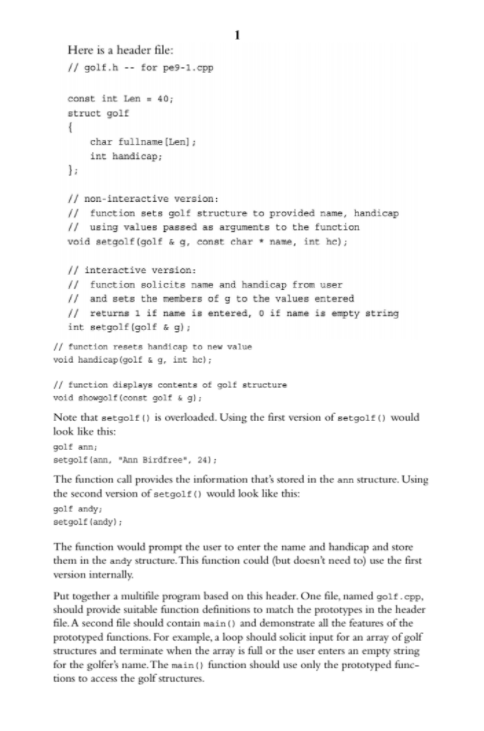
Rudenko Ruslan, SE-TE 2.01

Task:



Code:

MAIN.CPP

#include <iostream>

#include "golf.h"

const int NUM\_GOLF = 10;

int main() {

using namespace std;

golf golfers[NUM\_GOLF];

int numCollected = 0;

for (int i = 0; i < NUM\_GOLF; i++) {

cout << "Golfer #" << i + 1 << endl;

if (!setgolf(golfers[i])) {

break;

}

numCollected++;

}

cout << numCollected << " golfers were entered." << endl;

for (int i = 0; i < numCollected; i++) {

showgolf(golfers[i]);

}

golf tiger;

const char\* name = "Tiger Woods";

int hcap = 24;

setgolf(tiger, name, hcap);

showgolf(tiger);

handicap(tiger, 25);

showgolf(tiger);

return 0;

}

GOLF.H

#pragma once

#ifndef GOLF\_H\_

#define GOLF\_H\_

const int Len = 40;

struct golf {

char fullname[Len];

int handicap;

};

// non-interactive version:

// function sets golf structure to provided name, handicap

// using values passed as arguments to the function

void setgolf(golf& g, const char\* name, int hc);

// interactive version:

// function solicits name and handicap from user

// and sets the members of g to the values entered

// returns 1 if name is entered, 0 if name is empty string

int setgolf(golf& g);

// function resets handicap to new value

void handicap(golf& g, int hc);

// function displays contents of golf structure

void showgolf(const golf& g);

#endif /\* GOLF\_H\_ \*/

GOLF.CPP

#include "golf.h"

#include <cstring>

#include <iostream>

void setgolf(golf& g, const char\* name, int hc) {

strcpy(g.fullname, name);

g.handicap = hc;

}

int setgolf(golf& g) {

std::cout << "Enter name: ";

std::cin.getline(g.fullname, Len);

std::cout << "Enter handicap: ";

(std::cin >> g.handicap).get();

return strlen(g.fullname) == 0 ? 0 : 1;

}

void handicap(golf& g, int hc) {

g.handicap = hc;

}

void showgolf(const golf& g) {

std::cout << "Name: " << g.fullname << std::endl;

std::cout << "Handicap: " << g.handicap << std::endl;

}

Results

