Data Structures
BITL1

Worksheet 3

Fall 2020

17 September 2020

1. **Warm up:** Consider the following uncompiled `C++` code.

database.cpp

```
#include <iostream>
#include <string>
using namespace std;
using std::string;
#include "header.h"

int main() {
   Bank Trusty;
   Employee Dusty;
   ...
   return 0;
}
```

header.h

```
struct Bank {
   string name;
   int employeeNum;
   Bank();
}
Bank::Bank() {employeeNum = 0;}

class Employee {
   string name;
   Bank employer;
   Employee();
}
Employee::Employee() {}
```

Of the following options to place in the line `...`, which will produce an error when `database.cpp` is compiled and which will not?

(a) `cout << Trusty.employeeNum << endl;`

(b) `cout << Trusty.name << endl;`

(c) `Trusty.name = "Musty"; cout << Trusty.name << endl;`

(d) `Dusty.name = "Gutsy"; cout << Dusty.name << endl;`

(e) `Dusty.employer = Trusty;`

2. This question will have you create a class that reflects the properties of a matrix.

(a) Make a struct called `Matrix` that has the following in its definition:

- a private object `m` which is an integer representing the number of rows
- a private object `n` which is an integer representing the number of columns
- a public object `data` which is an array of length mn representing the entries
- a public function `replace` that replaces one entry of the matrix with some other value. This function takes in three arguments:
  - the first two specify the row and column position to be replaced, respectively
  - the third argument is the new value that replace the matrix's current value at the given position

Be sure to include a constructor and a destructor. You may asume that entries of the matrix are of type `float`.

(b) Make a function `addMatrices` that first checks if two matrices are of the same size, and if they are, returns a matrix that represents the sum of the two matrices.

(c) Make a function `multiplyMatrices` that first checks if two matrices are of the appropriate sizes, and if they are, returns a matrix that represents the product of the two matrices.

(d) Write code that, in order:

- initializes a 2 × 3 matrix $\left[\begin{smallmatrix} 1 & 2 & 3 \\ -2 & 0 & 7 \end{smallmatrix}\right]$ and a 3 × 2 matrix $\begin{bmatrix} 10 & -10 \\ 0 & 2 \\ -1 & -1 \end{bmatrix}$.
- replaces all the zeros with 999
- multiplies the two matrices together
- prints out the elements on the diagonal of the product of the two matrices