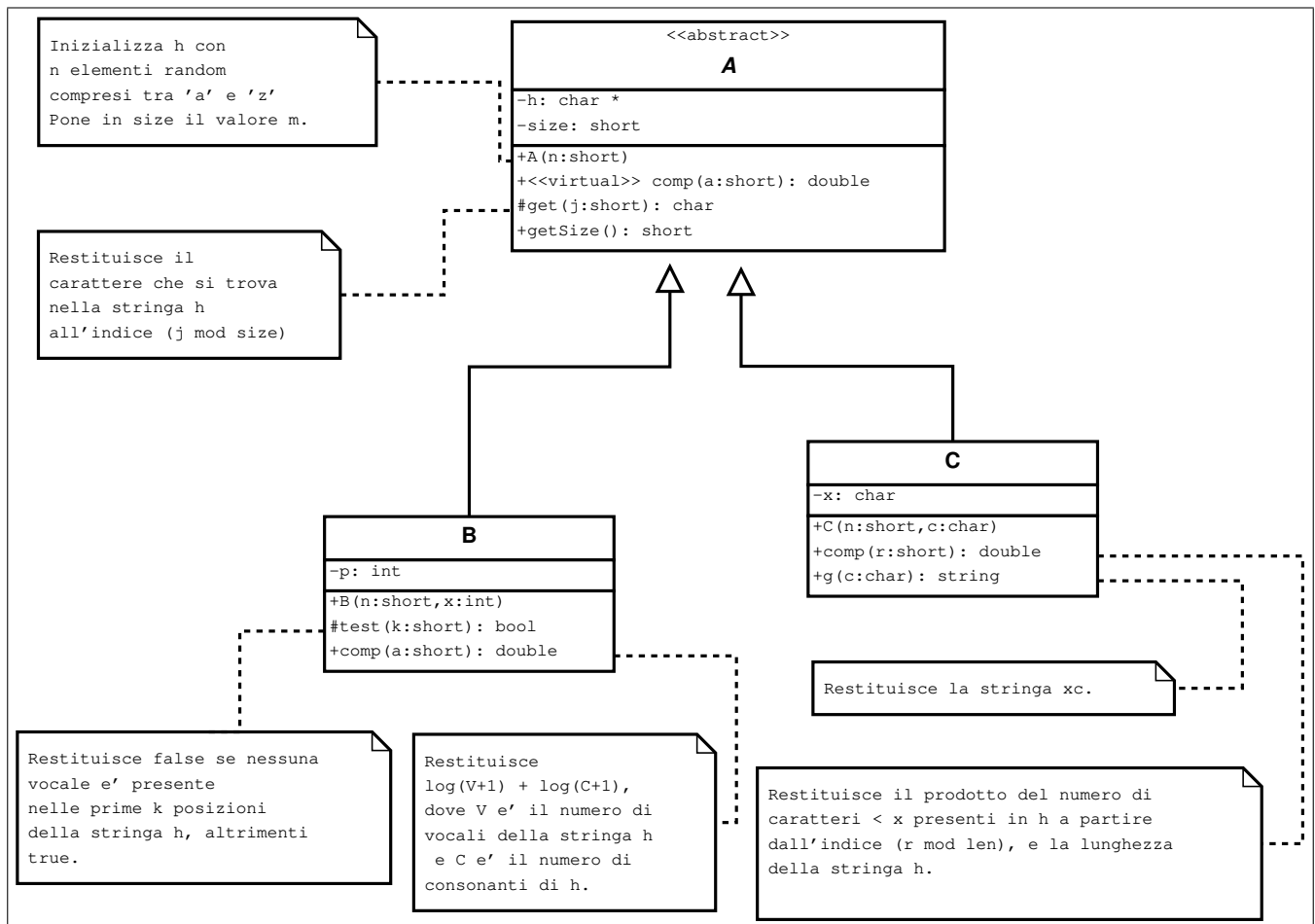


Università di Catania
Dipartimento di Matematica e Informatica
Corso di Studio in Informatica, A.A. 2019-2020
Prova di laboratorio di Programmazione I (9 CFU).
10 Febbraio 2020 - A

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:

```

24)1B, h= rkqbybjmz, p=3, comp(3)=2.30259
25)1C, h= nwgxsjoog, x=j, comp(3)=45
26)1B, h= fgwwgglr, p=2, comp(3)=2.19722

```

2. si calcoli la media dei valori `comp(3)` per tutti gli oggetti della collezione e si visualizzi la stringa concatenazione di tutte le stringhe restituite da `g('Z')` per tutti gli oggetti di tipo C;
3. si implementi l'overloading dello operatore membro prefisso `++` 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/~gfarinella/Prog1/10022020/frame-10_02_2020_A.cpp

oppure Short URL:

<http://tiny.cc/6keojz>

2. Output di controllo utenti Windows:

www.dmi.unict.it/~gfarinella/Prog1/10022020/out_windows_A.txt

oppure Short URL:

<http://tiny.cc/lmeojz>

3. Output di controllo utenti Linux:

www.dmi.unict.it/~gfarinella/Prog1/10022020/out_linux_A.txt

oppure Short URL:

<http://tiny.cc/pneojz>