David Martin

Project Proposal

The summer camp inspired this one. At the moment I am considering making a summer-camp registration program which will ask for basic info like the camper’s name, gender, and grade. At registration the camper will be asked to choose which, of six different hobbies, would best describe them. They will also be asked if they have any friends they would like to be in a cabin with. These four bits of camper info will be saved into a class and use file I/O. All campers will be split into six teams based on their grade, gender, and hobby-interest for the sake of balancing them. Campers will also be split into cabins based on their friends, grade, and gender; if they have no friend preference, they will be placed into the emptiest cabin of their grade. The program will also have an option to generate results for the camp roster showing their name, grade, team, and cabin number.

Project Specification

The program will include file I/O to compile a list of campers and it will max out at 200.

It will include two classes: a camper class consisting of their name, gender, grade, team, cabin number, and hobby.

The other will be a cabin class to pair a vector of 24 cabins with campers and their friends, according to average grade and gender. It will consist of get-functions that call this info, and it will have a function which will assign the average grade of the kids to the cabin so that children in different grades have the option to bunk with each other while allowing friendless kids to join cabins with space.

Class cabin

-int number

-int grade

-char gender

-camper camper

+getNumber()

+getGrade()

+getGender()

+setGrade(newGrade: int)

+setGender(newGender: char)

+calcAvgGrade()

Class camper

-name

-grade

-gender

-hobby

-friends[]

-team

-cabin

+getName()

+getGrade()

+getgender()

+getHobby()

+getFriends[()

+getTeam()

+getCabin()

+setName(newName: string)

+setGrade(newGrade: int)

+setGender(newGender: char)

+setHobby(newHobby: int)

+setFriends(newFriends: string)

+setTeam(newTeam: int)

+setCabin(newCabin: int)

If the roster is marked as complete, a function will be called which will organize kids into teams. There will be nested loops to do this. The outside loop will move through grades, and the inner one will work with interests. So, sixth graders who like technology will be placed into team red, then team orange, then yellow, until no more tech-loving sixth graders exist. Then sixth graders who like sports will be placed onto teams and so on until all sixth graders have been dealt with. Then the same thing will happen with seventh graders and so on until each grade is done.

After the six teams are organized, then a function will be called to create finished roster sorted by team which includes name, grade, cabin number, and team.