Volem implementar una classe Matriu que permeti guardar una matriu de nombres reals de qualsevol dimensió i fer algunes operacions bàsiques amb els valors de la matriu. Aquesta classe ha de contenir:

-          Tots els atributs necessaris per guardar una matriu dinàmica de nombres reals de qualsevol dimensió (la dimensió quedarà fixada quan s’inicialitzi la matriu).

-          Un constructor per defecte que inicialitza una matriu buida amb dimensions (0,0).

-          Un destructor.

-          Un constructor amb dos paràmetres que siguin el nº de files i el nº de columnes de la matriu, que inicialitzi correctament la matriu a les dimensions indicades, creant tota la memòria dinàmica que sigui necessària. Tots els valors de la matriu han de quedar inicialitzats a 0.

-          Un constructor de còpia que inicialitzi la matriu de forma que sigui exactament igual a la matriu que es passa com a paràmetre (tant en dimensions com en contingut).

-          Un operador d’assignació que permeti assignar a la matriu el valor d’una altra matriu (m1 = m2) de forma que passin a ser iguals (tant en dimensions com en contingut). El contingut previ de la matriu es perd (i per tant, també s’ha d’alliberar la memòria dinàmica que tingués reservada).

-          Un mètode resize que permeti modificar les dimensions de la matiu als valors que es passen com a paràmetre (creant la memòria dinàmica necessària). Igual que passa amb l’operador d’assignació, el contingut previ de la matriu (si n’hi havia) es perd (i per tant, també s’ha d’alliberar la memòria dinàmica que tingués reservada). Si les noves dimensions són més petites que les originals, els valors de la matriu han de quedar inicialitzats als mateixos valors que la matriu original per totes les posicions de la nova matriu que també existien a la matriu original. Si les dimensions són més grans (ja sigui per files o columnes), els nous valors s’han d’inicialitzar a 0. La capçalera d’aquest mètode ha de ser la següent:

void resize(int nFiles, int nColumnes);

-          Un mètode transpose que transposi la matriu (intercanviï files per columnes). S’ha d’alliberar la memòria dinàmica de la matriu original i assignar nova memòria dinàmica per les noves dimensions intercanviades de la matriu. La capçalera d’aquest mètode ha de ser la següent:

void transpose(int nFiles, int nColumnes);

-          Un mètode setValor per modificar el valor de la matriu en una posició determinada. Si la fila i columna que s’indiquen com a posició a modificar són més grans que el tamany de la matriu no s’ha de fer res.

-          Un mètode getValor per modificar el valor de la matriu en una posició determinada. Si la fila i columna que s’indiquen com a posició a recuperar són més grans que el tamany de la matriu s’ha de retornar el valor especial NAN.

-          Mètodes getNFiles i getNColumnes per recuperar el nº de files i de columnes de la matriu, respectivament.

-          Un mètode esBuida que retorni si la matriu està buida (encara no s’ha iniciallitzat a unes dimensions determinades).

-          La sobrecàrrega de l’operador + de suma perquè rebi com a paràmetre una matriu i retorni el resultat de sumar la matriu actual amb la matriu que es passa com a paràmetre. Si les dimensions de les dues matrius no són iguals s’ha de retornar una matriu buida.

-          Una segona sobrecàrrega de l’operador + de suma perquè rebi com a paràmetre un nº real i retorni el resultat de sumar aquest valor a tots els valors de la matriu.