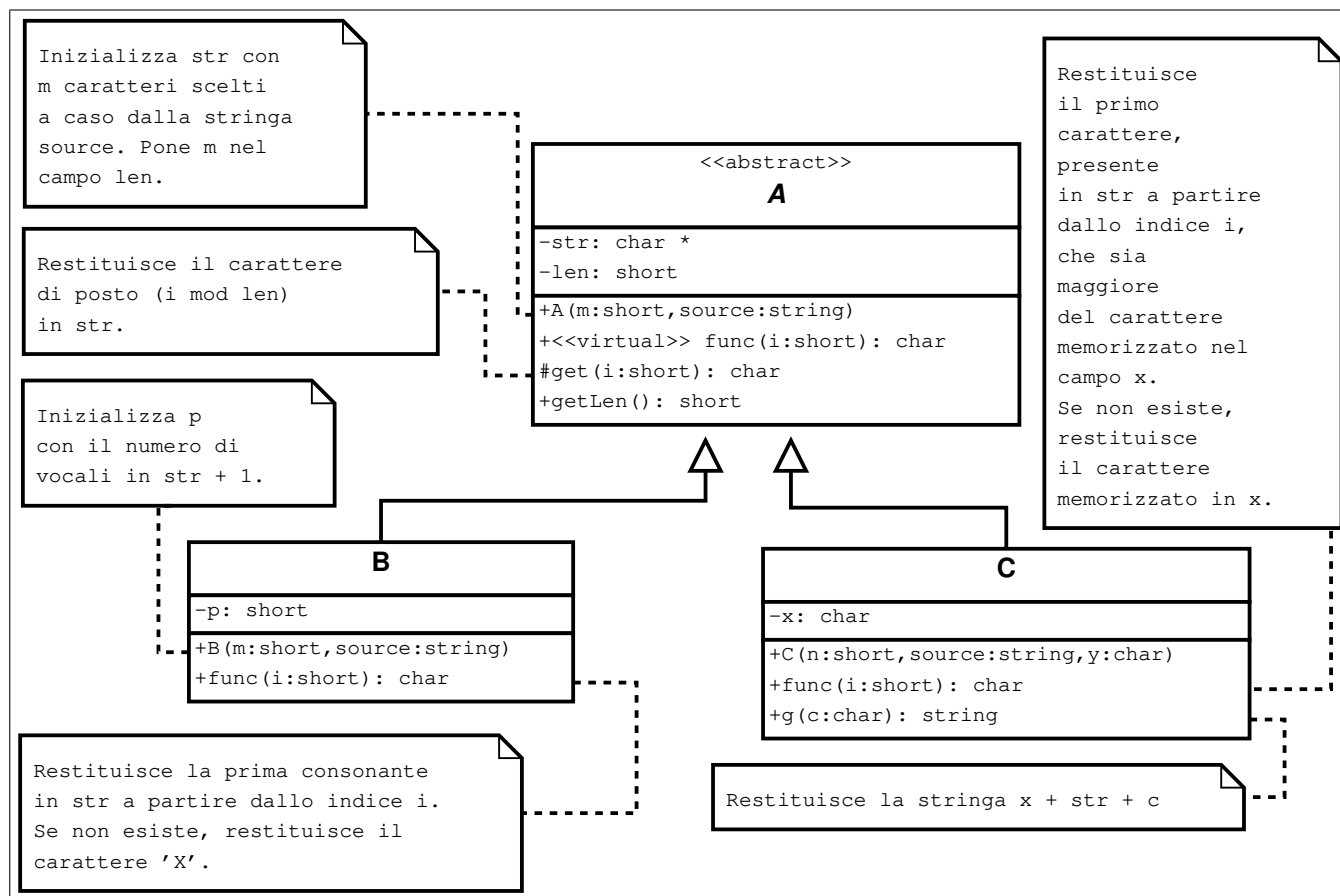


Si implementi in C++ la gerarchia ereditaria descritta dal seguente diagramma UML delle classi. NB: È necessario implementare tutti e soli i metodi indicati nel diagramma.



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:

```

36)1B, str = cosgeliis , p=5, func(3)=g
37)1C, str = oaecasa , x=p, func(3)=s
38)1B, str = rehisl , p=3, func(3)=s
    
```

2. 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 `"()` (function call) per la classe A e lo si utilizzi nel main. L'operatore deve prevedere due parametri formali interi `i1` e `i2` e restituire la sottostringa di str formata dai caratteri presenti in str dall'indice `i1` all'indice `i2`.

```
srand(111222333);
A *vett [DIM];

string S="supercalifragilistichepsiralidoso";

for(int i=0; i<DIM; i++){
    short n=3+rand()%10;
    if(rand()%2==0)
        vett[i]= new B(n, S);
    else
        vett[i]= new C(n, S, (char) (rand()%('z'-'a'+1)+'a'));
}
```

1. Codice da inserire nella funzione main:

www.dmi.unict.it/~messina/didat/prog1_18_19/30_04_2019/A/frame-30_04_A.cpp

oppure Short URL:

<https://tinyurl.com/y265kggg>