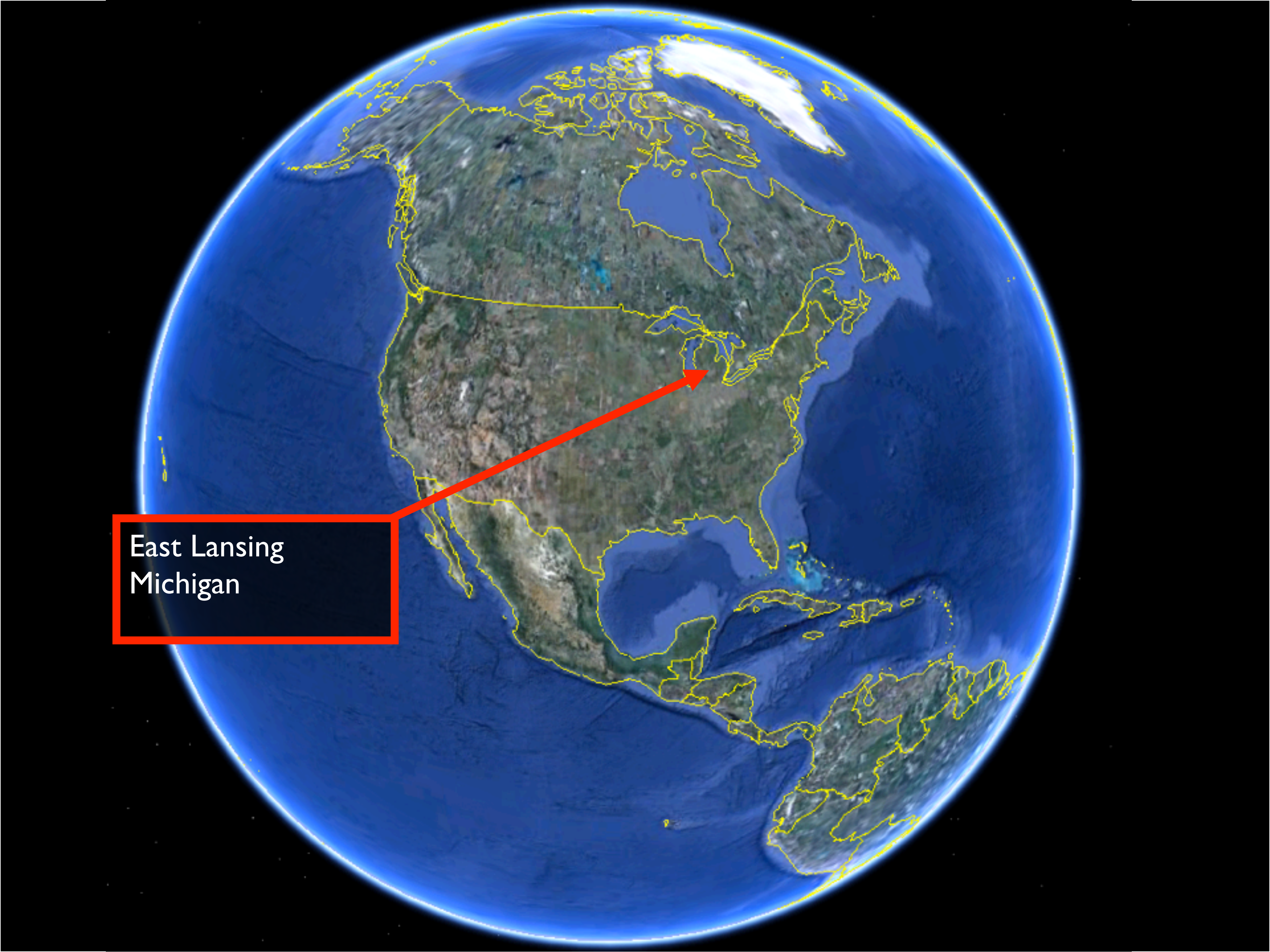
INTEGRATING CORE COMPETENCIES IN ENGINEERING EDUCATION

Wolfgang Bauer

Michigan State University


East Lansing
Michigan

MICHIGAN STATE UNIVERSITY

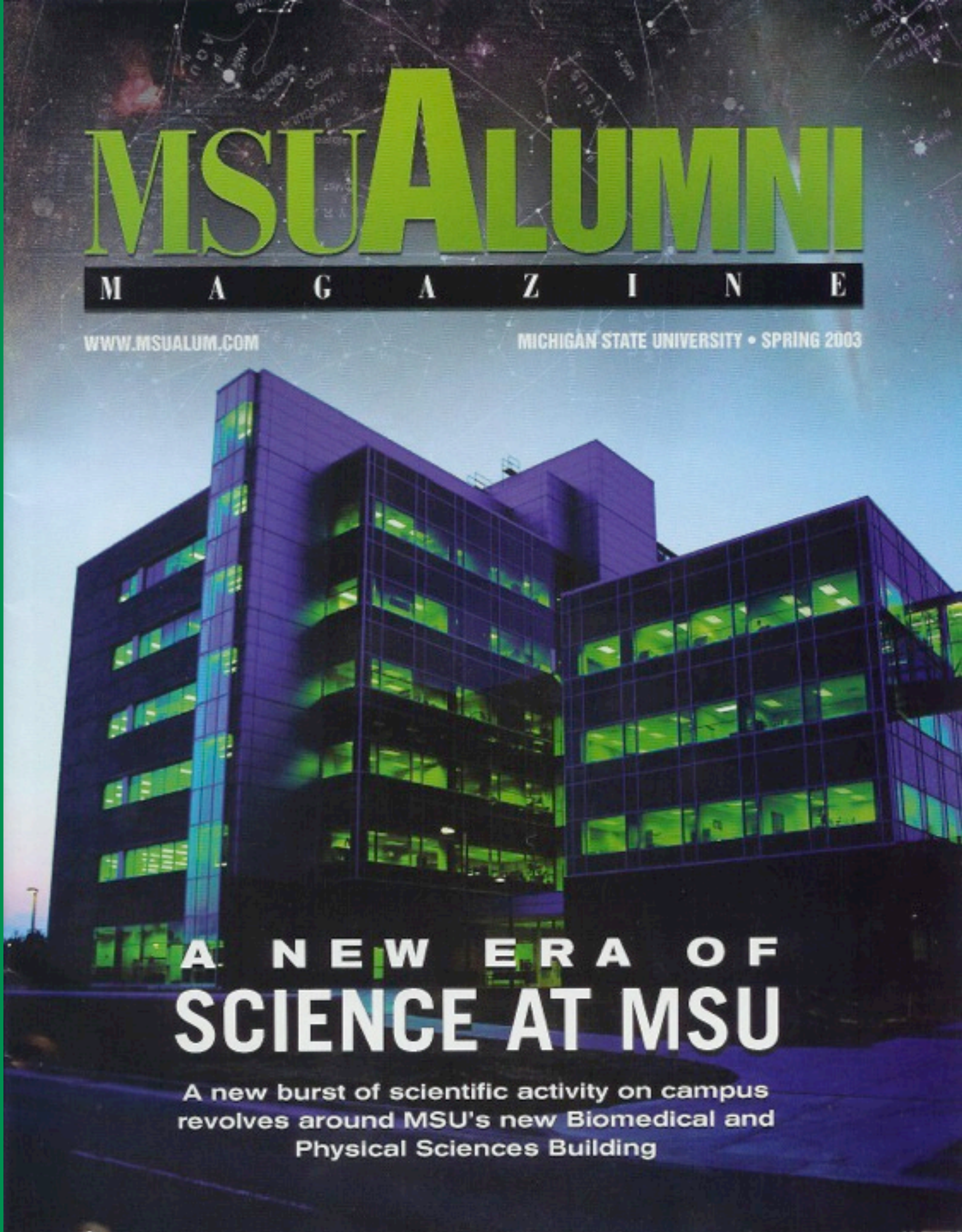
- Premier land-grant university (est. 1855)
  - 5,200-acre campus with 2,100 acres in existing or planned development
  - 577 buildings, including 83 with instructional space
- 36,500 undergraduate and 11,000 graduate student from 130 countries (53% women)
- ~430,000 living alumni
- ~5,000 faculty and academic staff
- 17 colleges, 200 different programs of study
- Top-100 university in the World

Department of Physics And Astronomy

Nuclear Physics (#1 in USA)
Particle Physics
Condensed Matter Physics
Astronomy

$8M annual budget
$30M federal grants

70 faculty (all ranks)



CONVENTIONAL CURRICULUM

- Calculus 1, 2, 3 – MSU Courses:
  - MTH132 (Calculus 1): Limits, continuous functions, derivatives and their applications. Integrals and the fundamental theorem of calculus.
  - MTH133 (Calculus 2): Applications of the integral and methods of integration. Improper integrals. Polar coordinates and parametric curves. Sequences and series. Power series.
  - MTH234 (Multivariable Calculus): Vectors in space. Functions of several variables and partial differentiation. Multiple integrals. Line and surface integrals. Green's and Stokes's theorems.
- Physics 1, 2, 3 – MSU Courses:
  - PHY183 (Physics for Scientists and Engineers 1): Mechanics, Newton's laws, momentum, energy conservation laws, rotational motion, oscillation, gravity, and waves.
  - PHY184 (Physics for Scientists and Engineers 2): Electricity and magnetism, electromagnetic waves, light and optics, interference and diffraction.
  - PHY215 (Thermodynamics and Modern Physics): Thermodynamics, atomic physics, quantized systems, nuclear physics, solids, elementary particles.
  - PHY191, PHY192 (Physics Lab for Scientists 1&2)

CONVENTIONAL CURRICULUM

- Computer Science 1, 2 – MSU Courses:
  - CSE231 (Introduction to Programming 1): Introduction to programming using Python. Design, implementation and testing of programs to solve problems such as those in engineering, mathematics and science. Programming fundamentals, functions, objects, and use of libraries of functions.
  - CSE232 (Introduction to Programming 2): Continuation of object-centered design and implementation in C++. Building programs from modules. Data abstraction and classes to implement abstract data types. Static and dynamic memory allocation. Data structure implementation and algorithm efficiency. Lists, tables, stacks, and queues. Templates and generic programming.

SHORTCOMINGS

- Students see different disciplines as not connected
- Concepts in mathematics are not applied to physics and engineering
- Only analytically solvable cases are addressed
- Numerical analysis is not connected to mathematics
- Computer skills are not applied to physics and engineering problems
- Real-world complications are ignored

NEW APPROACH: FYIE

First Year Integrated Engineering (FYIE)

- Replace Mathematics, Physics, and Computer Science Course by one integrating block course
- Problem-Based Learning (PBL) course
  - Central instructional unit is a problem, which needs to be solved by integrating the different disciplines
  - Not lecture-centered
  - Flipped classroom, using internet based lesson vignettes (5 – 10 minutes duration)
- SCALE-UP style classroom
  - Encourages collaboration
  - Students work in collaborative work teams

CORE TEAM (MICHIGAN STATE UNIVERSITY)

Bauer, Wolfgang	Physics
Bell, Bob	Mathematics
Briedis, Daina	Engineering
Esfahanian, Abdol	Engineering
Geier, Bob	Mathematics
Genik, Laura	Engineering
Grabill, Jeff	Writing
Hinds, Timothy	Engineering
Hjorth-Jensen, Morten	Physics (University of Oslo, Norway)
Idema, Amanda	Engineering
Keller, Brin	Mathematics
Punch, Bill	Computer Science
Sticklen, Jon	Computer Science
Syldic, Mary Anne	Evaluator (Western Michigan University)
Tessmer, Stuart	Physics
Urban-Lurain, Mark	Computer Science
Vergara, Claudia	Engineering
Walton, Pat	Engineering
Wolff, Tom	Engineering

TRAJECTORIES

- Mathematics
  - Differential Equations for motion with constant acceleration g

$$\frac{dv_x}{dt} = 0 \qquad \frac{dx}{dt} = v_x$$

$$\frac{dv_y}{dt} = -g \qquad \frac{dy}{dt} = v_y$$

  - Solution

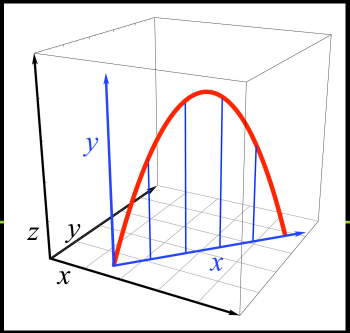
$$v_x(t) = v_{x0}$$

$$v_y(t) = v_{y0} - gt$$

$$x(t) = x_0 + v_{x0}t$$

$$y(t) = y_0 + v_{y0}t - \tfrac{1}{2}gt^2$$

TRAJECTORIES



- Physics
  - Live lecture demonstrations or videos
    - Independence of *x* and *y* motion
    - Projectile motion
    - Free-fall
  - Derivations and Extensions – Applications of Calculus
    - Parabolic trajectory

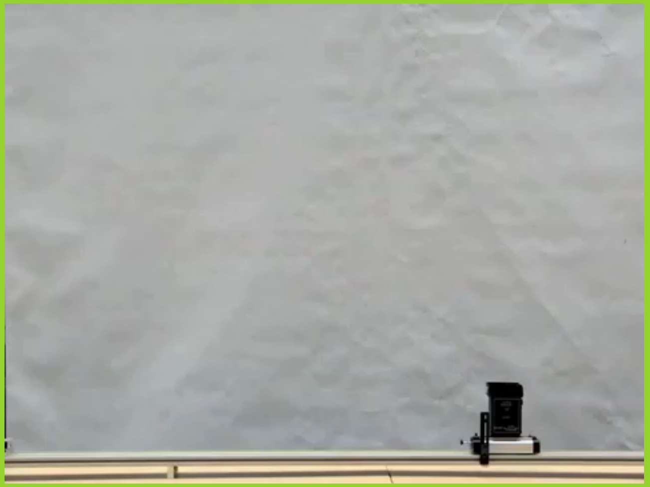
$$y = \left(y_0 - \frac{v_{y0} x_0}{v_{x0}} - \frac{g x_0^2}{2 v_{x0}^2} \right) + \left(\frac{v_{y0}}{v_{x0}} + \frac{g x_0}{2 v_{x0}^2} \right) x - \frac{g}{2 v_{x0}^2} x^2$$

    - Range of projectiles

$$R = \frac{v_0^2}{g} \sin 2\theta_0$$

    - Maximum height of projectiles

$$H = y_0 + \frac{v_{y0}^2}{2g}$$



TRAJECTORIES

- Interactive simulations (html5)

TRAJECTORIES

- E-Book

which is also the local slope of the flight path. At the top of the trajectory, the green and blue arrows are identical because the velocity vector has only an *x*-component—that is, it points in the horizontal direction.

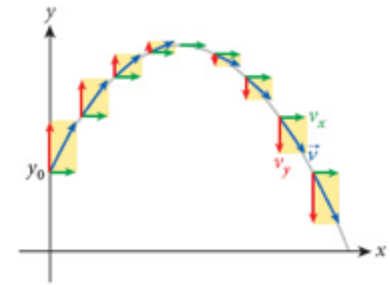


FIGURE 3.10 Graph of a parabolic trajectory with the velocity vector and its Cartesian components shown at constant time intervals.

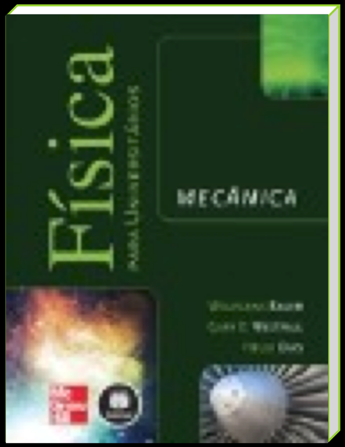
Velocity Components

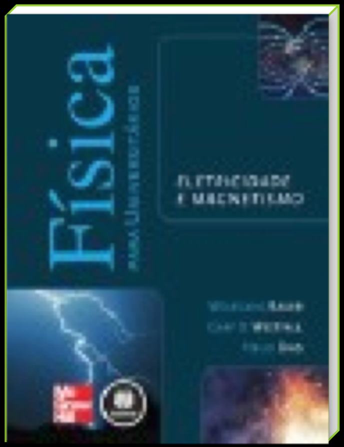
Although the vertical component of the velocity vector is equal to zero at the top of the trajectory, the gravitational acceleration has the same constant value as on any other part of the trajectory. Beware of the common misconception that the gravitational acceleration is equal to zero at the top of the trajectory. The gravitational acceleration has the same constant value everywhere along the trajectory.

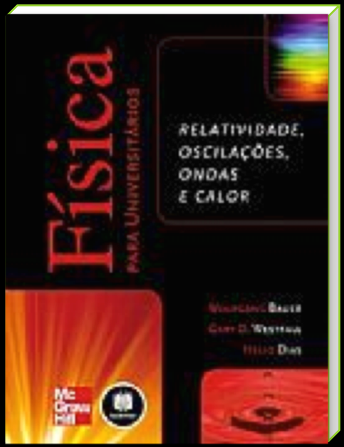
Concept-Check 3.3

Finally, let's explore the functional dependence of the absolute value of the velocity vector on time and/or the *y*-coordinate. We start with the dependence of $|\vec{v}|$ on *y*. We use the fact that the absolute value of a vector is given as the square root of the sum of the squares of the components. Then we use kinematical equation 3.12 for the *x*-component and kinematical equation 3.17 for the *y*-component. We obtain

PORTUGUESE VERSION





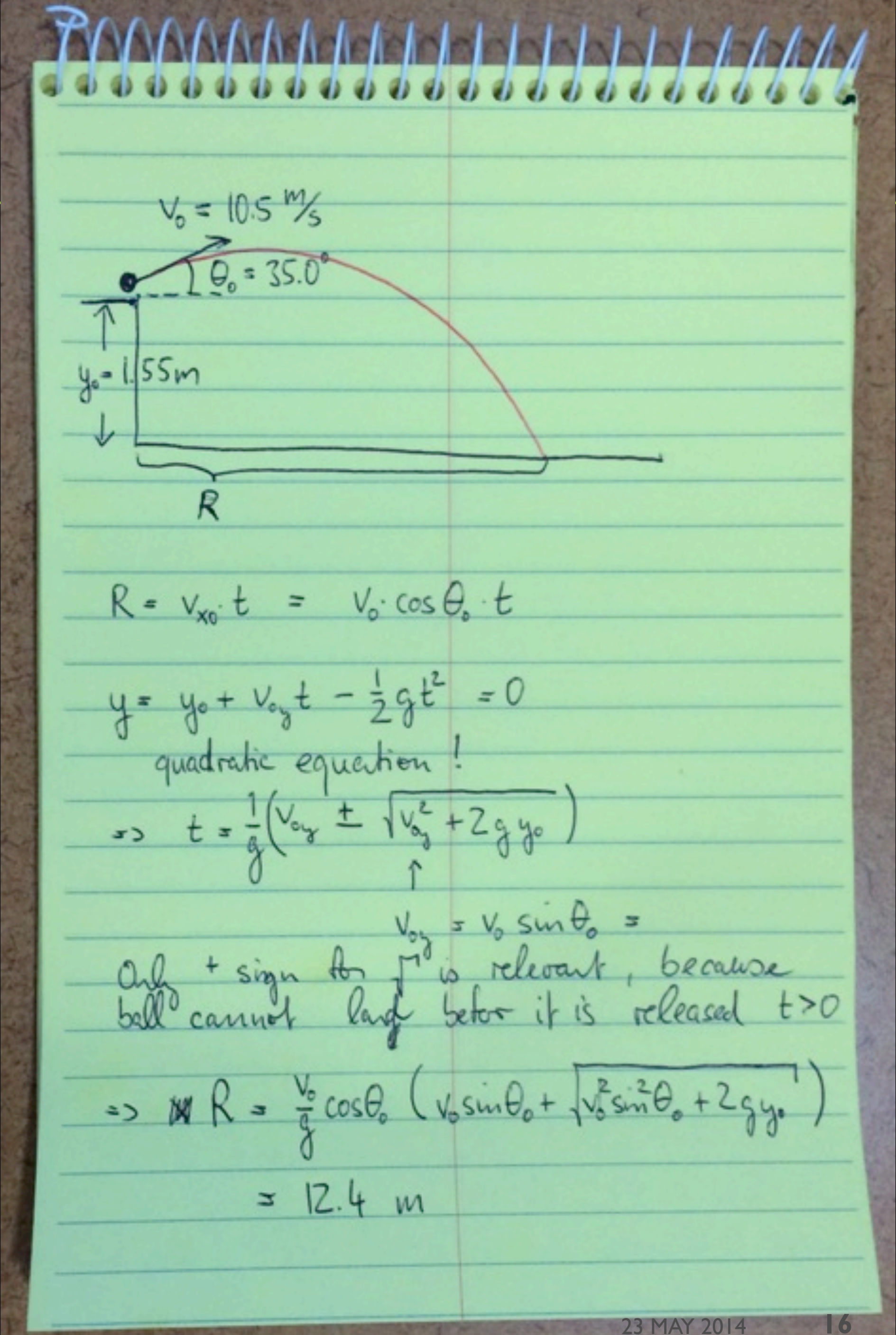


TRAJECTORIES

- Problem solving
  - Work with pencil and paper
  - Peer graded evaluations

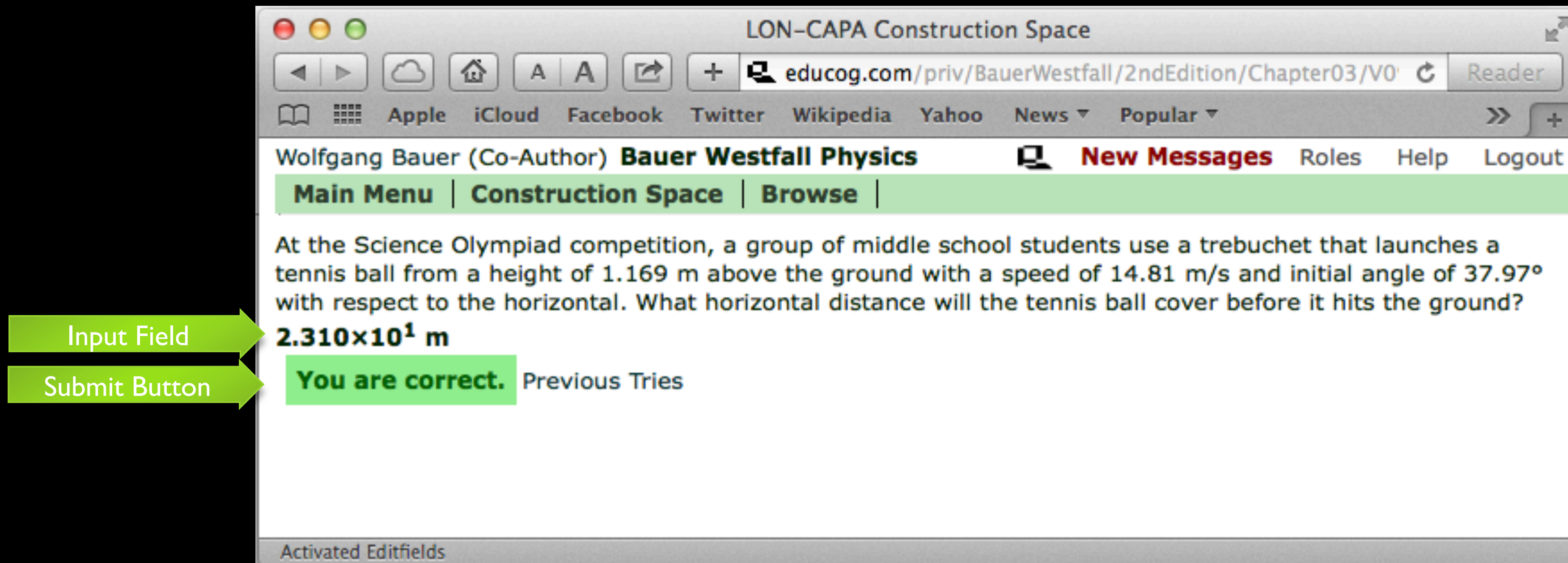
- Example

3.99. For a Science Olympiad competition, a group of middle school students build a trebuchet that can fire a tennis ball from a height of 1.55 m above the ground with a speed of 10.5 m/s and initial angle of 35.0° above the horizontal. What horizontal distance will the tennis ball cover before it hits the ground?



ELECTRONIC HOMEWORK

- Computer graded
- Randomized problems are different for every student (reduced copying and cheating)
- Immediate student feedback



COURSEWEAVER

http://www.courseweaver.com

AVAILABLE IN BRAZIL

- Course management system: LON-CAPA, now CourseWeaver
- Portuguese version of course management system available
- Partnership with Science Club

MULTI-VERSION EXERCISES

3.99 For a Science Olympiad competition, a group of middle school students build a trebuchet that can fire a tennis ball from a height of 1.55 m with a velocity of 10.5 m/s and a launch angle o
tal. What horizontal distance will the tennis bal
ground?

3.100 For a Science Olympiad competition, a g
students build a trebuchet that can fire a tennis
with a velocity of 10.5 m/s and a launch angle o
What is the *x*-component of the velocity of the
the ground?

3.101 For a Science Olympiad competition, a g
students build a trebuchet that can fire a tennis ball from a height of 1.55 m with a velocity of 10.5 m/s and a launch angle of 35.0° above the horizontal. What is the *y*-component of the velocity of the tennis ball just before it hits the ground?

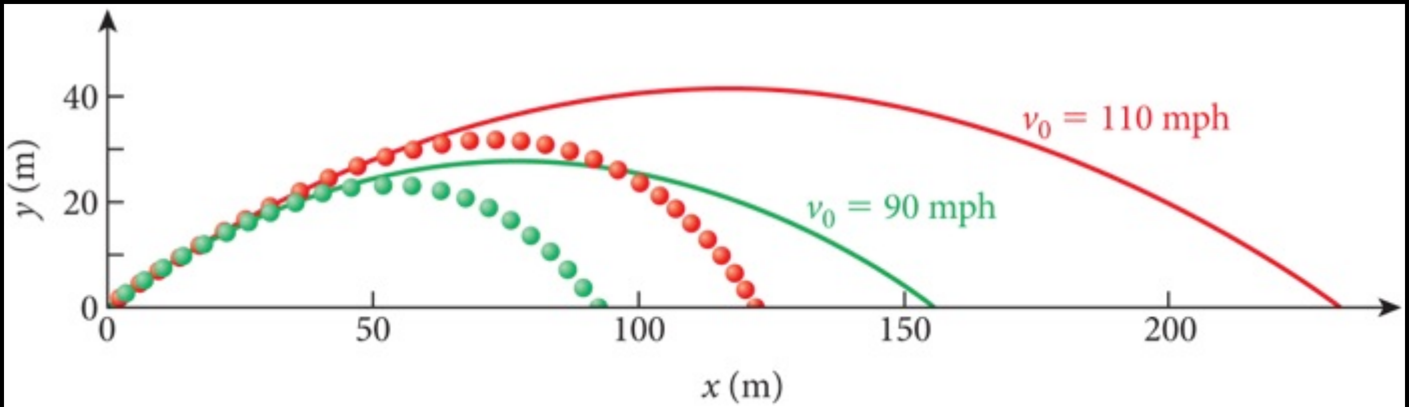
3.102 For a Science Olympiad competition, a group of middle school students build a trebuchet that can fire a tennis ball from a height of 1.55 m with a velocity of 10.5 m/s and a launch angle of 35.0° above the horizontal. What is the speed of the tennis ball just before it hits the ground?

Multiple Versions for

- Lecture
- Homework
- Quiz
- Midterm Exams
- Final Exam

TRAJECTORIES

- Real-world complications
  - Air resistance
  - Spin
  - …



- Solve numerically
  - Teach programming language (Python, html5, java, MatLab, Mathematica, FORTRAN, C, C++, …)
  - Programming tasks
  - Numerical analysis

TRAJECTORIES

- Computational task (4th Order Runge-Kutta)

```
*                                      ! First Step
      DO i = 1,n
        YT(i) = Y(i) + h/2*dYdt(i)
      END DO
*                                      ! Second Step
      CALL Derivs(n,t+h/2,YT,DYT)
      DO i = 1,n
        YT(i) = Y(i) + h/2*DYT(i)
      END DO
*                                      ! Third Step
      CALL Derivs(n,t+h/2,YT,DYM)
      DO i = 1,n
        YT(i) = Y(i) + h*DYM(i)
        DYM(i) = DYT(i) + DYM(i)
      END DO
*                                      ! Fourth Step
      CALL Derivs(n,t+h,YT,DYT)
*                                      ! Calculate Y(t+h)
      DO i = 1,n
        Y(i) = Y(i) + h/6*(dYdt(i)+DYT(i)+2.*DYM(i))
      END DO
```

VISUALIZATION → INSIGHT

Here: Terminal Speed in Free-Fall with Air Resistance

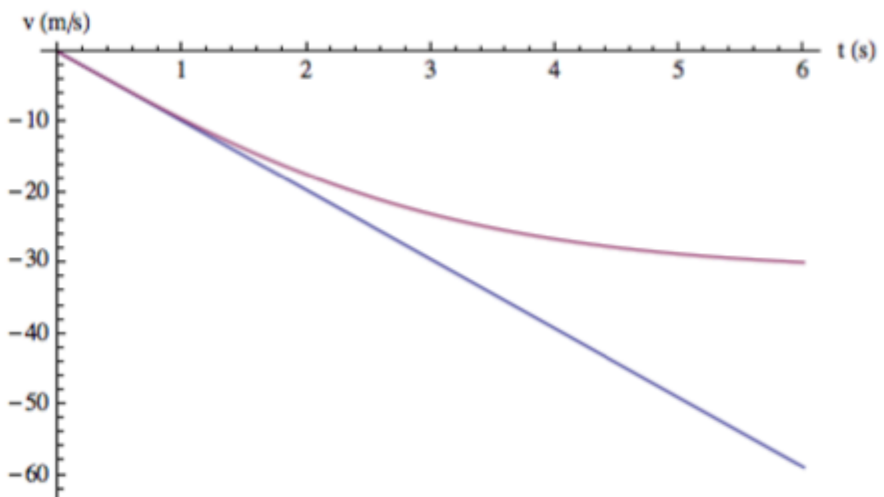
n[1]:=
```
ode1 = {y''[t] == -9.81, y[0.01] == 500 - 9.81*0.01*0.01, y[0] == 500}
```
ut[1]= $\{y''(t) = -9.81, y(0.01) = 499.999, y(0) = 500\}$

n[2]:=
```
ode2 = {y''[t] == -9.81 + 0.01*y'[t]^2, y[0.01] == 500 - 9.81*0.01*0.01, y[0] == 500}
```
ut[2]= $\{y''(t) = 0.01\, y'(t)^2 - 9.81, y(0.01) = 499.999, y(0) = 500\}$

n[3]:=
```
sol1 = NDSolve[ode1, y, {t, 0, 8}]
```
ut[3]= {{y → InterpolatingFunction[(0. 8.), <>]}}

n[4]:=
```
sol2 = NDSolve[ode2, y, {t, 0, 8}]
```
ut[4]= {{y → InterpolatingFunction[(0. 8.), <>]}}

n[5]:=
```
Plot[{Evaluate[y'[t] /. sol1], Evaluate[y'[t] /. sol2]}, {t, 0, 6},
 PlotRange → {-70, 0}, AxesLabel → {"t (s)", "v (m/s)"}]
```
ut[5]=



ADVANCED SKILLS

- Analyze real-world situations
  - Example: Analyze footage from sports videos
- Compare theoretical and computational findings to experimental situations
  - Integrated laboratory experiences
- Solve real-world engineering tasks
  - Design structures and devices
  - Design performance tests
  - Conduct performance evaluations

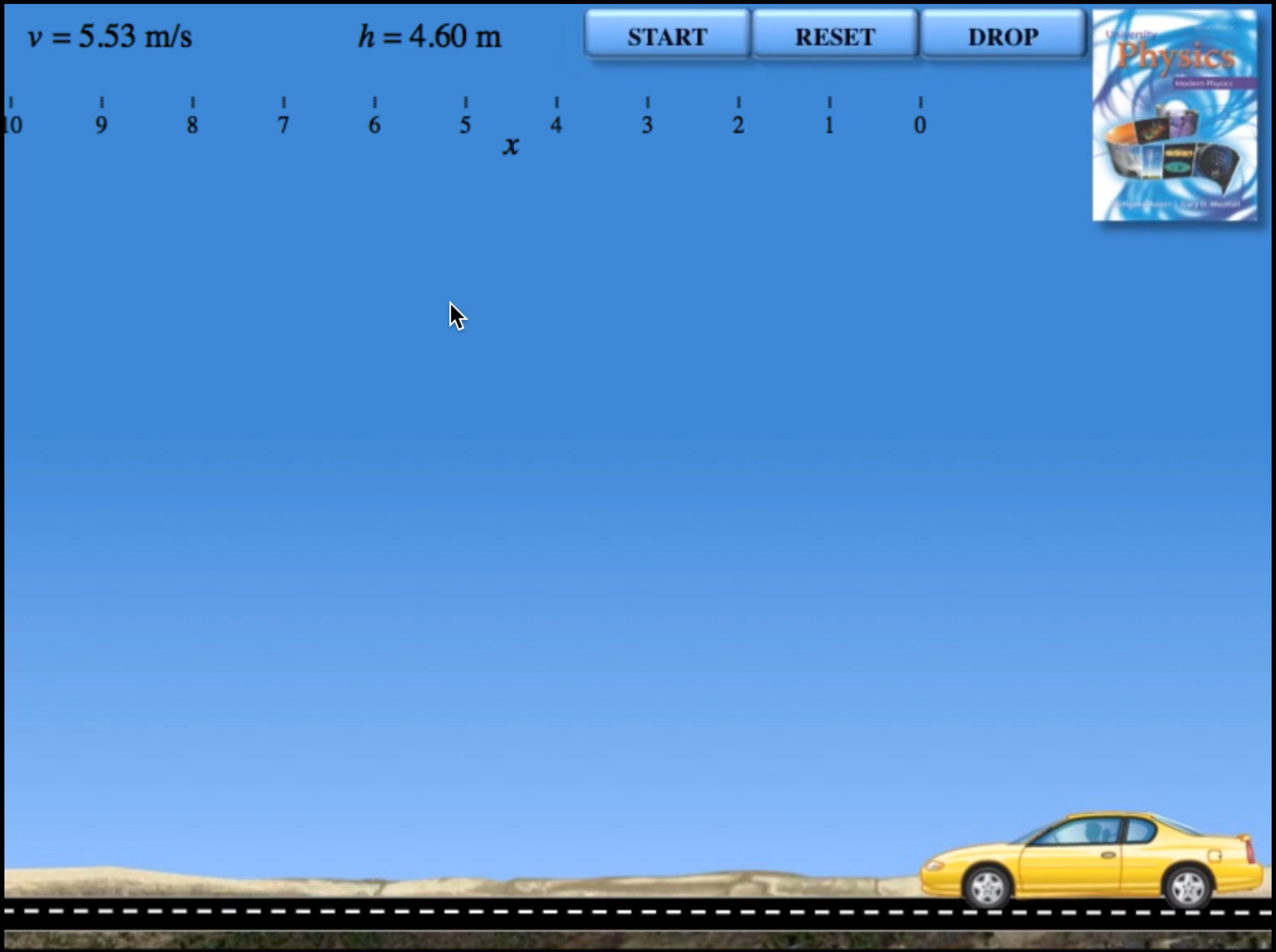
EXAMPLE: BASEBALL

EXAMPLE: BASEBALL

EXAMPLE: BASEBALL

Video Analysis: Impulse and Momentum Transfer

FUN/GAMING INTEGRATION

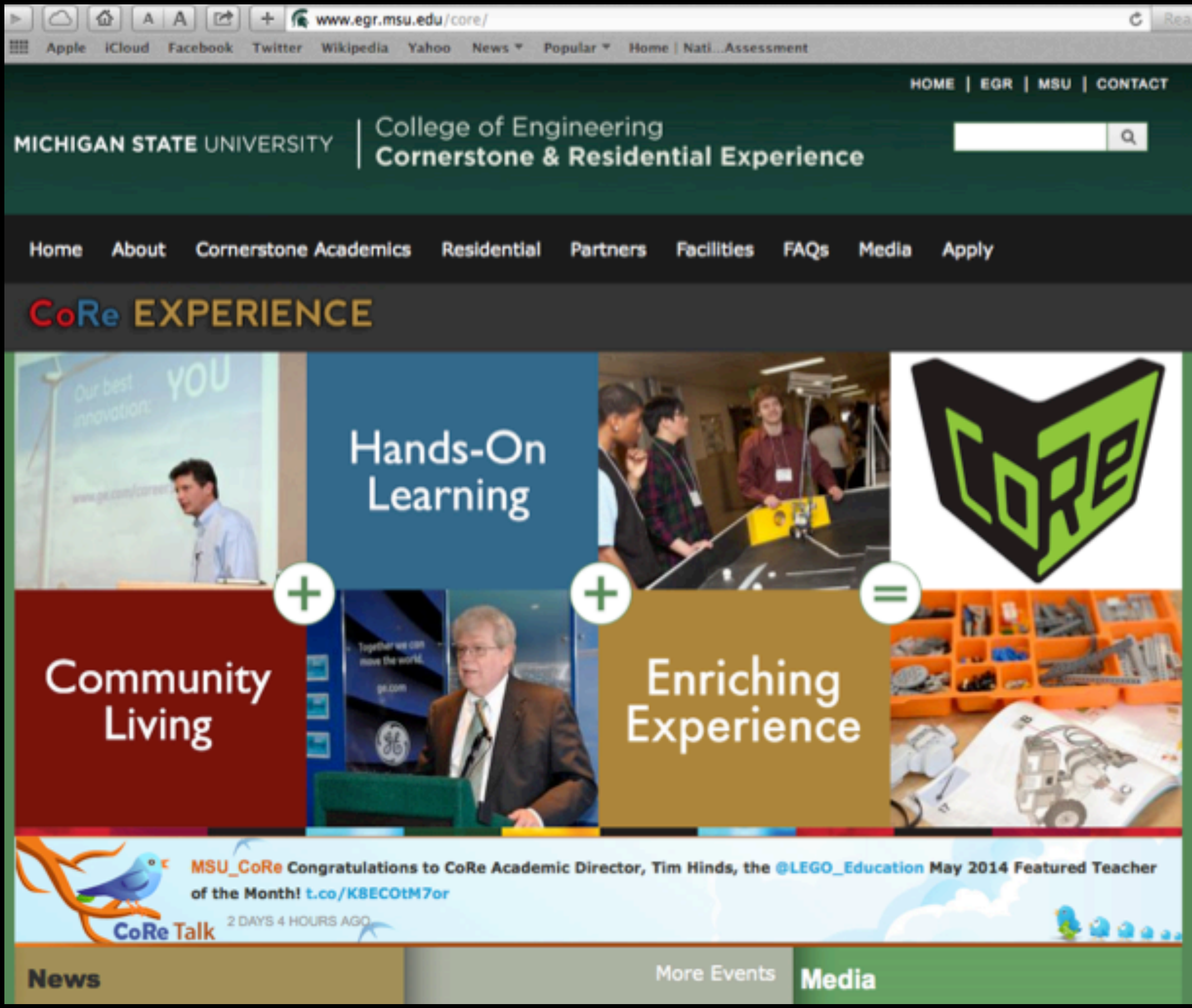


PHYSICAL SPACE

- SCALE-UP style collaborative classrooms

CORNERSTONE AND RESIDENTIAL EXPERIENCE

- Integration of first year academic experience with engineering living-learning community
- 'building the whole engineer'



SUMMARY: FYIE

- First Year Integrated Engineering
- Tightly integrated interdisciplinary curriculum
  - Mathematics
  - Physics
  - Computer Science
- Problem-Based Learning
- Flipped classroom
- New student experience
  - Work in collaborative teams
  - Cohort formation