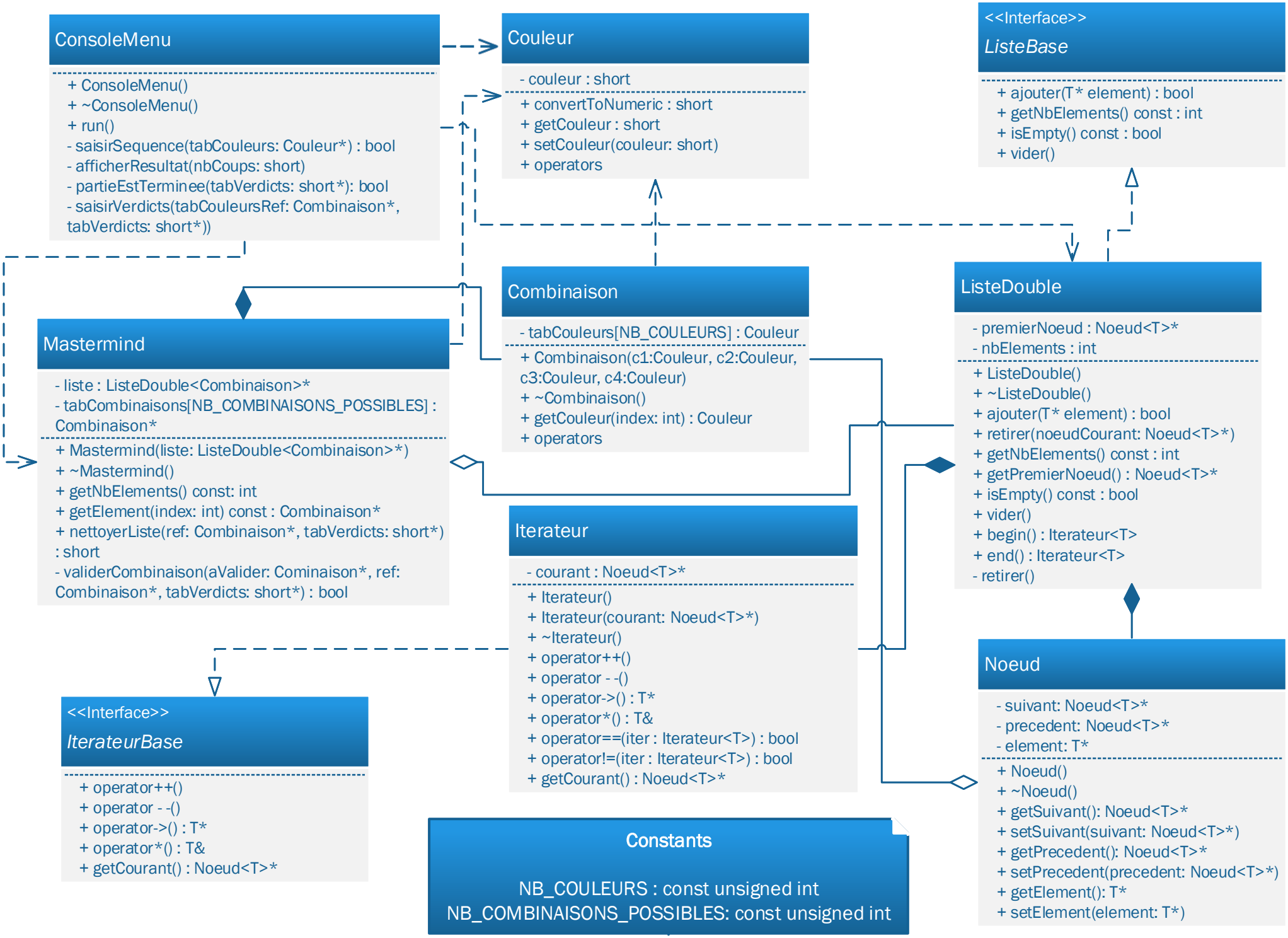
ConsoleMenu
+ ConsoleMenu()
+ ~ConsoleMenu()
+ run()
- saisirSequence(tabCouleurs: Couleur*) : bool
- afficherResultat(nbCoups: short)
- partieEstTerminee(tabVerdicts: short*): bool
- saisirVerdicts(tabCouleursRef: Combinaison*, tabVerdicts: short*))
Couleur
- couleur : short
+ convertToNumeric : short
+ getCouleur : short
+ setCouleur(couleur: short)
+ operators
<<Interface>>
ListeBase
+ ajouter(T* element) : bool
+ getNbElements() const : int
+ isEmpty() const : bool
+ vider()
Combinaison
- tabCouleurs[NB_COULEURS] : Couleur
+ Combinaison(c1:Couleur, c2:Couleur, c3:Couleur, c4:Couleur)
+ ~Combinaison()
+ getCouleur(index: int) : Couleur
+ operators
ListeDouble
- premierNoeud : Noeud<T>*
- nbElements : int
+ ListeDouble()
+ ~ListeDouble()
+ ajouter(T* element) : bool
+ retirer(noeudCourant: Noeud<T>*)
+ getNbElements() const : int
+ getPremierNoeud() : Noeud<T>*
+ isEmpty() const : bool
+ vider()
+ begin() : Iterateur<T>
+ end() : Iterateur<T>
- retirer()
Mastermind
- liste : ListeDouble<Combinaison>*
- tabCombinaisons[NB_COMBINAISONS_POSSIBLES] : Combinaison*
+ Mastermind(liste: ListeDouble<Combinaison>*)
+ ~Mastermind()
+ getNbElements() const: int
+ getElement(index: int) const : Combinaison*
+ nettoyerListe(ref: Combinaison*, tabVerdicts: short*) : short
- validerCombinaison(aValider: Cominaison*, ref: Combinaison*, tabVerdicts: short*) : bool
Iterateur
- courant : Noeud<T>*
+ Iterateur()
+ Iterateur(courant: Noeud<T>*)
+ ~Iterateur()
+ operator++()
+ operator - -()
+ operator->() : T*
+ operator*() : T&
+ operator==(iter : Iterateur<T>) : bool
+ operator!=(iter : Iterateur<T>) : bool
+ getCourant() : Noeud<T>*
<<Interface>>
IterateurBase
+ operator++()
+ operator - -()
+ operator->() : T*
+ operator*() : T&
+ getCourant() : Noeud<T>*
Noeud
- suivant: Noeud<T>*
- precedent: Noeud<T>*
- element: T*
+ Noeud()
+ ~Noeud()
+ getSuivant(): Noeud<T>*
+ setSuivant(suivant: Noeud<T>*)
+ getPrecedent(): Noeud<T>*
+ setPrecedent(precedent: Noeud<T>*)
+ getElement(): T*
+ setElement(element: T*)
Constants
NB_COULEURS : const unsigned int
NB_COMBINAISONS_POSSIBLES: const unsigned int