Basics of Algorithms

- Computers are devices that do only one kind of thing:
  They carry out *algorithms* to process information.

- To computer scientists, the algorithm is the central unifying concept of computing, the mode of thought that is the core of the computing perspective.

What is an algorithm?

- A set of logical steps to accomplish a task.
- A "recipe of action".
- A way of describing *behavior.*

Everyday Algorithms

Chocolate Chip Cookies

Ingredients:

2 ¼ cups flour	1 tsp salt
1 tsp baking soda	2 eggs
¾ cup brown sugar	1 tsp vanilla extract
¾ cup granulated sugar	1 cup soft butter

12 oz. semi-sweet chocolate chips

Steps:

Preheat oven to 375 degrees
Combine flour, salt, baking soda, in bowl. Set mixture aside.
Combine sugars, butter, vanilla and beat until creamy.
Add eggs and beat.
Add dry mixture and mix well.
Stir in chocolate chips.
Drop mixture by teaspoons onto ungreased cookie sheet.
Bake 8 to 10 minutes.

What is wrong with the following algorithm? (From the back of a shampoo bottle.) Directions: Wet hair. Apply a small amount of shampoo, lather, rinse, repeat.

If you follow this algorithm, you will never finish washing your hair!

Algorithms in Computing

In the realm of computer algorithms, an algorithm is useful only if:

- The algorithm accepts input data (not all do, however).
- The algorithm processes that data in some fashion.
- The algorithm produces some output (the results).

However, to be a correct algorithm, it must correctly solve the problem for any valid input data. Also, for the same input data, it must always give the same answer. Invalid input data should produce an error message or some other indication that the algorithm cannot correctly solve the problem. It should not produce an answer when given incorrect data since the user will think that the answer is valid.

Successful algorithms must consider all possible cases presented by acceptable data. You will succeed more quickly at constructing algorithms if you make it a habit to:

- Think about the problem and its data, then
- Enumerate all the special cases that the algorithm must handle.

Describing Algorithms

In specifying behavior, an algorithm must be:

- Precise
- Unambiguous
- Complete
- Correct

There are various techniques that can be used to describe algorithms:

- Natural language (English)
- Pictures (flow-charts)
- Pseudocode or a specific programming language

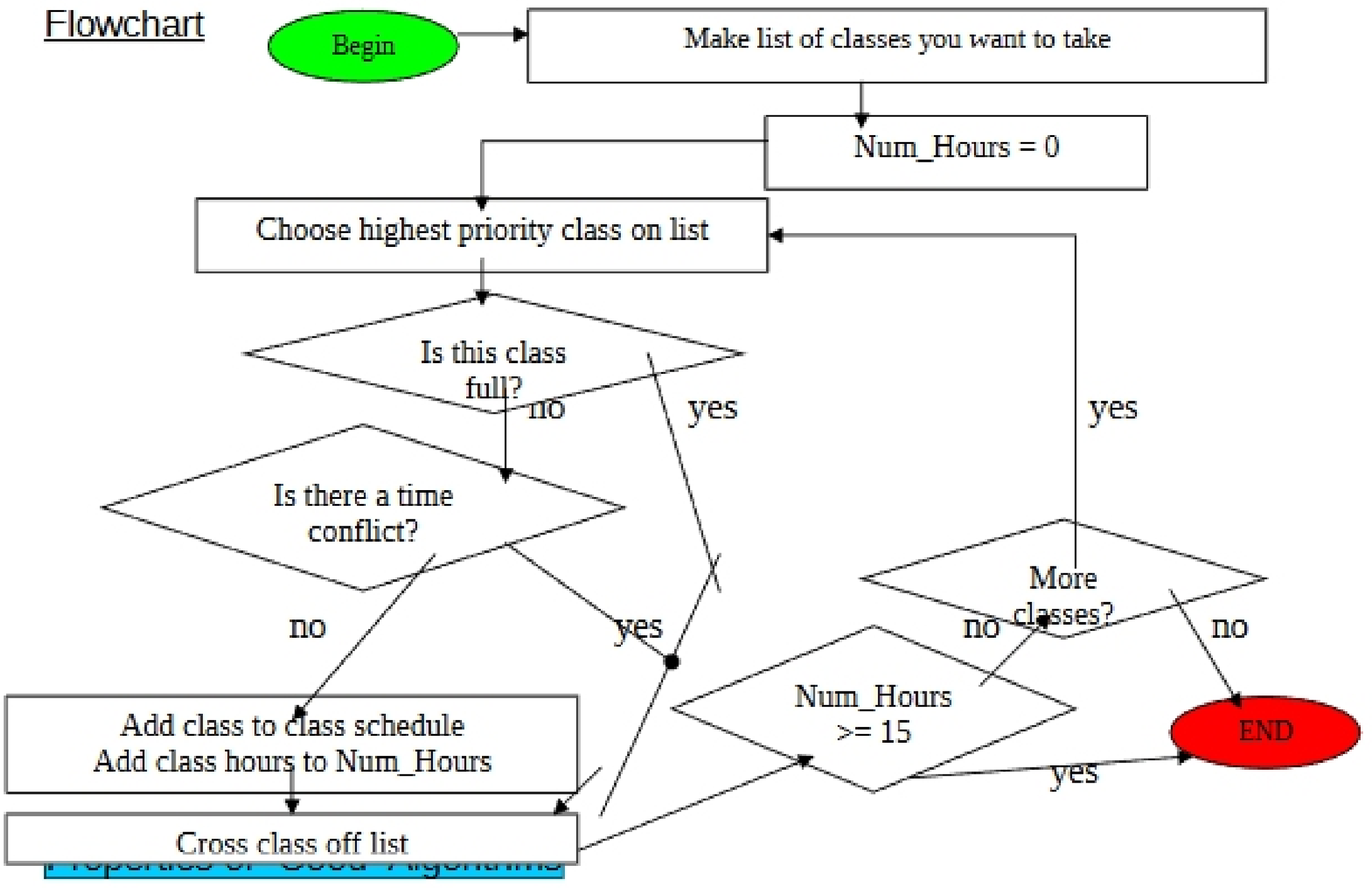
Example – Algorithm Representation

Consider an algorithm for registering for classes at UCF.

Natural Language Algorithm

1. Make a list of courses you want to register for, in order of priority.
2. Start with an empty schedule. Number of hours = 0.
3. Choose highest priority class on list.
4. If the chosen class is not full and its class time does not conflict with any class already in the schedule, then register for the class:
   4a. Add the class to the schedule.
   4b. Add the class hours to the number of hours scheduled.
5. Cross that class off your list.
6. Repeat steps 3 through 5 until the number of hours scheduled is >= 15, or until all classes have been crossed out.
7. Stop.

Flowchart



1. Precision