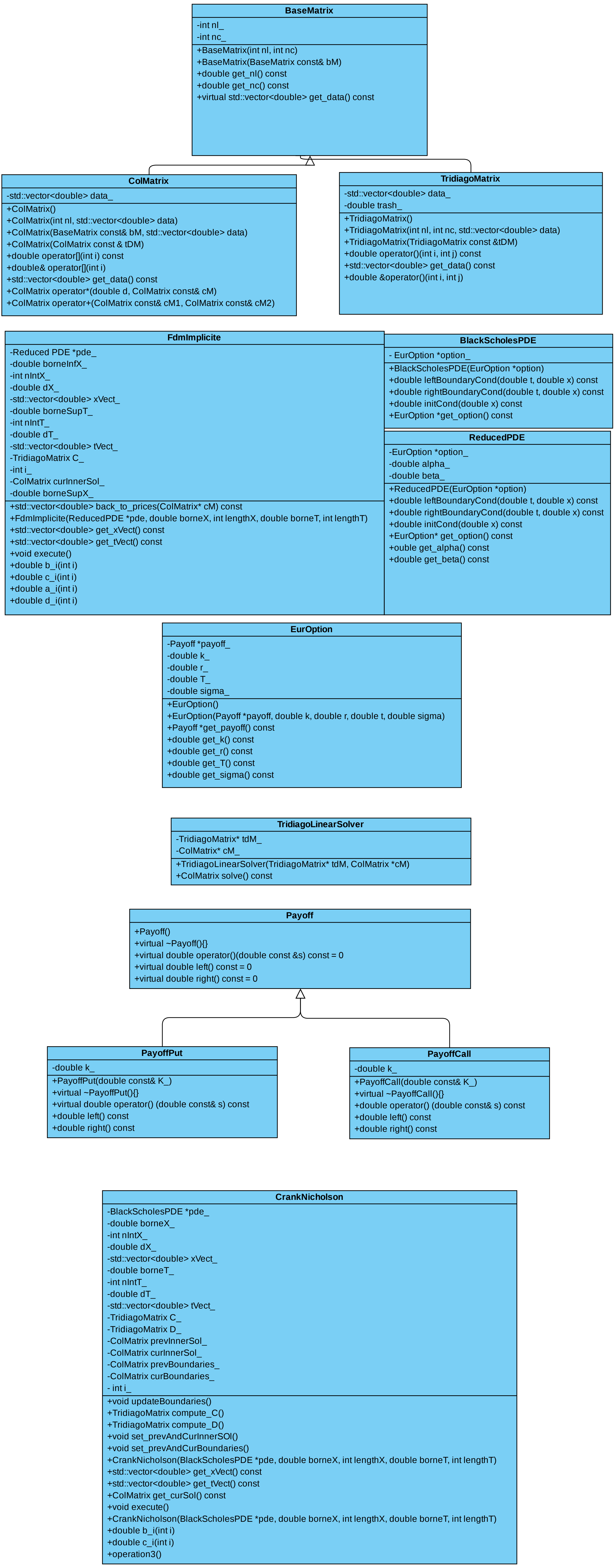

**BaseMatrix**

-int nl_
-int nc_

+BaseMatrix(int nl, int nc)
+BaseMatrix(BaseMatrix const& bM)
+double get_nl() const
+double get_nc() const
+virtual std::vector<double> get_data() const

**ColMatrix**

-std::vector<double> data_

+ColMatrix()
+ColMatrix(int nl, std::vector<double> data)
+ColMatrix(BaseMatrix const& bM, std::vector<double> data)
+ColMatrix(ColMatrix const & tDM)
+double operator[](int i) const
+double& operator[](int i)
+std::vector<double> get_data() const
+ColMatrix operator*(double d, ColMatrix const& cM)
+ColMatrix operator+(ColMatrix const& cM1, ColMatrix const& cM2)

**TridiagoMatrix**

-std::vector<double> data_
-double trash_

+TridiagoMatrix()
+TridiagoMatrix(int nl, int nc, std::vector<double> data)
+TridiagoMatrix(TridiagoMatrix const &tDM)
+double operator()(int i, int j) const
+std::vector<double> get_data() const
+double &operator()(int i, int j)

**FdmImplicite**

-Reduced PDE *pde_
-double borneInfX_
-int nIntX_
-double dX_
-std::vector<double> xVect_
-double borneSupT_
-int nIntT_
-double dT_
-std::vector<double> tVect_
-TridiagoMatrix C_
-int i_
-ColMatrix curInnerSol_
-double borneSupX_

+std::vector<double> back_to_prices(ColMatrix* cM) const
+FdmImplicite(ReducedPDE *pde, double borneX, int lengthX, double borneT, int lengthT)
+std::vector<double> get_xVect() const
+std::vector<double> get_tVect() const
+void execute()
+double b_i(int i)
+double c_i(int i)
+double a_i(int i)
+double d_i(int i)

**BlackScholesPDE**

- EurOption *option_

+BlackScholesPDE(EurOption *option)
+double leftBoundaryCond(double t, double x) const
+double rightBoundaryCond(double t, double x) const
+double initCond(double x) const
+EurOption *get_option() const

**ReducedPDE**

-EurOption *option_
-double alpha_
-double beta_

+ReducedPDE(EurOption *option)
+double leftBoundaryCond(double t, double x) const
+double rightBoundaryCond(double t, double x) const
+double initCond(double x) const
+EurOption* get_option() const
+ouble get_alpha() const
+double get_beta() const

**EurOption**

-Payoff *payoff_
-double k_
-double r_
-double T_
-double sigma_

+EurOption()
+EurOption(Payoff *payoff, double k, double r, double t, double sigma)
+Payoff *get_payoff() const
+double get_k() const
+double get_r() const
+double get_T() const
+double get_sigma() const

**TridiagoLinearSolver**

-TridiagoMatrix* tdM_
-ColMatrix* cM_

+TridiagoLinearSolver(TridiagoMatrix* tdM, ColMatrix *cM)
+ColMatrix solve() const

**Payoff**

+Payoff()
+virtual ~Payoff(){}
+virtual double operator()(double const &s) const = 0
+virtual double left() const = 0
+virtual double right() const = 0

**PayoffPut**

-double k_

+PayoffPut(double const& K_)
+virtual ~PayoffPut(){}
+virtual double operator() (double const& s) const
+double left() const
+double right() const

**PayoffCall**

-double k_

+PayoffCall(double const& K_)
+virtual ~PayoffCall(){}
+double operator() (double const& s) const
+double left() const
+double right() const

**CrankNicholson**

-BlackScholesPDE *pde_
-double borneX_
-int nIntX_
-double dX_
-std::vector<double> xVect_
-double borneT_
-int nIntT_
-double dT_
-std::vector<double> tVect_
-TridiagoMatrix C_
-TridiagoMatrix D_
-ColMatrix prevInnerSol_
-ColMatrix curInnerSol_
-ColMatrix prevBoundaries_
-ColMatrix curBoundaries_
- int i_

+void updateBoundaries()
+TridiagoMatrix compute_C()
+TridiagoMatrix compute_D()
+void set_prevAndCurInnerSOl()
+void set_prevAndCurBoundaries()
+CrankNicholson(BlackScholesPDE *pde, double borneX, int lengthX, double borneT, int lengthT)
+std::vector<double> get_xVect() const
+std::vector<double> get_tVect() const
+ColMatrix get_curSol() const
+void execute()
+CrankNicholson(BlackScholesPDE *pde, double borneX, int lengthX, double borneT, int lengthT)
+double b_i(int i)
+double c_i(int i)
+operation3()