1. **Simulation and Graphics**
   CST111—Introduction to Information Technology

2. **What Is Simulation?**
   - Simulation—a model of a complex system and the experimental manipulation of the model to *observe the results*
     - Complex systems which have multiple components and usually can be captured mathematically are *dynamic, interactive and complicated*
   - Helps answer questions like:
     - “Would materials flow through the manufacturing line if there were more space between stations?”
     - “Where is the optimal place to put the new fire station?”

3. **Models**
   - A model is an *abstraction* of a real system
     - It is not the system itself but rather a small subset of the features of the system to describe the behavior under that is under investigation
   - It is a representation of:
     - The objects within the system and ...
     - The rules that govern the interactions of the objects

4. **Continuous Simulation** *(Page 1)*
   - A model that treats time as *continuous*
     - Expresses changes in terms of a set of differential equations that reflect the relationships among the set of characteristics
   - Meteorological models falls into this category
     - Characteristics are wind component, temperature, water vapor, cloud formation, precipitation, etc.
     - Interactions of these components change over time and can be modeled mathematically over some three-dimensional region

5. **Continuous Simulation** *(Page 2)*
   - Continuous simulation is found inside Wii stations, commercial flight simulators, jet plane auto pilots, and advanced engineering design tools
   - Much of modern technology that we enjoy today would not be possible without continuous simulation

6. **Discrete-Event Simulation** *(Page 1)*
   - In discrete-event simulation, system operation is represented as a chronological sequence of events
     - Made up of *entities, attributes and events*
   - Each event occurs at an instant in time and marks a change of state in the system
     - I.e. if an elevator is simulated, an event could be “level 6 button pressed” with resulting system state of “elevator moving” and eventually (unless one chooses to simulate the failure of the elevator) “elevator at level 6”

7. **Discrete-Event Simulation** *(Page 2)*
   - Example—customers arriving at bank to be served:
     - The system entities are CUSTOMER-QUEUE and TELLERS
     - The system events are CUSTOMER-ARRIVAL and CUSTOMER-DEPARTURE (the
After all design decisions have been finalized, programmers produce the code to create a rendering engine for graphics (uses 50% of the CPU’s processing power) and stations, commercial flight simulators, jet meteorological models fall into this category. The target game platform determines which service the programmer will use (some models for hurricane tracking are solved to define the values of the variables at some later time. These models are applied to a moving target, not replace him or her. Different models can give conflicting results)

Queuing Systems

Queuing system—a discrete-event model that uses random numbers to represent arrival and duration of events. The system is made up of servers (the objects that provide a service) and queues of objects to be served.

To construct a queuing model, we must know:
- The number of events and how they affect the system in order to determine the rules of entity interaction
- The number of servers
- The distribution of arrival times in order to determine if an entity enters the system
- The expected service time in order to determine the duration of an event

Queuing Systems

To construct a queuing model, we must know:

- The number of events and how they affect the system in order to determine the rules of entity interaction
- The number of servers
- The distribution of arrival times in order to determine if an entity enters the system
- The expected service time in order to determine the duration of an event

Meteorological Models

Meteorological models are based on the time-dependent partial differential equations of fluid mechanics and thermodynamics. Initial values for the variables are entered from observations, and the equations are solved to define the values of the variables at some later time. Variable characteristics are wind component, temperature, water vapor, cloud formation, precipitation, etc.

Computer models are designed to assist the weathercaster, not replace him or her. The outputs from the computer models are predictions of the values of variables in the future, but it up to the weathercaster to determine what the values mean. Output from the models is only as good as the inputs. Different models can give conflicting results.

Meteorological Models

Computer models are designed to assist the weathercaster, not replace him or her. The outputs from the computer models are predictions of the values of variables in the future, but it up to the weathercaster to determine what the values mean. Output from the models is only as good as the inputs. Different models can give conflicting results.

How To Read Computer Weather Models

http://www.youtube.com/watch?v=pQt9qCNfqv4

Hurricane Tracking

Models for hurricane tracking are relocatable models because they are applied to a moving target. That is the geographical location of the model’s forecast varies for each hurricane
17 ▶ Computational Biology
- An interdisciplinary field that applies techniques of computer science, applied mathematics, and statistics to problems in biology
- Encompasses bioinformatics, computational biomodeling, computational genomics, molecular modeling, and protein structure prediction
- Has lead to the complete mapping of the human genome as well as assisting in locating genes for many diseases

18 ▶ Graphics (Page 1)
- Computer graphics—the representation and manipulation of image data by a computer.
  - Originally a communications language of for engineers, designers, and architects

19 ▶ Graphics (Page 2)
- Some examples of computer graphics:
  - Graphical user interface (GUI) operating systems such as Windows and Mac OS
  - Computer-aided design (CAD) which uses computer systems with advanced graphics hardware and software to create precision drawings and technical illustrations
  - Artists who use “paint” programs, photographers who use image manipulation software, etc.
  - Scientific experimentation and simulation, and medicine
  - Computer games, animated films and special effects

20 ▶ Algor Simulation 2011 — AutoCAD and Autodesk Inventor
- http://www.youtube.com/watch?v=q12auaBa3rA&feature=related

21 ▶ Computer Gaming (Page 1)
- Computer gaming is a simulation of a virtual world which draws players as participants
  - An interactive, computer-generated environment
  - Often designed to resemble reality where real-world rules and laws apply, or they can be built in fantasy worlds where the rules do not apply

22 ▶ Computer Gaming (Page 2)
- To make people, objects, and environments behave realistically in a virtual world, game designers must have knowledge of:
  - Computer graphics
  - Artificial intelligence
  - Human-computer interactions and simulation
  - Software engineering
  - Computer security
  - Fundamentals of mathematics
  - Laws of physics relating to gravity, elasticity, light & sound

23 ▶ Gameplay
- Gameplay—defined as the type of interactions and experiences a player has during the game (the type of game)
- Game genres based on gameplay include:
  - Action games
  - Shooter games
- Action-adventure games
- Life-simulation games
- Role-playing games
- Strategy games

24 Creating the Virtual World  (Page 1)
- Game engine—a software system within which games can be created
- The following functionality should be provided by tools of a game engine:
  - A rendering engine for graphics (uses 50% of the CPU’s processing power)
  - A physics engine based on Newtonian physics to provide collision detection system and dynamics simulation to solve problems related to forces affecting simulated objects
  - A sound-generating component

24 Creating the Virtual World  (Page 2)
- The following functionality should be provided by tools of a game engine (con.):
  - *Scripting language* apart from the code driving the game
  - Animation
  - Artificial intelligence algorithms (e.g., path-finding algorithms) provide the “illusion” of intelligence
  - A scene graph that holds the *spatial representation* in a graphical sense

26 Game Programming
- [http://www.youtube.com/watch?v=4ea2N_O3m2M&feature=related](http://www.youtube.com/watch?v=4ea2N_O3m2M&feature=related)

27 Soft Skills  (Page 1)
- High quality game design and development requires effective use of “soft skills”:
  - Effective collaboration with designers, programmers, and artists on various technical ideas throughout the entire game design and development process
  - Flexibility and adaptability as game design constantly evolves and changes throughout the development and production process

27 Soft Skills  (Page 2)
- High quality game design and development requires effective use of “soft skills” (con.):
  - Willingness to abandon much of the completed design work when the game’s story line, mechanics, art, programming, audio, video, and/or scripting requires significant changes

29 Game Programming  (Page 1)
- After all design decisions have been finalized, programmers produce the code to create virtual world of the game
- Popular languages include C++, Java, and C
- Some well-established game engineers have created custom languages based on their games
  - E.g., Epic Game’s UnrealScript for the Unreal Game

30 Game Programming  (Page 2)
- A variety of application programming interfaces (APIs) and libraries are available to help developers with key programming tasks
  - The choice of API determines which vocabulary and calling conventions the
Coding process begins with creation of the “game loop” which is responsible for managing the game world (regardless of any input from the user).
- I.e. the game loop might update enemy movement in the game or check for victory/loss conditions
- Basically the game loop manages the simulation

31 Game Programming (Page 3)

Large design teams have different programmers focusing on different aspects of the game:
- A Junior Engine programmer might be writing and maintaining code for the game loop, or...
- A 3D Software programmer might be designing and implementing the 3D graphics component, or...
- A User-Interface programmer might be working on the APIs in the game engine

32 Game Programming (Page 4)

Collaborating effectively is essential to creating a streamlined, “killer” computer game. Despite beta testing and demoing, invariably bugs may begin to adversely affect the game operation.
- Therefore collaboration and cooperation are necessary
- For online games, fixes can be performed without having to interrupt the ongoing action or forcing the company to order a costly recall of the game