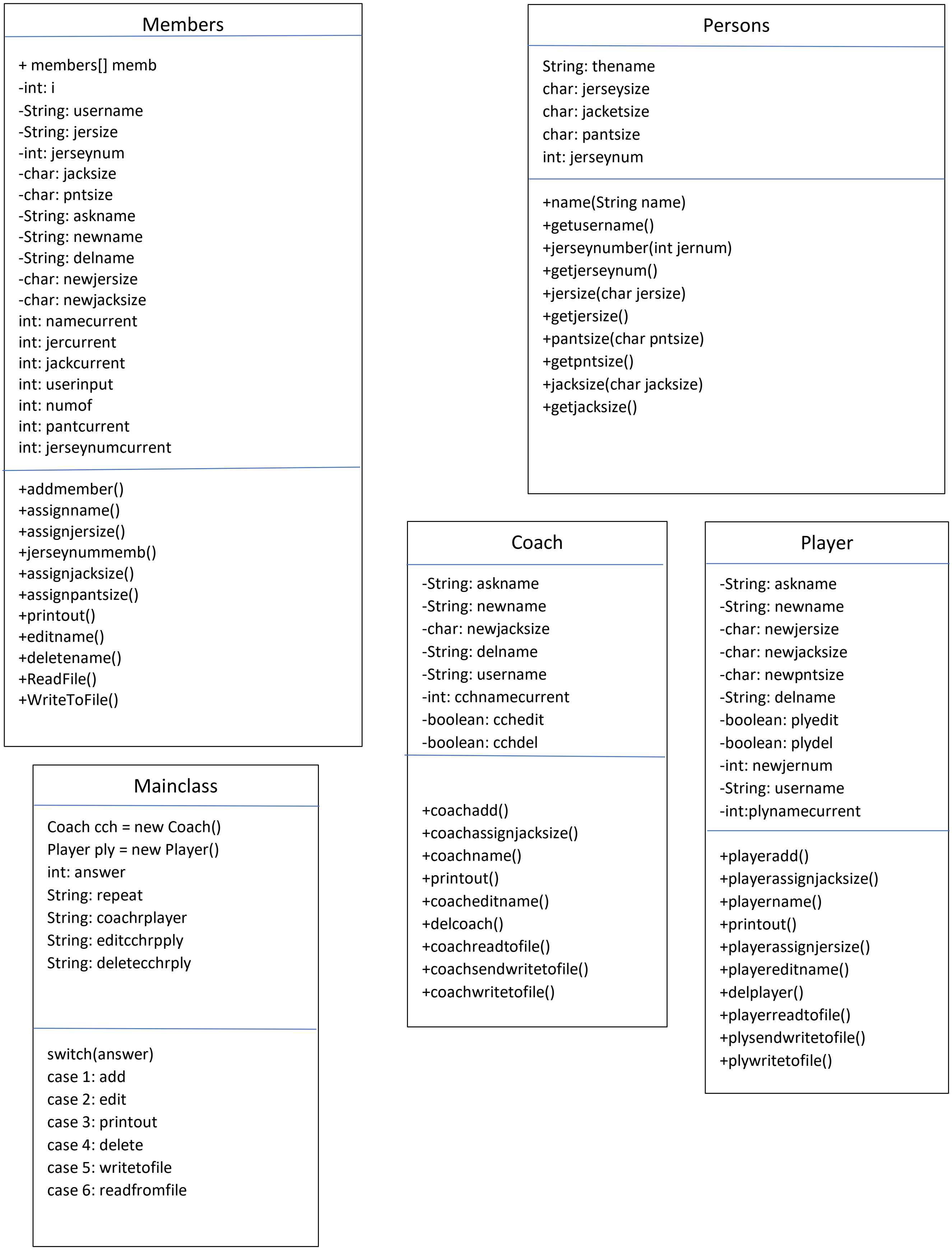
Criterion B: Design

Local Soccer Club Program (UML Diagram)



Members

+ members[] memb
-int: i
-String: username
-String: jersize
-int: jerseynum
-char: jacksize
-char: pntsize
-String: askname
-String: newname
-String: delname
-char: newjersize
-char: newjacksize
int: namecurrent
int: jercurrent
int: jackcurrent
int: userinput
int: numof
int: pantcurrent
int: jerseynumcurrent

+addmember()
+assignname()
+assignjersize()
+jerseynummemb()
+assignjacksize()
+assignpantsize()
+printout()
+editname()
+deletename()
+ReadFile()
+WriteToFile()

Persons

String: thename
char: jerseysize
char: jacketsize
char: pantsize
int: jerseynum

+name(String name)
+getusername()
+jerseynumber(int jernum)
+getjerseynum()
+jersize(char jersize)
+getjersize()
+pantsize(char pntsize)
+getpntsize()
+jacksize(char jacksize)
+getjacksize()

Coach

-String: askname
-String: newname
-char: newjacksize
-String: delname
-String: username
-int: cchnamecurrent
-boolean: cchedit
-boolean: cchdel

+coachadd()
+coachassignjacksize()
+coachname()
+printout()
+coacheditname()
+delcoach()
+coachreadtofile()
+coachsendwritetofile()
+coachwritetofile()

Player

-String: askname
-String: newname
-char: newjersize
-char: newjacksize
-char: newpntsize
-String: delname
-boolean: plyedit
-boolean: plydel
-int: newjernum
-String: username
-int:plynamecurrent

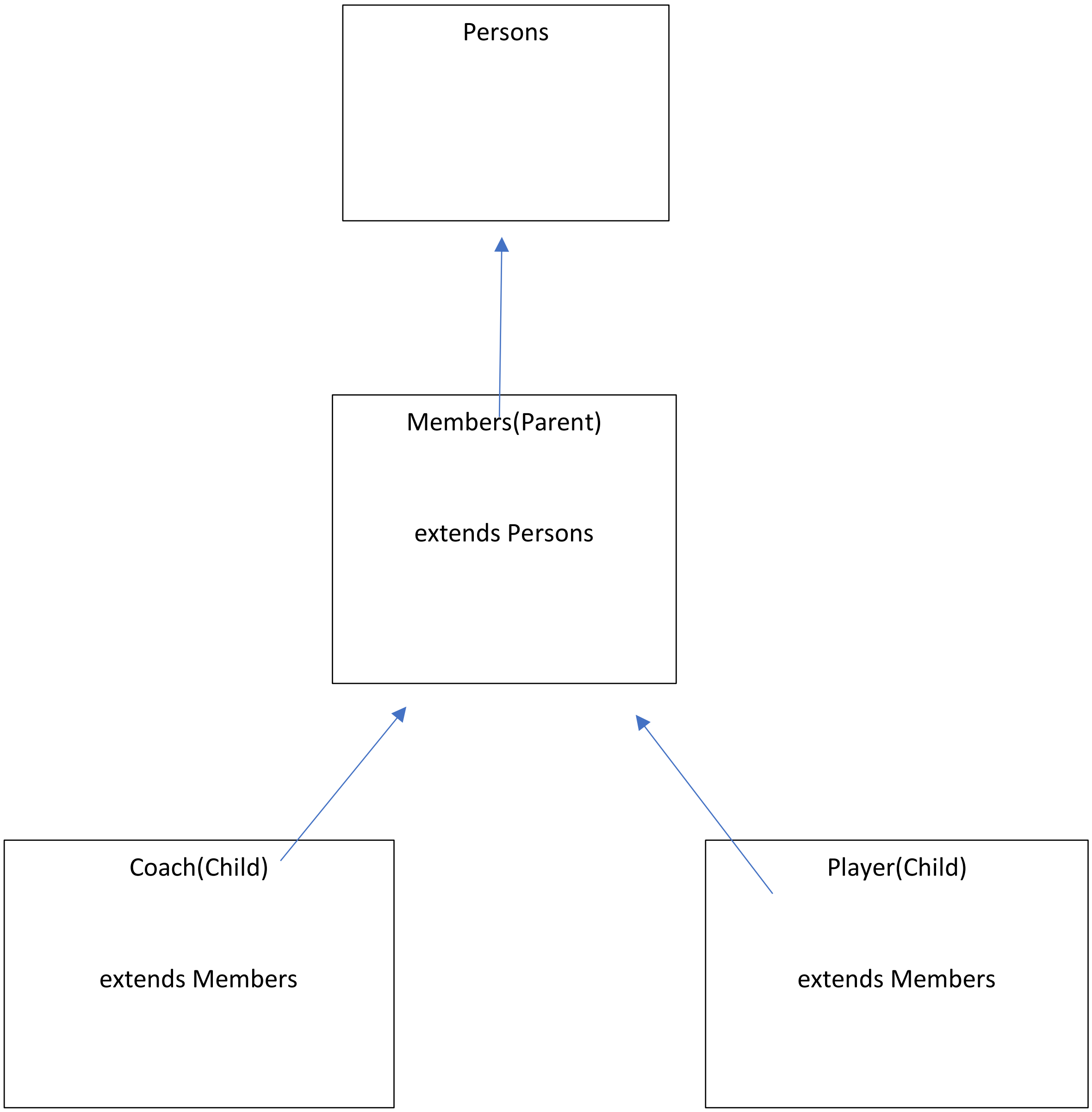
+playeradd()
+playerassignjacksize()
+playername()
+printout()
+playerassignjersize()
+playereditname()
+delplayer()
+playerreadtofile()
+plysendwritetofile()
+plywritetofile()

Mainclass

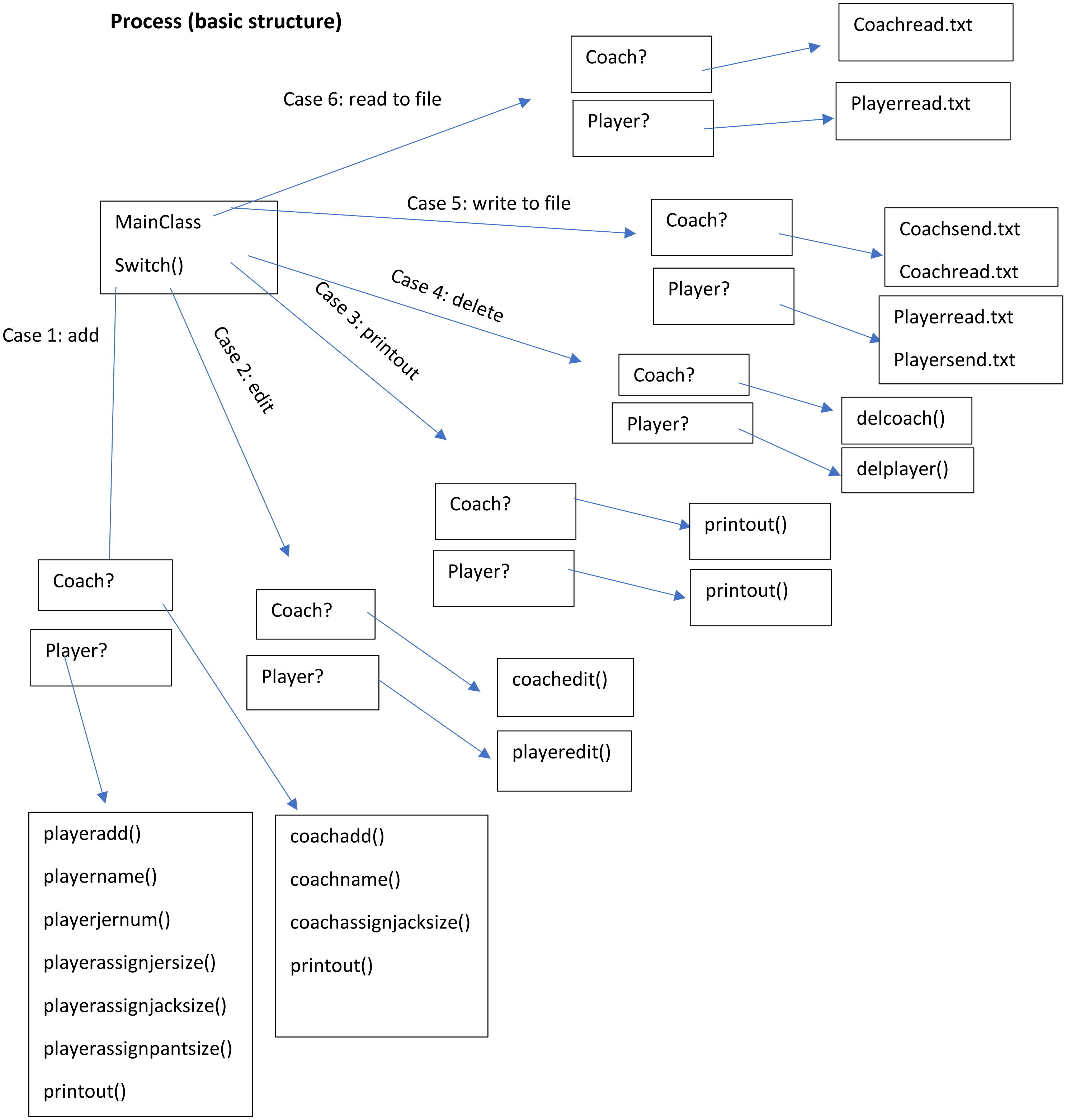
Coach cch = new Coach()
Player ply = new Player()
int: answer
String: repeat
String: coachrplayer
String: editcchrpply
String: deletecchrply

switch(answer)
case 1: add
case 2: edit
case 3: printout
case 4: delete
case 5: writetofile
case 6: readfromfile

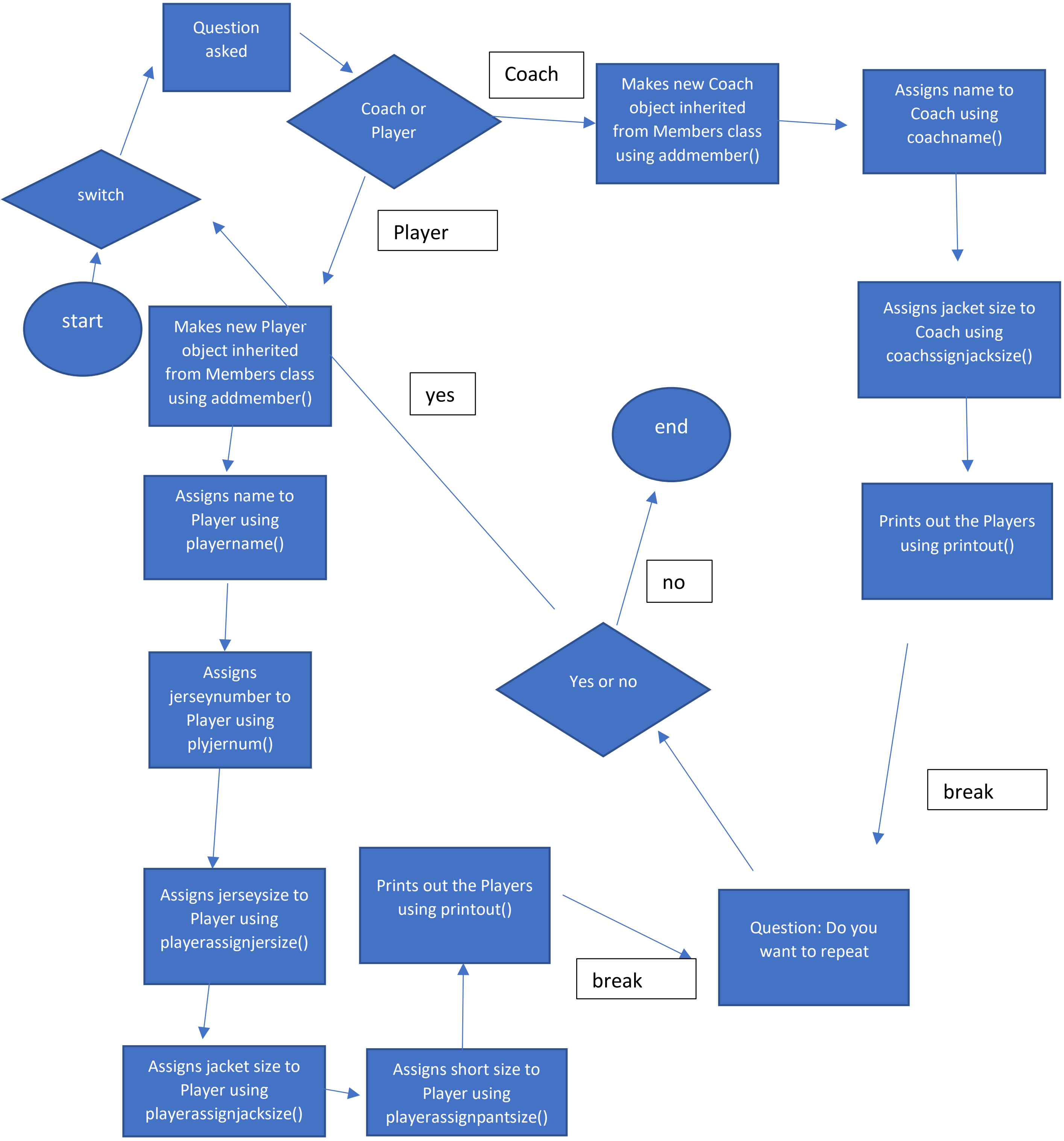
Overview of design:



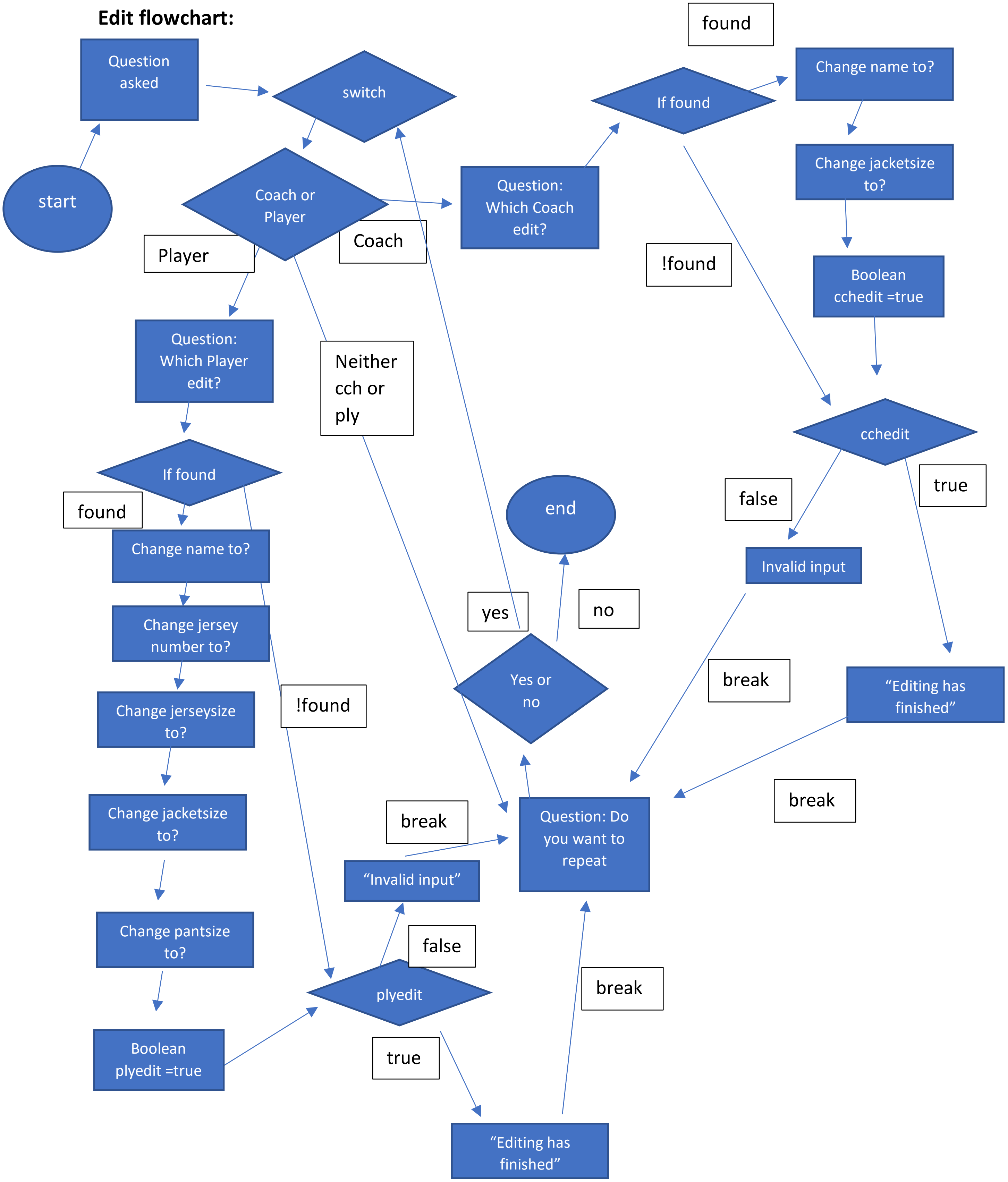
Process (basic structure)



Adding flowchart:


start
switch
Question asked
Coach or Player
Coach
Makes new Coach object inherited from Members class using addmember()
Assigns name to Coach using coachname()
Assigns jacket size to Coach using coachssignjacksize()
Prints out the Players using printout()
break
Player
Makes new Player object inherited from Members class using addmember()
Assigns name to Player using playername()
Assigns jerseynumber to Player using plyjernum()
Assigns jerseysize to Player using playerassignjersize()
Assigns jacket size to Player using playerassignjacksize()
Assigns short size to Player using playerassignpantsize()
Prints out the Players using printout()
break
Question: Do you want to repeat
Yes or no
yes
no
end

Edit flowchart:

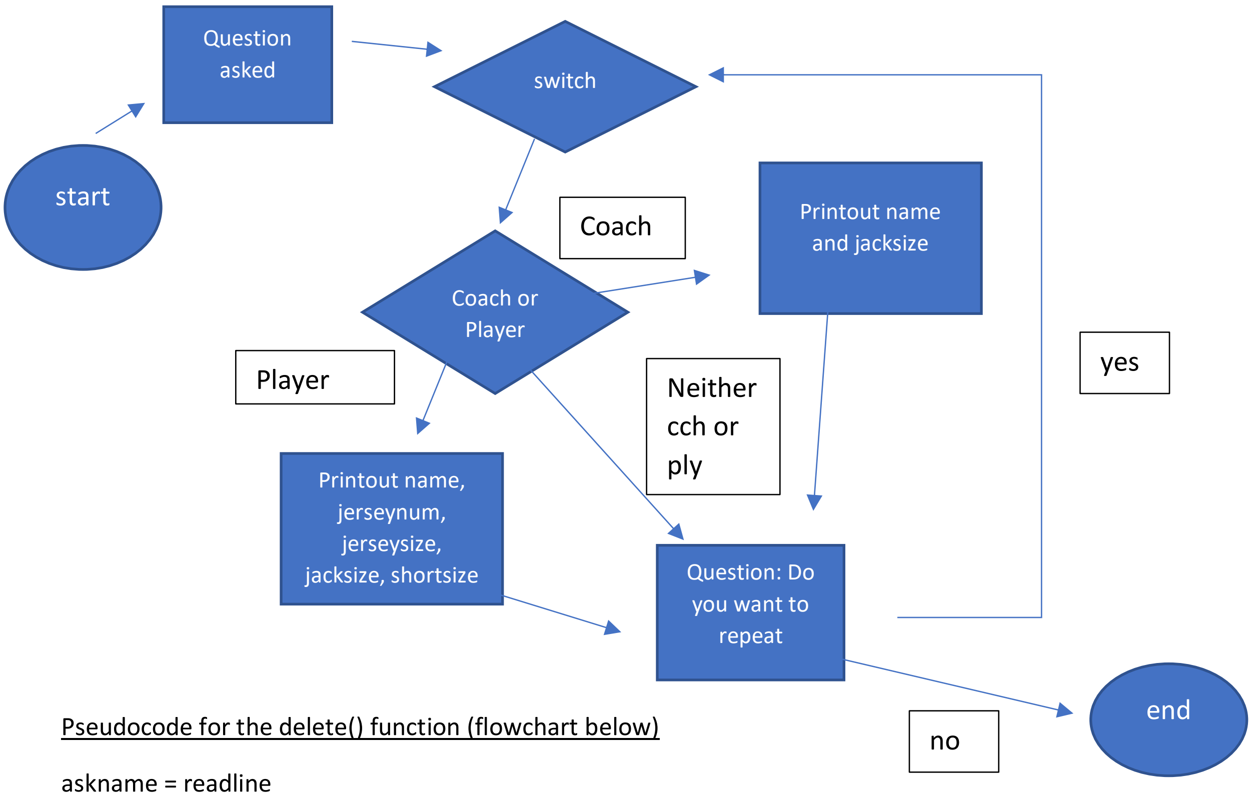


Pseudocode for edit() function (flowchart above)

```
askname = readline
For i=0 to i<userinput(global variable)
   If askname = name in memb obj array
      Then newname, jersize = readline and store in memb[i] (replaces old value with new value)
      Bool Edit = true

  If Edit=true
      Then output “editing finished”
  Else
      Output “you did not enter name correctly”
```

Printout flowchart:

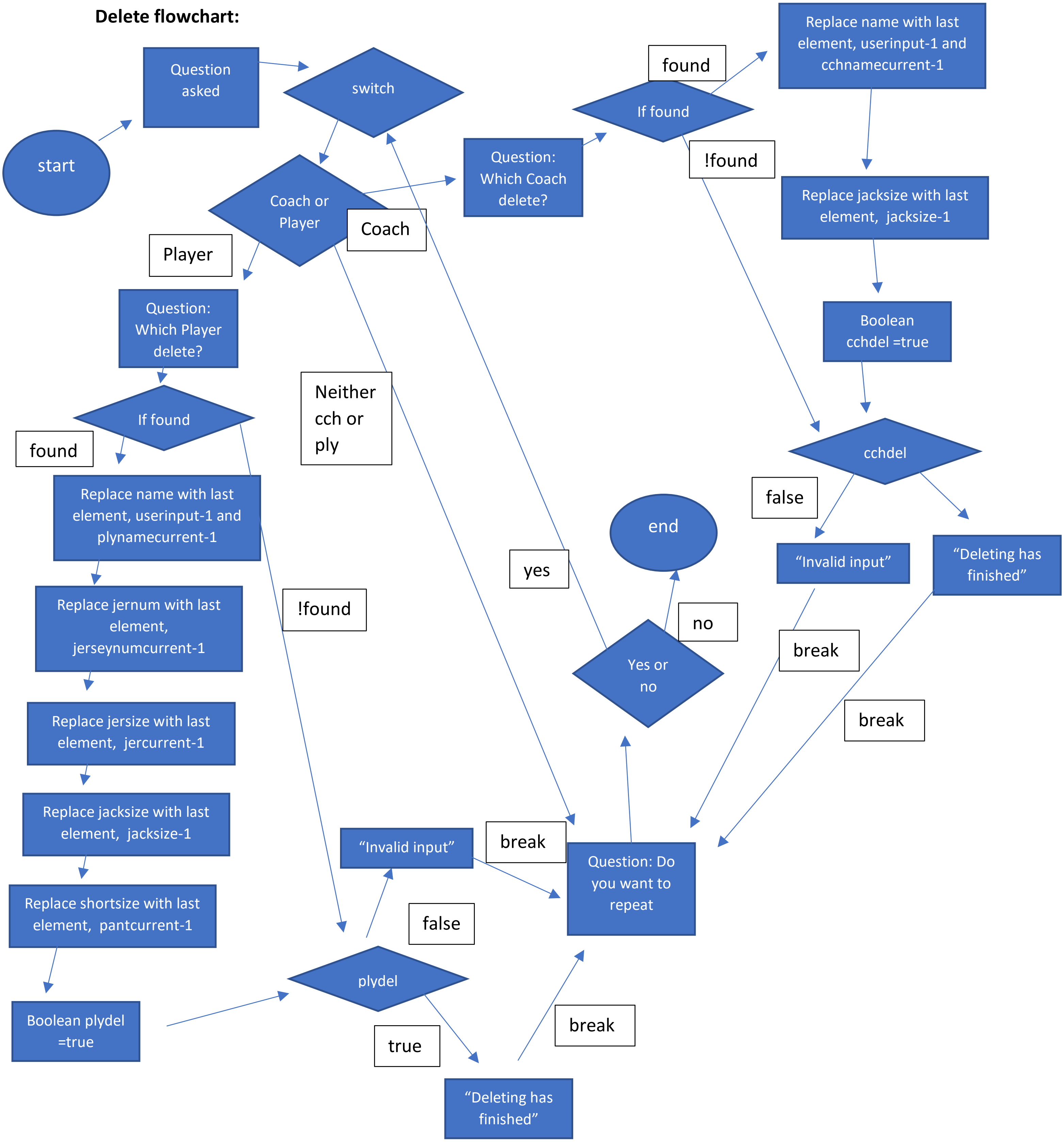

start
Question asked
switch
Coach or Player
Coach
Printout name and jacksize
Player
Printout name, jerseynum, jerseysize, jacksize, shortsize
Neither cch or ply
Question: Do you want to repeat
yes
no
end

Pseudocode for the delete() function (flowchart below)

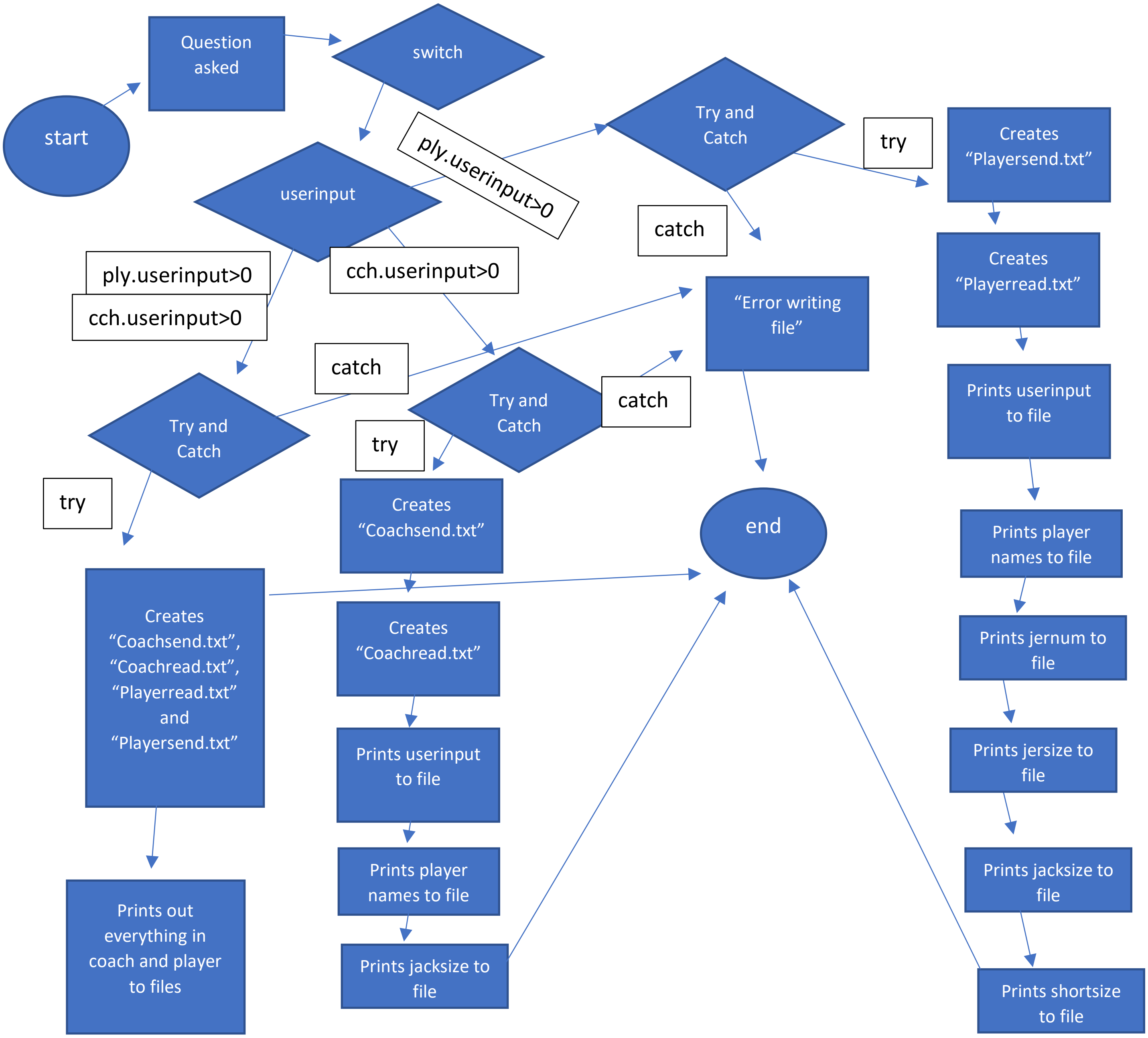
```
askname = readline
For i=0 to i<userinput(global variable)
   If askname = name in memb obj array
      Then prevname = memb[userinput-1] (last position)
      memb[i] = prevname (copy the value at last position to current position in loop)
      userinput -1 (since last element is duplicated it must be deleted)
      namecurrent-1(decrement counter for name)
      (same for all attributes of object)
      Bool Delete = true

  If Edit=true
      Then output “delete finished”
  Else
      Output “you did not enter name correctly”
```

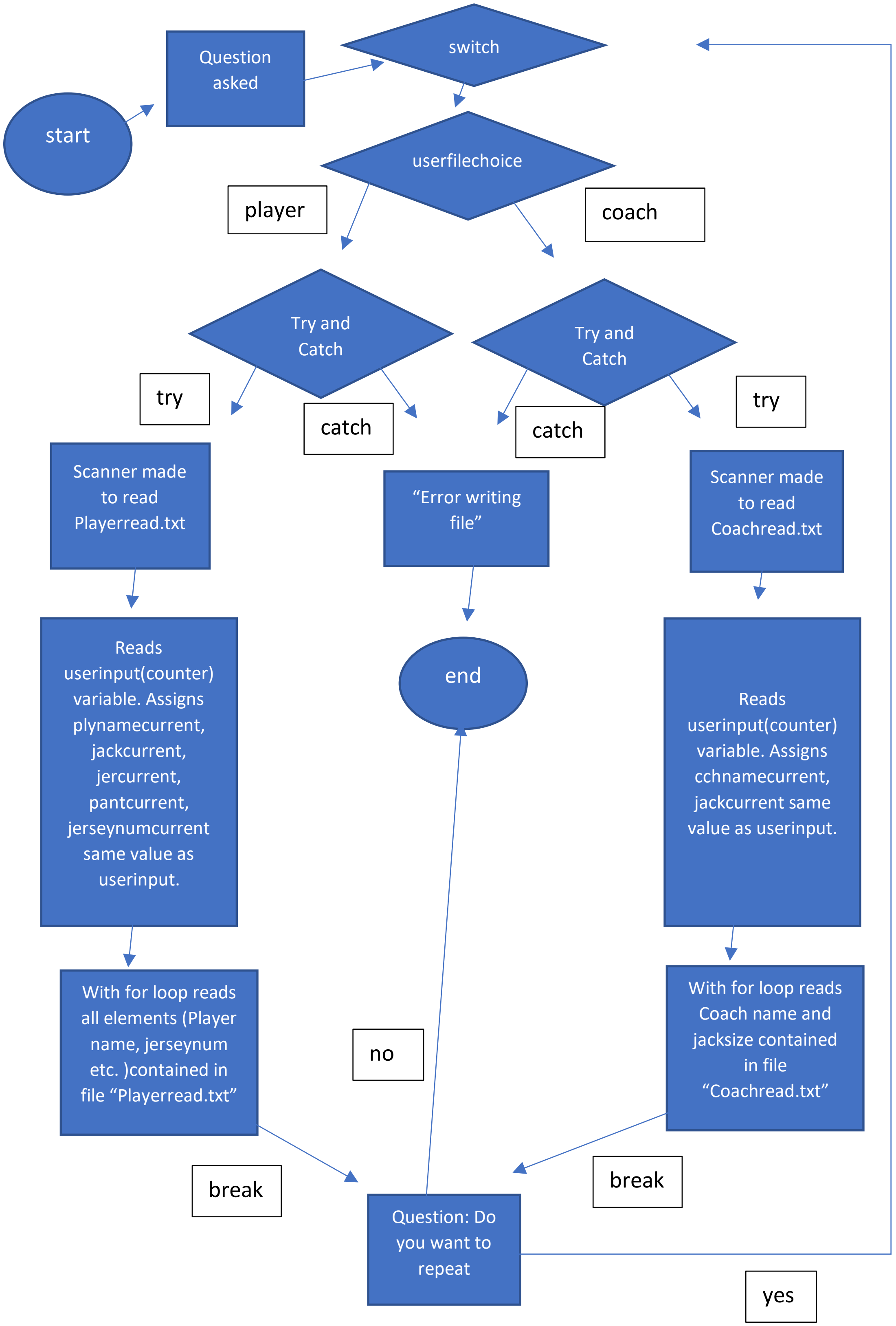
Delete flowchart:


start
Question asked
switch
Coach or Player
Player
Coach
Neither cch or ply
Question: Which Player delete?
If found
found
!found
Replace name with last element, userinput-1 and plynamecurrent-1
Replace jernum with last element, jerseynumcurrent-1
Replace jersize with last element, jercurrent-1
Replace jacksize with last element, jacksize-1
Replace shortsize with last element, pantcurrent-1
Boolean plydel =true
plydel
false
true
"Invalid input"
break
"Deleting has finished"
break
Question: Do you want to repeat
Yes or no
yes
no
end
Question: Which Coach delete?
If found
found
!found
Replace name with last element, userinput-1 and cchnamecurrent-1
Replace jacksize with last element, jacksize-1
Boolean cchdel =true
cchdel
false
"Invalid input"
"Deleting has finished"
break
break

Write to file flowchart:



Read from file flowchart:


start
Question asked
switch
userfilechoice
player
coach
Try and Catch
Try and Catch
try
catch
catch
try
Scanner made to read Playerread.txt
"Error writing file"
Scanner made to read Coachread.txt
Reads userinput(counter) variable. Assigns plynamecurrent, jackcurrent, jercurrent, pantcurrent, jerseynumcurrent same value as userinput.
end
Reads userinput(counter) variable. Assigns cchnamecurrent, jackcurrent same value as userinput.
With for loop reads all elements (Player name, jerseynum etc.)contained in file "Playerread.txt"
no
With for loop reads Coach name and jacksize contained in file "Coachread.txt"
break
break
Question: Do you want to repeat
yes

Schedule for developing product

The product will be divided into three sections:

Section 1:

Firstly, the base class that is to be made is the Members class. Once, the functions and variables for this class is defined then all the other classes will fall into course. After this, the Persons class will be created that will contain the accessor and mutator methods.

Section 2:

Create the Coach and Player classes. This will become easier because the Members class will have already been created.

Section 3:

Create the Mainclass which will only govern which class the program will go to, based on the input of the user. This is required for the running of the program since classes are just “templates”. Therefore the test plan must be run after the Mainclass is created.

Test Plan

Action Test	Way of testing
Do while: test if the program goes into the loop	Run the program and see if the program prints a sample test output in the do while statement. Then, see if the program asks “do you want to repeat” at the end
Case 1: test if userchoice = “player” and userchoice = “coach” works	Type player or coach and see if error statement is produced
Case 1: see if program can add members, names, jersey num, jersize, jacksize, shortsize	See if program can add members. To do this printline(“Created member at i”) in the for loop. This will show if and where member is created. Also, see if all the other functions are working in Case 1(player) Then, try the same for coach as well.
Case 2: test if userchoice = “player” and userchoice = “coach” works	Same (above)
Case 2: see if edit case is working	See if editply and editcch are evaluating to true in the for loop. This checks if the program goes through the edit() function. Also, see if userinput matches all the other counter variables (ex. jackcurrent) by doing a printline.
Case 3: see if printout() function works	Do a printline in the printout() function to see if program is printing correctly
Case 4: test if userchoice = “player” and userchoice = “coach” works	Same(above)

Case 4: test the delete case	See if delply and delcch are evaluating to true in the for loop. Also, see if userinput matches the other counter variables.
Case 5: Are files written	Printline at the end of the file write to see if the whole function has been executed. Look at the workspace and see if file has been created with all elements.
Case 6: Are files read	Printline at the end of file saying “got it from player” and if no exception is thrown then file is read. Try a printout().