CISS245: Advanced Programming
Quiz q6002

Name: cadalebout1@cougars.ccis.edu   Score:

Open `main.tex` and enter answers (look for `answercode`, `answerbox`, `answerlong`).
Turn the page for detailed instructions. To rebuild and view pdf, in bash shell execute `make`. To build a gzip-tar file, in bash shell execute `make s` and you'll get `submit.tar.gz`.

Q1. Write a class `Time` that has private members `hh_`, `mm_`, `ss_`, for hours, minutes, seconds in 24-hour format, i.e., hours is in the range 0..23. Write enough public methods so that you can create an array `t` of 10 `Time` objects so that `t[0]` models the time 00:00:00, `t[1]` models the time 00:01:00, `t[2]` models the time 00:02:00, ... `t[9]` models the time 00:08:00. Print all the object in `t`.
ANSWER:

```cpp
#include <iostream>
#include <iomanip>

class Time
{
public:
    Time(int h = 0, int m = 0, int s = 0)
    :hh_(h), mm_(m), ss_(s)
    {}

    void setHHMMSS(int hh, int mm, int ss)
    {
        hh_ = hh;
        mm_ = mm;
        ss_ = ss;
    }

    void println()
    {
        std::cout << std::setw(2) << std::setfill('0') << hh_ << ':'
                  << std::setw(2) << std::setfill('0') << mm_ << ';'
                  << std::setw(2) << std::setfill('0') << ss_ << std::endl;
    }

private:
int hh_, mm_, ss_;
};

int main()
{
```

```
    Time t[10];// Declare t to be an array of 10 Time objects

    for(int i = 0; i < 10; ++i)
    {
        t[i].setHHMMSS(0,i,0);
        t[i].println();
    }

    return 0;
}
```

Q2. This is the same as the previous question *except* that `t` is an array of `Time` pointers. You can copy the `Time` class from above to the answer here.
ANSWER:

```cpp
#include <iostream>
#include <iomanip>

class Time
{
public:
    Time(int h = 0, int m = 0, int s = 0)
    :hh_(h), mm_(m), ss_(s)
    {}

    void setHHMMSS(int hh, int mm, int ss)
    {
        hh_ = hh;
        mm_ = mm;
        ss_ = ss;
    }

    void println()
    {
        std::cout << std::setw(2) << std::setfill('0') << hh_ << ':'
                  << std::setw(2) << std::setfill('0') << mm_ << ';'
                  << std::setw(2) << std::setfill('0') << ss_ << std::endl;
    }

private:
int hh_, mm_, ss_;
};

int main()
{
    Time *t = new Time[10];// Declare t to be an array of 10 Time objects

    for(int i = 0; i < 10; ++i)
    {
        t[i].setHHMMSS(0,i,0);
```

```
        t[i].println();
    }
    delete[] t;
    return 0;
}
```

Q3. Wumpus lives in the Wumpus world, which is a 4-by-4 grid. For instance in the following

```
+-+-+-+-+
| | | | |
+-+-+-+-+
| | | |W|
+-+-+-+-+
| | | | |
+-+-+-+-+
| | | | |
+-+-+-+-+
```

Wumpus is at row 1, column 3. You are given this:

```
// file: WumpusWorld.h
#ifndef WUMPUSWORLD_H
#define WUMPUSWORLD_H

#include <iostream>

class WumpusWorld
{
public:
    init();             // initialize so that Wumpus is at (0,0).
    println();          // print according to the above format
    move_wumpus();      // randomize a direction rand() % 4 where 0,1,2,3
                        // are N,S,E,W for Wumpus. Of course Wumpus
                        // must stay in the world.
                        // If wumpus tries to move N but its row is 0,
                        // wumpus stays put.
private:
    char world_[4][4];
};
#endif
```

Do not add anything to the above. You want to make Wumpus wonder around like this:

```
#include <iostream>
#include <ctime>
#include <cstdlib>
#include "WumpusWorld.h"

int main()
{
    srand((unsigned int) time(NULL));

    WumpusWorld ww;
    ww.init();
    ww.println();
    for (int i = 0; i < 5; ++i)
    {
        ww.move_wumpus();
        ww.println();
    }
    return 0;
}
```

Complete the following cpp file that contains the implementation of the methods declared in the WumpusWorld class.

ANSWER:

```
// file: WumpusWorld.cpp

void WumpasWorld::init()
{
    for(int i = 0; i < 4; ++i)
    {
        for(int j = 0; j < 4; ++j)
        {
            world_[i][j] = ' ';
        }
    }
    world_[0][0] = 'W';
}

void WumpasWorld::move_wumpas()
{
        int direction = rand() % 4;

        switch (direction)
        {
            case 0://north
                for(int i = 1; i < 4; ++i)
                {
                    for(int j = 0; j < 4; j++)
                    {
                        if(world_[i][j] = 'W')
                        {
```

```
                                    world_[i][j] = " ";
                                    world_[i-1][j] = 'W';
                                }
                            }
                        }
                        break;
                    case 1:
                        for(int i = 0; i < 3; ++i)
                        {
                            for(int j = 0; j < 4; j++)
                            {
                                if(world_[i][j] = 'W')
                                {
                                    world_[i][j] = " ";
                                    world_[i+1][j] = 'W';
                                }
                            }
                        }
                        break;
                    case 2:
                        for(int i = 0; i < 4; ++i)
                        {
                            for(int j = 1; j < 4; j++)
                            {
                                if(world_[i][j] = 'W')
                                {
                                    world_[i][j] = " ";
                                    world_[i][j-1] = 'W';
                                }
                            }
                        }
                        break;
                    case 3:
                        for(int i = 0; i < 4; ++i)
                        {
                            for(int j = 0; j < 3; j++)
                            {
                                if(world_[i][j] = 'W')
                                {
                                    world_[i][j] = " ";
                                    world_[i][j+1] = 'W';
                                }
                            }
                        }
                        break;
            }
}

void WumpasWorld::println()
{
    std::cout << "+-+-+-+-+\n";
```

```
    for(int i = 0; i < 4; ++i)
    {
        for(int j = 0; j < 4; ++j)
        {
            std::cout << '|' << world_[i][j];
        }
        std::cout << "|\n";
        std::cout << "+-+-+-+-+\n";
    }
}
```

Instructions

In `main.tex` change the email address in

```
\renewcommand\AUTHOR{jdoe5@cougars.ccis.edu}
```

to yours. In the bash shell, execute "`make`" to recompile `main.pdf`. Execute "`make v`" to view `main.pdf`. Execute "`make s`" to create `submit.tar.gz` for submission.

For each question, you'll see boxes for you to fill. You write your answers in `main.tex` file. For small boxes, if you see

```
1 + 1 = \answerbox{}.
```

you do this:

```
1 + 1 = \answerbox{2}.
```

`answerbox` will also appear in "true/false" and "multiple-choice" questions.

For longer answers that needs typewriter font, if you see

```
Write a C++ statement that declares an integer variable name x.
\begin{answercode}
\end{answercode}
```

you do this:

```
Write a C++ statement that declares an integer variable name x.
\begin{answercode}
int x;
\end{answercode}
```

`answercode` will appear in questions asking for code, algorithm, and program output. In this case, indentation and spacing is significant. For program output, I do look at spaces and newlines.

For long answers (not in typewriter font) if you see

```
What is the color of the sky?
\begin{answerlong}
\end{answerlong}
```

you can write

```
What is the color of the sky?
\begin{answerlong}
The color of the sky is blue.
\end{answerlong}
```

For students beyond 245: You can put LaTeX commands in `answerbox` and `answerlong`.

A question that begins with "T or F or M" requires you to identify whether it is true or false, or meaningless. "Meaningless" means something's wrong with the statement and it is not well-defined. Something like "$1+_2$" or "$\{2\}^{\{3\}}$" is not well-defined. Therefore a question such as "Is $42 = 1+_2$ true or false?" or "Is $42 = \{2\}^{\{3\}}$ true or false?" does not make sense. "Is $P(42) = \{42\}$ true or false?" is meaningless because $P(X)$ is only defined if X is a set. For "Is $1 + 2 + 3$ true or false?", "$1 + 2 + 3$" is well–defined but as a "numerical expression", not as a "proposition", i.e., it cannot be true or false. Therefore "Is $1 + 2 + 3$ true or false?" is also not a well-defined question.

When writing results of computations, make sure it's simplified. For instance write 2 instead of $1 + 1$. When you write down sets, if the answer is $\{1\}$, I do not want to see $\{1, 1\}$.

When writing a counterexample, always write the simplest.

Here are some examples (see `instructions.tex` for details):

1. T or F or M: $1 + 1 = 2$ .... $\boxed{\text{T}}$

2. T or F or M: $1 + 1 = 3$ .... $\boxed{\text{F}}$

3. T or F or M: $1+^2 = $ .... $\boxed{\text{M}}$

4. $1 + 2 = \boxed{3}$

5. Write a C++ statement to declare an integer variable named `x`.

   ```
   int x;
   ```

6. Solve $x^2 - 1 = 0$.

   Since $x^2 - 1 = (x - 1)(x + 1)$, $x^2 - 1 = 0$ implies $(x - 1)(x + 1) = 0$. Therefore $x - 1 = 0$ or $x = -1$. Hence $x = 1$ or $x = -1$.

7. Which is true? .... $\boxed{\text{C}}$
   (A) $1 + 1 = 0$
   (B) $1 + 1 = 1$
   (C) $1 + 1 = 2$
   (D) $1 + 1 = 3$
   (E) $1 + 1 = 4$