Directions: Read the problems carefully. *Circle* **all** correct answers to multiple choice questions. Explain "why" if you *circle* FALSE. Fill in **all** blanks (unless you decide a T/F is FALSE). *Circle* the answer you feel most appropriately fills the blank if choices are given. Answer to the best of your ability.

This is a take-home test. You may ask **ME** questions for clarification — *not* each other. Feel free to attach pages to the end of the test if you need to continue a long answer. Please keep in mind that questions that ask you to write code are intended to be done without the aid of the compiler.[1] Take your time, but don't lose track of how much time is left.

Good luck! Have fun!

1) When dividing any number by `n`, the remainders are always in [_____ .. _____] (note the discrete interval notation!). This operation (mathematically known as *modulo*; and its divisor as the *modulus*) is represented in C++ by the _____ (_____) operator. You can only use this operator on _____-typed expressions (i.e. no decimal places). Such math is often used in _____.

   A) digit [group] extraction
   B) divisibility testing
   C) interest calculations
   D) random number generation
   E) unit conversions

2) Find[2] any errors you can in the following code fragment (they may be logic or syntax — but **NOT** stylistic errors). (Tip: This is a code *fragment*! Any errors we are interested in will *not* be about things that would have happened before or will happen after these lines!)

```
min_approx = flooring( min_exact
                       + 15
                       / .5 )
             - 15;
```

3) Explain what is wrong with the following code fragment? (See the tip back in #2. In that vein, it has ***NOTHING*** to do with the comment. The comment merely tells what else would have gone on between the variable declarations and the assignment statement.)

```
double rate;
long start_popul, end_popul;

// fill in start and end _popul with data
// (...probably using a cin)

rate = end_popul / start_popul;
```

Now **tell** me the name of the method we use to fix this situation and **show** how it would be applied here.

[1] I can't keep you from using it, but try to do it on your own first. . . please?! *big sad puppy dog eyes*
[2] ...and explain how you would fix...

4) Which of the following are errors in the following program? An error may be syntax, logic, or even style (indention, spacing, identifiers, etc.).

```
#include<iostream>
using namespace std;int main(){int x,y,z;double w;cin>>
x;y=x/12;z=x%12;w=x/12.0;cout<<x<<'='<<y<<'&'<<z<<'('<<w<<')'
<<endl;return 0;}
```

A) spacing is needed around operators
B) change `int` variables to `double`
C) add comments
D) make `12` a `short` `constant`
E) wrap any long lines properly
F) change single quotes to double quotes
G) prompt before `cin`
H) clean up labeling on results
I) typecast on `w` calculation
J) indent lines between { and }
K) `void` in `main`'s `()`
L) blank lines between logical sections/at logical breaks
M) fix variable names (it's an *inches* conversion program)
N) change `int` variables to `short` or `long`
O) join last three lines and then break after `;`, {, }, and `()` of `main`

If the user enters 43, what will the program produce (as currently written)?