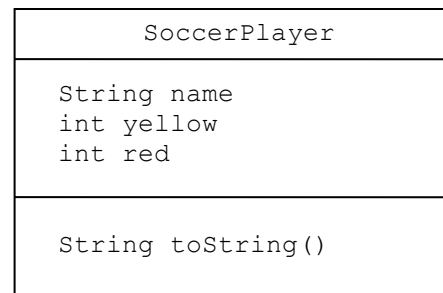


SoccerPlayer

1. Opret en klasse, *SoccerPlayer*, der repræsenterer en fodboldspiller. Klassen er specificeret i UML-diagrammet til højre. De tre feltvariabler skal initialiseres i en konstruktør (via parametre af passende type). Metoden *toString* skal returnere en streng-repræsentation for en *SoccerPlayer* på formen

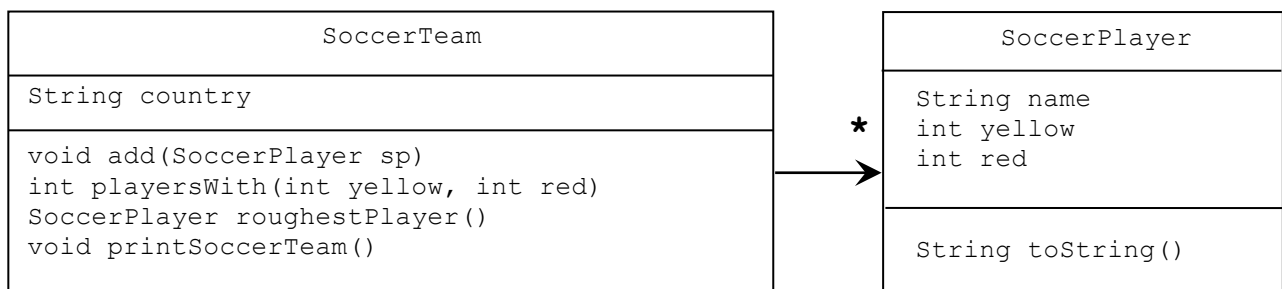
"Bendtner has 5 yellow cards and 1 red"



2. Lav en *Driver*-klasse med en *exam*-metode. Metoden skal være static, have returtype void og være uden parametre.
3. Opret fem velvalgte *SoccerPlayer*-objekter i *exam*-metoden, via objektreferencer *sp1*, *sp2*, *sp3*, *sp4* og *sp5*, og udskriv disse vha. *toString*-metoden.

Tilkald tilsynsførende og demonstrer det du har lavet indtil nu.

4. Opret en ny klasse, *SoccerTeam*, der repræsenterer et hold med fodboldspillere. Klassen *SoccerTeam*, og dens relation til klassen *SoccerPlayer*, er specificeret i følgende UML-diagram:



5. Programmér metoden *add*, der tilføjer *SoccerPlayer*-objektet *sp* til *SoccerTeam*-objektet.
6. Opret et objekt af typen *SoccerTeam* i *exam*-metoden i *Driver*-klassen og knyt de allerede oprettede *SoccerPlayer*-objekter hertil.
7. Programmér metoden *playersWith*. Metoden skal returnere antallet af spillere, der mindst har de angivne antal gule og røde kort. Udvid *SoccerPlayer*-klassen med de nødvendige get-metoder.
8. Afprøv metoden *playersWith* i *exam*-metoden i *Driver*-klassen.

Tilkald tilsynsførende og demonstrer det du har lavet indtil nu.

9. Programmér metoden *roughestPlayer*. Metoden skal returnere den spiller der har fået flest kort, idet røde kort vægtes tre gange så kraftigt som gule kort. Hvis der ikke findes en sådan spiller returneres null. Afprøv *roughestPlayer* i *exam*-metoden.
10. Programmér metoden *printSoccerTeam*. Metoden skal udskrive navnet på det land holdet kommer fra efterfulgt af alle spillere sorteret efter antal røde kort (lavest til højest). Hvis to har lige mange røde kort sorteres efter gule kort (lavest til højest). Afprøv *printSoccerTeam* i *exam*-metoden.

Tilkald tilsynsførende og demonstrer din færdige løsning.