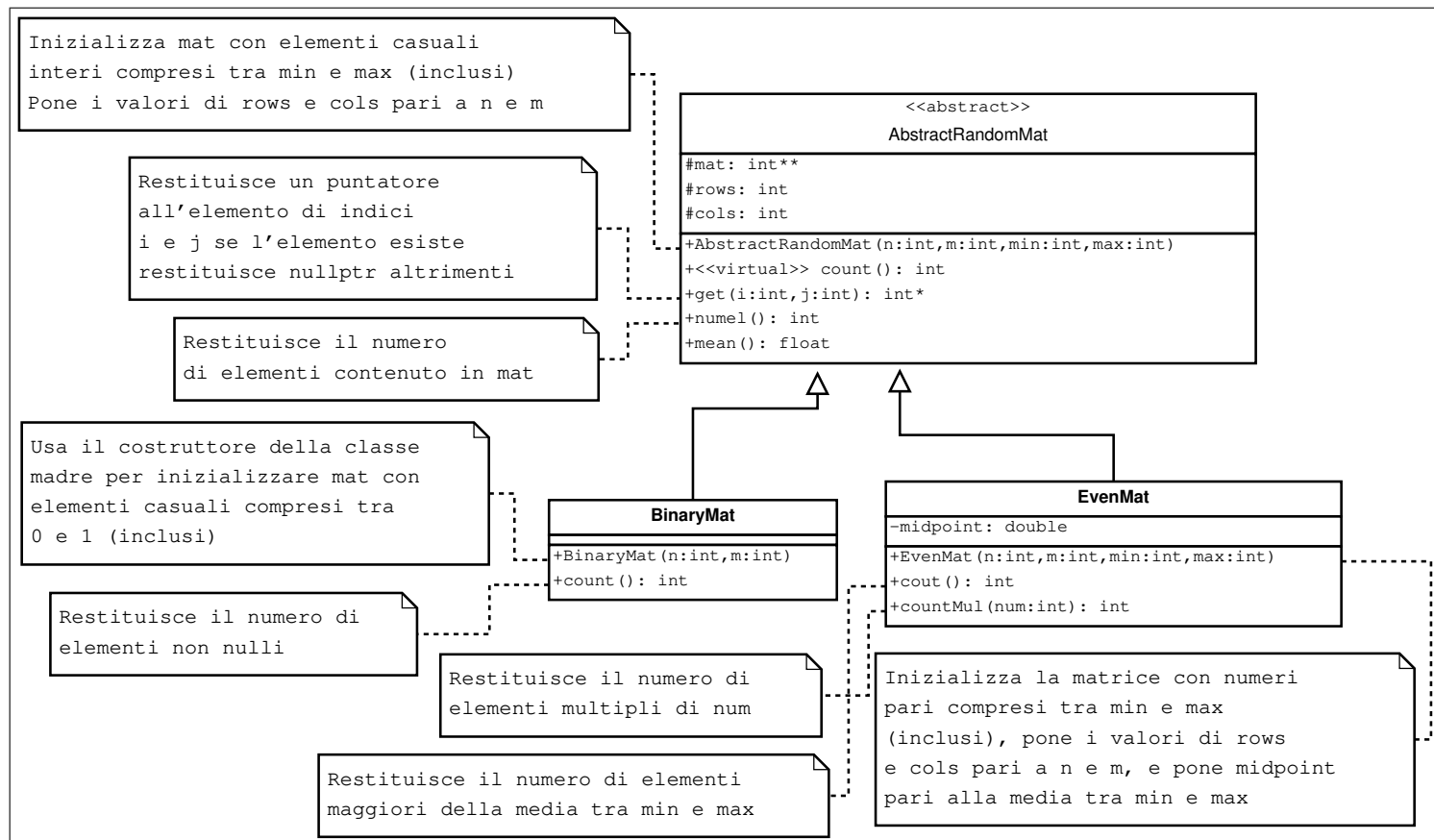


Università di Catania
Dipartimento di Matematica e Informatica
Corso di Studio in Informatica, A.A. 2021-2022
Prova di laboratorio di Programmazione I M-Z (9 CFU).
08 Febbraio 2022 - Turno 2

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



In un main, si generi una collezione di 25 oggetti utilizzando la sequenza di valori casuali riportata sul retro del foglio. **NB:** È possibile scaricare il frammento di codice da inserire nel main dal seguente URL: https://www.antoninofurnari.it/downloads/lab080222_2/frame-08_02_2022.cpp

Successivamente:

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

21) `Class=9BinaryMat, rows=5, cols=6, mat=[[0 0 1 1 0 1][0 0 0 0 0 1][1 1 0 0 0 1] [0 0 0 0 1 0][0 0 0 0 0 1]]`

22) `Class=7EvenMat, rows=7, cols=6, mat=[[14 14 12 14 12 12][16 16 16 14 12 14][16 12 16 14 12] [12 14 16 12 12 16][16 16 12 12 12 14][12 12 16 12 14 16][16 16 14 14 14 14]], midpoint=14`

2. si calcoli il massimo valore `count()` per tutti gli oggetti della collezione
3. si calcoli la media dei valori `countMul(4)` per per tutti gli oggetti di tipo `EvenMat`;

Output di controllo indicativo (Linux):

https://www.antoninofurnari.it/downloads/lab080222_2/out_lin.txt

La consegna deve avvenire attraverso il seguente form: <https://forms.gle/9gxaY8kiSTQe7e8w6>

```
srand(424242);

for(int i=0; i<DIM; i++) {
    int n=5+rand()%5;
    int m=5+rand()%5;
    int min = rand()%10 + 5;
    int max = min + rand()%20;

    switch(rand()%2) {
        case 0:
            vett[i]= new BinaryMat(n, m);
            break;
        case 1:
            vett[i]= new EvenMat(n, m, min, max);
    }
}
```