Northern Michigan University (Marquette Co, MI)

CS201-01-22F: Programming In C++ (Andrew A Poe) Name: _____
Midterm Examination (Exam 1) Page 1/2 Friday 14 October 2022 11:00 A.M. EDT

Time: 50 minutes

For the following programs, write the code as directed. Do not worry about #include files or minor syntactic errors such as semicolons or matching braces. However, your code should be as correct as possible. Make sure all special cases are handled correctly. Make sure you call and use all methods correctly. If you write more than one method to solve a problem, make sure you declare your headers correctly, but you don't have to worry about prototypes.

1. Write the code for the method

```
bool Palindrome (string word);
```

This method prints nothing. This method returns true if word is a palindrome (same backward as forward) and false if it is not. The trick is that you should consider lowercase letters identical to their uppercase equivalents. "Bob" is a palindrome, for example. You may assume that only letters will appear in word.

[If char c is an uppercase letter, you can get its lowercase equivalent with (char)(c+32). If c is a lower case letter, you can get its uppercase equivalent with (char)(c-32).]

```
bool Palindrome (string word) {

 for (int i=0; i < word.length(); i++)
  if (word[i] >= 'a' && word[i] <= 'z')
   word[i] -= 32;
 for (int i=0; i < word.length()/2; i++)
  if (word[i] != word[word.length()-i-1)
   return false;
 return true;
}
```

2. Write the code for the method

```
int **BinaryMatrix (int r, int c);
```

This method creates a matrix of integers with r rows and c columns and initializes them with an alternating pattern of zeroes and ones, as demonstrated below:

r = 3; c = 4;

0 1 0 1
1 0 1 0
0 1 0 1

Notice that the upper left corner is always zero.

```
int ** BinaryMatrix (int r, int c) {
```

```
 int **A = new int * [r];
 for (int i = 0; i < r; i++) {
  A[i] = new int[c];
  for (int i=0; i < c; i++)
   A[i][j] = (i+j)%2;
 }
 return A;
}
```

3. Write code for the method

```
bool InOrder ();
```

This method chooses five random integers from 1 to 100 and returns true if they are in increasing OR in decreasing numerical order. If the five integers are 12, 23, 28, 36, 97, it returns true. If they are 54, 24, 13, 12, 30, it returns false. Adjacent identical numbers are deemed to be in order, both increasing and decreasing. 2,3,3,3,4 is TRUE.

```
bool InOrder () {

 int a = rand()%100+1, b = rand()%100+1, c = rand()%100+1, d = rand()%100+1,
     e = rand()%100+1;

 if (a <= b && b <= c && c <= d && d <= e) return true;
 if (a >= b && b >= c && c >= d && d >= e) return true;
 return false;
}
```

4. Write the following method:

```
int CountABBA (string fname);
```

This counts and returns the number of times the line ABBA (all caps) appears in the file whose name is fname. In other words, how many times does the word ABBA appear all by itself on a line in the file?

```
int CountABBA (string fname) {

 ifstream in (fname);
 int ct = 0;
 string line;
 while (getline (in,line)) {
  if (line=="ABBA") ct++;
 in.close ();
 return ct;
}
```