operatore membro postfixo `++` per la classe B e lo si utilizzi nel main per un qualche oggetto di tipo B.

```
srand(111222333);

for(int i=0; i<DIM; i++) {
    short n=1+rand()%10;
    if(rand()%2==0)
        vett[i]= new B(n, rand()%10 + 1);
    else
        vett[i]= new C (n, (char) (rand()%((int) 'z' - (int) 'a' + 1) + (int) 'a'));
}
```

1. Frame di codice da inserire nella funzione main:

www.dmi.unict.it/~messina/didat/03072019/frameB.cpp

oppure Short URL:

<https://tinyurl.com/y3jq64gf>

2. Output di controllo utenti Windows:

www.dmi.unict.it/~messina/didat/03072019/out_win_B.txt

oppure Short URL:

<https://tinyurl.com/y3tl7p4w>

3. Output di controllo utenti Linux:

www.dmi.unict.it/~messina/didat/03072019/out_linux_B.txt

oppure Short URL:

<https://tinyurl.com/yyjgcdxs>