Abstract
In 2016, students in General Physics at Bridgewater College, most of whom had no previous programming experience, were introduced to programming in Glowscript. In homework and in class, students simulated objects moving under the influence of forces.

A written survey probed student comfort level with the programming activities (N = 30), with mixed results. While most students felt adequately prepared, a significant minority of students expressed frustration with the coding assignments.

In response, two changes were implemented in the 2017 course offering: more code templates were provided to students, and in-class activities replaced the homework. This cohort (N = 22), only one student reported feeling confused. Overall, the majority of students found the computational aspect helpful.

Introduction
The AAPT recommends incorporating at least three types of computational tools into undergraduate physics programs. Bridgewater College aims to familiarize physics students with Excel, Mathematica, and at least one language suitable for scientific programming. For the past two years, our introductory physics class has incorporated programming in VPython. The students in this class complete a survey at the end of the first semester.

Research Goals
- Assess student comfort level and perceived value in programming assignments
- Identify obstacles or challenges in incorporating programming into introductory physics course

Student Perceptions of Cost/Benefit
Frequently-cited Benefits
- Visualizing the physics
- Gaining programming experience
- Practice applying the equations

Frequently-cited Challenges
- Learning curve required for novice programmers
- Too much time spent learning programming
- Frustration during debugging (especially when working on assignments at home)

Conclusions
Improving the efficiency of student programming activities by doing them in-class and providing templates was helpful in reducing student frustration. Perceived value remained highly variable, but overall students found the activities helpful.

Recommendations
- Use class time for programming to ensure students have access to the instructor while developing and debugging their code.
- Encourage concurrent or prior enrollment in a programming course
- Participate in PICUP (Partnership for Integration of Computation in Undergraduate Physics)²

References & Suggested Resources
1. AAPT Recommendations for Computational Physics in the Undergraduate Physics Curriculum, 2016 (aapt.org)
2. PICUP (gopicup.org)