Name: Date:
Period:

Summary Sheet : Unit Testing, Integration Testing & User accepted testing.

- There are three main types of coding; Unit Testing, Integration Testing, and User Acceptance Testing.

- Unit Testing catches small logical errors.

- Integration Testing tests the bodies of code.

- User Acceptance Testing is the final display to the audience.

- Testing catches errors within the code, both large and small. It is an important part of the coding process. Because it allows the creator to more easily identify and remove errors.

BUP is good for reusing code which saves programmers time.

BUP is easier than top down because BUP groups together small systems into larger more complex systems

BUP sounds way cooler than top down

BUP is sometimes too generic for specific needs

Innovative classes are incredibly complex to create

The imp cost of the program is likely to be higher

Object Oriented Programming

Varun Ravichandran, Dylan Nezaj, Ruben DeMan, Kaleb Bogardus

What is it?

- Process where one represents a real world entity(object) and all its states and behaviors through a class.
- Objects of the same type are defined by the same class, and all states and behaviors are defined in the class.

History:

- First appearance in MIT.
- SIMULA was the first known language to incorporate it.
- Designed to created graphics-oriented apps at first. \

Objects:

- Entity in the real world that can be described in terms of a state and behavior.
- States are represented by fields and variables, properties and attributes.
- Behaviors are represented by methods.

Classes:

- The recipe/blueprint for all objects using that class.
- Different states w/in a class are defined by variables.

Inheritance:

- When an object or class is based off another class, using the same implementation.
- Maintains same behavior.

Polymorphism:

- References the programs ability to process objects differently depending on their data type or class.
- Handles values uniformly.

Example: Tic Tac Toe:

- See board for example of how this all works together.

Kevin Parker

Alex Rakowsky

Michael Kern

Object Oriented Programming

Key Concepts

October 20th, 2015

Period 7

AP Computer Science

- Object Oriented Programming is generally the idea of using data structures to make the program behave as the coder wants.
- A class is essentially a blueprint and another way to describe the object, or define its action.
- You can reference features amongst classes with the use of inheritance.
- You can create subclasses of an existing superclass using the syntax extends.
- Objects are composed of a method and state.
- Interface Segregation Principle - make interfaces that are client specific.
- The Single Responsibility Class - a class should have only one reason to change.
- A method will alter the state of an object.

Top-Down Design

Top down design is the process of breaking a large project into several smaller process; then breaking those smaller processes into even smaller processes.
Advantages:

- Having a design prior to coding helps ensure that code doesn't get off track.
- A large project can seem very daunting. By breaking it up into pieces, it will seem much more manageable.
- Breaking a project into pieces also streamlines the thought process. While the project as a whole may be very complicated, smaller parts of it will be much simpler.

Disadvantages:

- If specifications change, which is likely, it is possible that large amounts of code may need to be rewritten.
- Top down design generally creates very specific code that may not be reusable.
- Testing is very difficult. No meaningful tests can be done until large amounts of code are written.

Haiku

top down design helps

those who see a bigger view

catch the small details

Extreme Programming
Finn McManus, Corey Buckley, Logan Stone

The first example of extreme programming was _____'s payroll system in 1997.

The 12 principles of XProgramming:

_____:The desired features of the software which are communicated by the customer, are combined with cost estimates provided by the programmers to determine what the most important factors of the software are.

_____:The software is developed frequently in small stages.

_____:All members on a team use names and descriptions to guide development and communicate on like terms.

_____:Software should only contain code that is necessary to achieve the desired results by the customer at each stage in the process.

_____:Programmers test consistently in order to fulfill the requirements of the test and the customer provides "acceptance tests" to ensure the desired results are being achieved.

_____:Programmers improve the design of the software through every stage of development rather than waiting until the end then going back and correcting flaws.

_____:All code is written by pairs of programmers working on the same machine.

_____:Every line of code belongs to every programmer so there's no issues of "proprietary authorship".

_____:The team integrates and builds the software system multiples times a day to keep all programmers on the same stage of development.

_____:The programmers do NOT work excessive overtime to ensure the team stays well rested.

_____:The project is directed by the customer who is available to answer any questions and set priorities.

_____:Programmers all write in the same way which allows them to work in pairs and share ownership of the code.