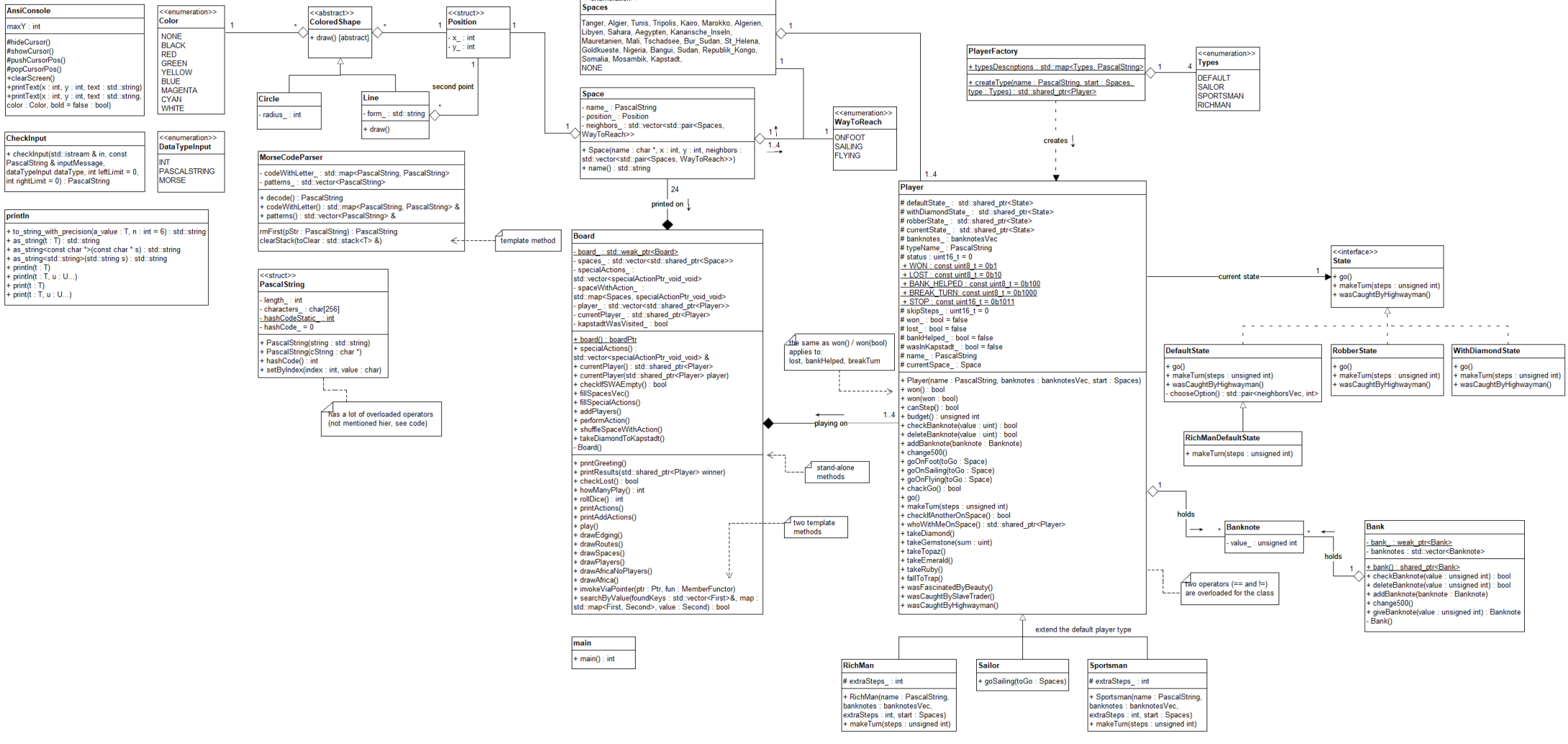
AnsiConsole
maxY : int
#hideCursor()
#showCursor()
#pushCursorPos()
#popCursorPos()
+clearScreen()
+printText(x : int, y : int, text : std::string)
+printText(x : int, y : int, text : std::string, color : Color, bold = false : bool)
<<enumeration>>
Color
NONE
BLACK
RED
GREEN
YELLOW
BLUE
MAGENTA
CYAN
WHITE
<<abstract>>
ColoredShape
+ draw() {abstract}
<<struct>>
Position
- x_ : int
- y_ : int
Circle
- radius_ : int
Line
- form_ : std::string
+ draw()
second point
CheckInput
+ checkInput(std::istream & in, const PascalString & inputMessage, dataTypeInput dataType, int leftLimit = 0, int rightLimit = 0) : PascalString
<<enumeration>>
DataTypeInput
INT
PASCALSTRING
MORSE
MorseCodeParser
- codeWithLetter_ : std::map<PascalString, PascalString>
- patterns_ : std::vector<PascalString>
+ decode() : PascalString
+ codeWithLetter() : std::map<PascalString, PascalString> &
+ patterns() : std::vector<PascalString> &
rmFirst(pStr : PascalString) : PascalString
clearStack(toClear : std::stack<T> &)
template method
println
+ to_string_with_precision(a_value : T, n : int = 6) : std::string
+ as_string(t : T) : std::string
+ as_string<const char *>(const char * s) : std::string
+ as_string<std::string>(std::string s) : std::string
+ println(t : T)
+ println(t : T, u : U...)
+ print(t : T)
+ print(t : T, u : U...)
<<struct>>
PascalString
- length_ : int
- characters_ : char[256]
- hashCodeStatic_ : int
- hashCode_ = 0
+ PascalString(string : std::string)
+ PascalString(cString : char *)
+ hashCode() : int
+ setByIndex(index : int, value : char)
has a lot of overloaded operators (not mentioned hier, see code)
<<enumeration>>
Spaces
Tanger, Algier, Tunis, Tripolis, Kairo, Marokko, Algerien, Libyen, Sahara, Aegypten, Kanarische_Inseln, Mauretanien, Mali, Tschadsee, Bur_Sudan, St_Helena, Goldkueste, Nigeria, Bangui, Sudan, Republik_Kongo, Somalia, Mosambik, Kapstadt, NONE
Space
- name_ : PascalString
- position_ : Position
- neighbors_ : std::vector<std::pair<Spaces, WayToReach>>
+ Space(name : char *, x : int, y : int, neighbors : std::vector<std::pair<Spaces, WayToReach>>)
+ name() : std::string
<<enumeration>>
WayToReach
ONFOOT
SAILING
FLYING
printed on
Board
- board_ : std::weak_ptr<Board>
- spaces_ : std::vector<std::shared_ptr<Space>>
- specialActions_ : std::vector<specialActionPtr_void_void>
- spaceWithAction_ : std::map<Spaces, specialActionPtr_void_void>
- player_ : std::vector<std::shared_ptr<Player>>
- currentPlayer_ : std::shared_ptr<Player>
- kapstadtWasVisited_ : bool
+ board() : boardPtr
+ specialActions() : std::vector<specialActionPtr_void_void> &
+ currentPlayer() : std::shared_ptr<Player>
+ currentPlayer(std::shared_ptr<Player> player)
+ checkIfSWAEmpty() : bool
+ fillSpacesVec()
+ fillSpecialActions()
+ addPlayers()
+ performAction()
+ shuffleSpaceWithAction()
+ takeDiamondToKapstadt()
- Board()
+ printGreeting()
+ printResults(std::shared_ptr<Player> winner)
+ checkLost() : bool
+ howManyPlay() : int
+ rollDice() : int
+ printActions()
+ printAddActions()
+ play()
+ drawEdging()
+ drawRoutes()
+ drawSpaces()
+ drawPlayers()
+ drawAfricaNoPlayers()
+ drawAfrica()
+ invokeViaPointer(ptr : Ptr, fun : MemberFunctor)
+ searchByValue(foundKeys : std::vector<First>&, map : std::map<First, Second>, value : Second) : bool
the same as won() / won(bool) applies to: lost, bankHelped, breakTurn
stand-alone methods
two template methods
playing on
main
+ main() : int
PlayerFactory
+ typesDescriptions : std::map<Types, PascalString>
+ createType(name : PascalString, start : Spaces, type : Types) : std::shared_ptr<Player>
<<enumeration>>
Types
DEFAULT
SAILOR
SPORTSMAN
RICHMAN
creates
Player
# defaultState_ : std::shared_ptr<State>
# withDiamondState_ : std::shared_ptr<State>
# robberState_ : std::shared_ptr<State>
# currentState_ : std::shared_ptr<State>
# banknotes_ : banknotesVec
# typeName_ : PascalString
# status : uint16_t = 0
+ WON : const uint8_t = 0b1
+ LOST : const uint8_t = 0b10
+ BANK_HELPED : const uint8_t = 0b100
+ BREAK_TURN: const uint8_t = 0b1000
+ STOP : const uint16_t = 0b1011
# skipSteps_ : uint16_t = 0
# won_ : bool = false
# lost_ : bool = false
# bankHelped_ : bool = false
# wasInKapstadt_ : bool = false
# name_ : PascalString
# currentSpace_ : Space
+ Player(name : PascalString, banknotes : banknotesVec, start : Spaces)
+ won() : bool
+ won(won : bool)
+ canStep() : bool
+ budget() : unsigned int
+ checkBanknote(value : uint) : bool
+ deleteBanknote(value : uint) : bool
+ addBanknote(banknote : Banknote)
+ change500()
+ goOnFoot(toGo : Space)
+ goOnSailing(toGo : Space)
+ goOnFlying(toGo : Space)
+ chackGo() : bool
+ go()
+ makeTurn(steps : unsigned int)
+ checkIfAnotherOnSpace() : bool
+ whoWithMeOnSpace() : std::shared_ptr<Player>
+ takeDiamond()
+ takeGemstone(sum : uint)
+ takeTopaz()
+ takeEmerald()
+ takeRuby()
+ fallToTrap()
+ wasFascinatedByBeauty()
+ wasCaughtBySlaveTrader()
+ wasCaughtByHighwayman()
current state
<<interface>>
State
+ go()
+ makeTurn(steps : unsigned int)
+ wasCaughtByHighwayman()
DefaultState
+ go()
+ makeTurn(steps : unsigned int)
+ wasCaughtByHighwayman()
- chooseOption() : std::pair<neighborsVec, int>
RobberState
+ go()
+ makeTurn(steps : unsigned int)
+ wasCaughtByHighwayman()
WithDiamondState
+ go()
+ makeTurn(steps : unsigned int)
+ wasCaughtByHighwayman()
RichManDefaultState
+ makeTurn(steps : unsigned int)
holds
Banknote
- value_ : unsigned int
Bank
- bank_ : weak_ptr<Bank>
- banknotes : std::vector<Banknote>
+ bank() : shared_ptr<Bank>
+ checkBanknote(value : unsigned int) : bool
+ deleteBanknote(value : unsigned int) : bool
+ addBanknote(banknote : Banknote)
+ change500()
+ giveBanknote(value : unsigned int) : Banknote
- Bank()
two operators (== and !=) are overloaded for the class
extend the default player type
RichMan
# extraSteps_ : int
+ RichMan(name : PascalString, banknotes : banknotesVec, extraSteps : int, start : Spaces)
+ makeTurn(steps : unsigned int)
Sailor
+ goSailing(toGo : Spaces)
Sportsman
# extraSteps_ : int
+ Sportsman(name : PascalString, banknotes : banknotesVec, extraSteps : int, start : Spaces)
+ makeTurn(steps : unsigned int)