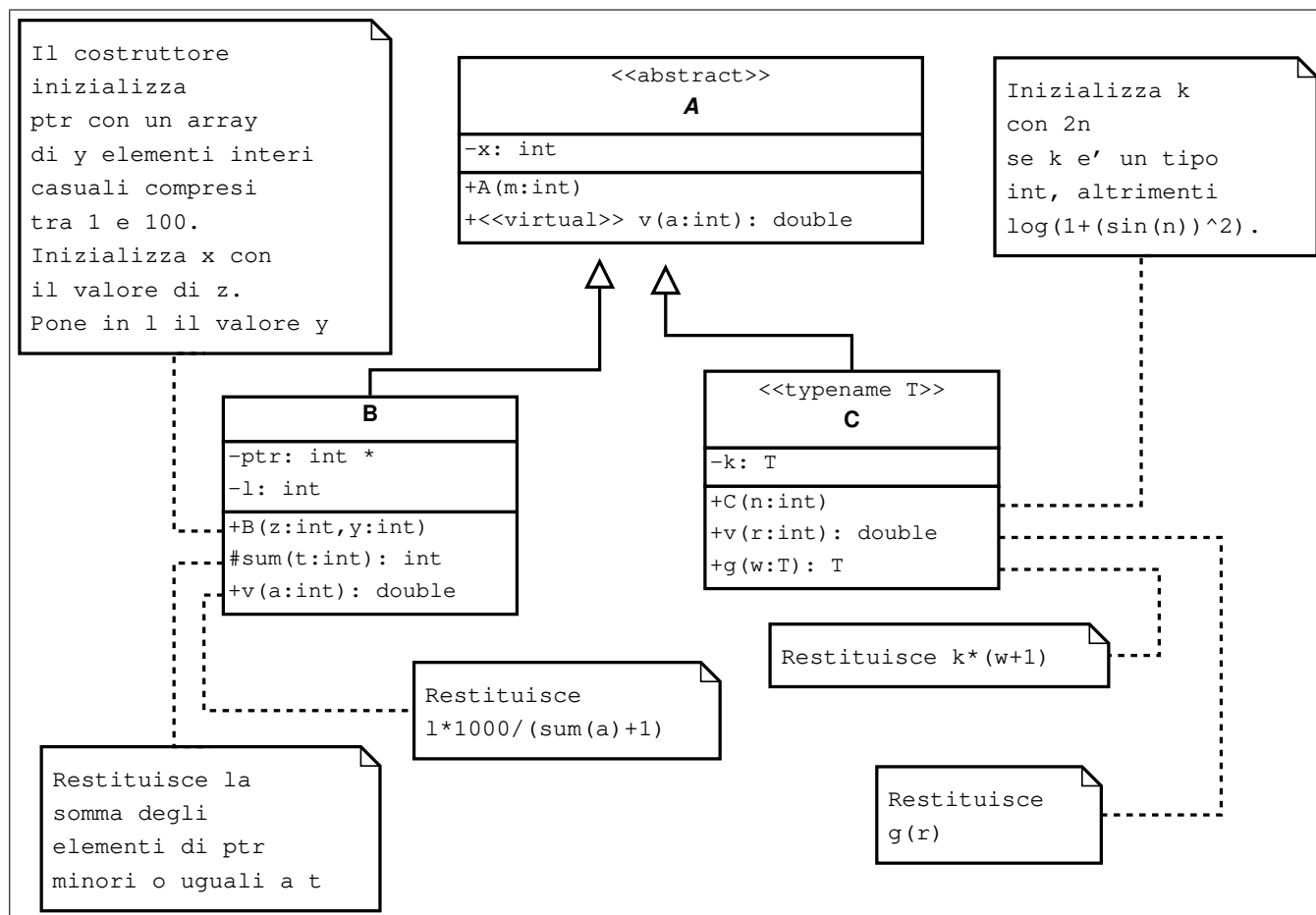


Si implementi in C++ la gerarchia ereditaria descritta dal seguente diagramma UML delle classi. NB: È necessario implementare tutti e soli i metodi indicati nel diagramma.



In un main, si generi una collezione di 50 oggetti utilizzando la sequenza di valori casuali riportata sul retro del foglio. **NB:** È possibile scaricare il frammento di codice da inserire nel main a partire dai link indicati sul retro del foglio

Successivamente:

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

```

35)1B, x=9, ptr=[ 82 24 4 93 84 85 ], v(50)=214.286
36)1CIiE, x=2, k=4, v(50)=204
37)1CIiE, x=10, k=20, v(50)=1020
    
```

2. si calcoli il minimo ed il massimo valore v(50) per tutti gli oggetti della collezione, e la media dei valori g(5) per per tutti gli oggetti di tipo C<double>;
3. si implementi l'overloading dello operatore membro "[]" per la classe B e lo si utilizzi nel main in una istruzione di assegnamento.

```
srand(111222333);

A *vett [DIM];

for(int i=0; i<DIM; i++) {
    int n=1+rand()%10;
    switch(rand()%3) {
        case 0:
            vett[i]= new B(n, rand()%10 + 1);
            break;
        case 1:
            vett[i]= new C<int>(n);
            break;
        case 2:
            vett[i]= new C<double>(n);
    }
}
```

1. Frame di codice da inserire nella funzione main:

www.dmi.unict.it/~messina/didat/prog1_18_19/15_02_2019/frameB.cpp

oppure Short URL:

<https://goo.gl/Gdn472>