MAT 305 TEST 1

Directions: Solve these problems. You may write on this paper, but I will not read it. Problems are not weighted equally. Show all necessary work: **computations that are not obvious must be shown**. As for what is "obvious", better safe than sorry!

True/False. If false, provide a true statement.

1. Computer science is the study of computers. *False: Computer science is the study of what can be computed.*
2. The syntax of a language is its meaning, and semantics is its form. *False: syntax is its form, while semantics is its meaning.*
3. A loop is used to skip over a section of a program. *False: A loop is used to* ***repeat*** *a section of a program.*
4. An algorithm can be written without using a programming language. *True*
5. Programs no longer require modification after they are written and debugged. *False: Programs often undergo modification; for instance, to add new features.*
6. In mathematics, $x = x + 1$ is a true statement. *False:* $x = x + 1$ *is a false statement.*
7. In Sage and Python, `x=x+1` is a valid statement. *True (it's an assignment).*
8. A Python string literal is always enclosed in double quotes. *False: a string literal can be enclosed in single quotes.*
9. In Python, `''4''+''5''= ''9''`. *False:* `''4''+''5''=''9''`. *(I didn't notice the problem with the quotes on the original.)*
10. ASCII is a standard for representing characters using numeric codes. *True.*
11. Sage code and Python code are always interchangeable. *False: Python's* `input()` *command won't work in a Sage worksheet, but more importantly, most useful Sage code won't run in Python.*
12. Sage consists of state-of-the-art software written by the world's leading mathematicians for the sake of their own research. *True.*
13. Sage is "free as in beer" because its development was paid for by taxpayers. *True.*
14. All computer algebra systems let you look at their computer code, making them "free as in speech." *False: Commercial systems like Mathematica, Maple, etc. do not allow you to look at their computer code.*
15. Mathematics is a tool for solving problems. *True.*

Multiple choice.

1. What is the fundamental question of computer science?
   A. How fast can a computer compute?
   B. What can be computed?
   C. What is the most effective programming language?
   D. How much money can a programmer make?
2. An algorithm is like a
   A. newspaper
   B. venus flytrap

C. drum
D. recipe

3. A problem is **intractable** when
   A. you cannot reverse its solution.
   B. it involves tractors. (I think the book's author gets a little bored sometimes.)
   C. it has many solutions.
   D. it is not practical to solve.
4. One difference between a compiler and an interpreter is
   A. a compiler is a program.
   B. a compiler is used to translate high-level language into machine language.
   C. a compiler is no longer needed after the program is translated.
   D. a compiler processes source code.
5. The process of describing exactly *what* a computer program will do to solve a problem is called
   A. design
   B. implementation
   C. programming
   D. specification
6. Which of the following are *not* allowed in *Python and Sage* expressions?
   A. variables
   B. statements *(an expression is allowed in a statement, but not vice versa; this is a big difference with C and C++, which allow insanely subtle, hard-to-catch bugs like* `if (a=2) { /* do something */ }`*)*
   C. operators
   D. literals
7. Fragments of code that produce or calculate new data values are called
   A. identifiers
   B. expressions
   C. productive clauses
   D. assignment statements
8. Accessing a single element, such as a character, from a list or a string is called
   A. slicing
   B. concatenation
   C. assignment
   D. indexing
9. If `C` is a collection or a string, which of the following is the same as `C[0:-1]`?
   A. `C[-1]`
   B. `C[:]`
   C. `C[:len(C)-1]`
   D. `C[0:len(C)]`
10. Which of the following commercial systems is an example of the type of mathematics Sage specializes in?
    A. Maple *(mentioned repeatedly in class as a computer algebra system)*
    B. R *(statistics)*
    C. Matlab *(numerical analysis and numerical linear algebra)*
    D. LaTeX *(typesetting mathematics, not computation)*

Discussion

1. What is wrong with the following loop?
```
for c in C:
  C.append(c+1)
```
The loop is infinite. Despite having the form of a definite loop, it modifies the list which controls it.
2. How would I define a *function* in Sage that held the expression $3x^3-1$, and for which I could evaluate at $x=5$ using the usual notation $f(5)$?
`f(x) = 3*x^3 - 1` *or* `f(x) = 3*x**3 - 1`
3. How does Sage tell me that a system of linear equations has *no* solution?
It returns an empty list.
4. How does Sage tell me that a system of linear equations has *infinitely many* solutions?
It returns a list of parameterized values, where the parameter is something like `r1`, `r2`, *etc.*
5. If a system of linear equations has exactly one solution for x, y, and z, how would I extract from Sage the numerical value of x, *after* solving the system by typing `sols = solve(`[system of equations]`,[x,y,z])`?
`sols[0][0].rhs()`
You need the name of the list of solutions, which is `sols`. *This is* ***a list of lists****, in the form* `[[x=...,y=...,z=...]]`. *The first* `[0]` *obtains the first list of solutions* `[x=...,y=...,z=...]`. *The second* `[0]` *obtains the first element of that list, which is the* ***equation*** `x=...`. *To extract the* ***numerical*** *value of* x*, we need the dot-command* `.rhs()`*, which abbreviates "right-hand side."*