CS+PA^2: Learning Computer Science with Physical Activities and Animation - a MathDance Experiment

ChanJin Chung and Mark Kocherovsky
Lawrence Technological University, cchung, mkocherov@LTU.edu

***Abstract* – A holistic approach to integrate classroom physical activities and programming to animate the physical activities is presented. The goal of this CS+PA^2 program is to introduce an effective learning model to teach computer science, coding, and STEM concepts. As an example of CS+PA^2, we taught an experimental "MathDance" program to seventh grade students using the Scratch Programming. The assessment results (one assessment taken before the activities, one after) show that the MathDance program improved students' knowledge in math and computer science significantly. It also increased students' confidence in STEM classes and interest in STEM careers. Students who repeated the MathDance motions did better, though not statistically significant, on a mathematics and computer science test than those students who did not.**

Index Terms – Computer science education, physical activity in classrooms, Scratch, coding, animation, STEM education, integrated curriculum, visual learning, kinesthetic learning.

INTRODUCTION

Computer science curricula are being introduced into classrooms at an increasing rate [1], since more states (in the United States of America) mandate high schools count computer science as math or science subjects [2]. In 2017, Virginia became the first state to require computer science instruction [3]. Worldwide, several countries include computer science in their compulsory curricula [4].

To provide an effective learning environment to introduce computer science into classrooms, a new initiative called CS+PA^2 (learning Computer Science with Physical Activities and Animation) was launched. The idea of CS+PA^2 is based on an integrated curriculum framework [13][14] where integrated lessons help students make connections and see the relevance between subjects. In CS+PA^2 students first learn various physical activities, and then illustrate and animate the activities by writing code, while learning concepts of computer science. Examples of planned physical activities include dance, cheerleading, Zumba, Yoga, Tai-chi, martial arts training, and fitness boot camps.

Why do we use physical activities for learning? First of all, numerous research results suggest that classroom-based physical activities can help learning and improve academic performance, because the body movement can be an effective cognitive strategy to strengthen learning, improve memory and retrieval, and enhance learner motivation and morale [5]-[8].

Why do we combine physical activities and animation? Walter Barbe and his colleagues proposed three learning modalities: auditory, visual, and kinesthetic [9][10]. Auditory learners best learn through listening to lectures, discussions, tapes, etc. [10]. Visual learners have a preference for learning though visual aids such as graphs, charts, diagrams, and symbols that represent ideas [10]. Kinesthetic/tactile learners prefer to learn via experience - moving, touching, and doing [10]. Kinesthetic learning involves hands-on experience with the goal of forming cognitive associations (the connections made between physical motion and the curriculum). It is reported that learning modality strengths can occur independently or in combination [9]. The CS+PA^2 approach focuses on integrating Kinesthetic and Visual learning to teach CS and STEM effectively. As a CS+PA^2 program, we first taught the MathDance introduced as "Beautiful Dance Moves" in [11] and then explained how to create a 2 dimensional animation of the dance using the Scratch programming language, which has been used world-wide to teach coding to create interactive stories, games, and animations in K-12 environments [12].

The next section in this paper describes our methods for the first experiment employing the "MathDance" and the animation of the dance using Scratch to demonstrate that the CS+PA^2 idea enhances learning. In the third section we analyze pre and post assessment data, then the summary and conclusion are presented in the last section.

METHODS IN MATH DANCE EXPERIMENTS

We worked with the University Prep Science & Math Middle School in Detroit to bring two classes, on different days, to Lawrence Technological University for the MathDance experiment. Eighteen 7th grade students came for the first experiment. We setup a room for them, and had them pair off into teams of two, with one laptop computer assigned to each team. Then, we gave them a pre-assessment test as shown in Appendix A. For math questions, the students were shown simple graphs ($y = x^2$, for example), and asked to identify the correct equations for each graph. For computer science questions, the students were given simple decision problems with a variable and nested loop questions. They were also asked two additional questions to gauge their interest in STEM fields shown in Appendix C. Once all the pre-assessments were complete, we taught them about graphs and functions. After that, we walked them through basic coding concepts such as variables, operators, loops, nested loops, conditional loops, and decisions using the Scratch Cat as a Sprite (a character on the screen that the code manipulates). To play dance music, the concept of multi-tasking was also introduced. These were designed to prepare the students for creating their own MathDance animations in the program.

After the initial tasks, we sent them to another "station" to have their pictures taken in 9 different poses. Each pose represented a mathematical function as shown in FIGURE II. For example, a pose in a "T" shape represented the function y = 0. After that, we taught them how to remove the background of the photos in Microsoft PowerPoint, then import the 9 photos into Scratch as Sprites.

After explaining a segment of pseudo code shown in FIGURE III for the complete MathDance, we showed & assisted them with the first steps in creating the MathDance code with a pre-recorded music loop, called "drum_jam", as shown in FIGURE IV. Then, we let them continue on their own to complete the MathDance. The remaining parts of the codes are shown in FIGURE V and VI. At the end of the event, we gave them a post-assessment as shown in Appendix B, with a similar difficulty level and similar questions to the first and the common questions in Appendix C. The whole agenda for the first control group (who did not perform the actual dance in class) is shown in FIGURE I in black.

- Check-in & Forming teams of two
- Pre-assessment
- Math equations review
  - **🡸 MathDance! (All together, for ~3 minutes)**
- Learning eight Scratch basic tasks with help
  - **🡸 MathDance! (All together, for ~3 minutes)**
- Taking pictures of you
- Removing background of the pictures
- Learning the first part of MathDance Animation with help
  - **🡸 MathDance! (All together, for ~3 minutes)**
- Completing the MathDance Animation without help
- Lunch and presentations
- Post-assessment

FIGURE I
MATHDANCE AGENDA; PURPLE PART WAS DONE ONLY BY THE 2ND GROUP

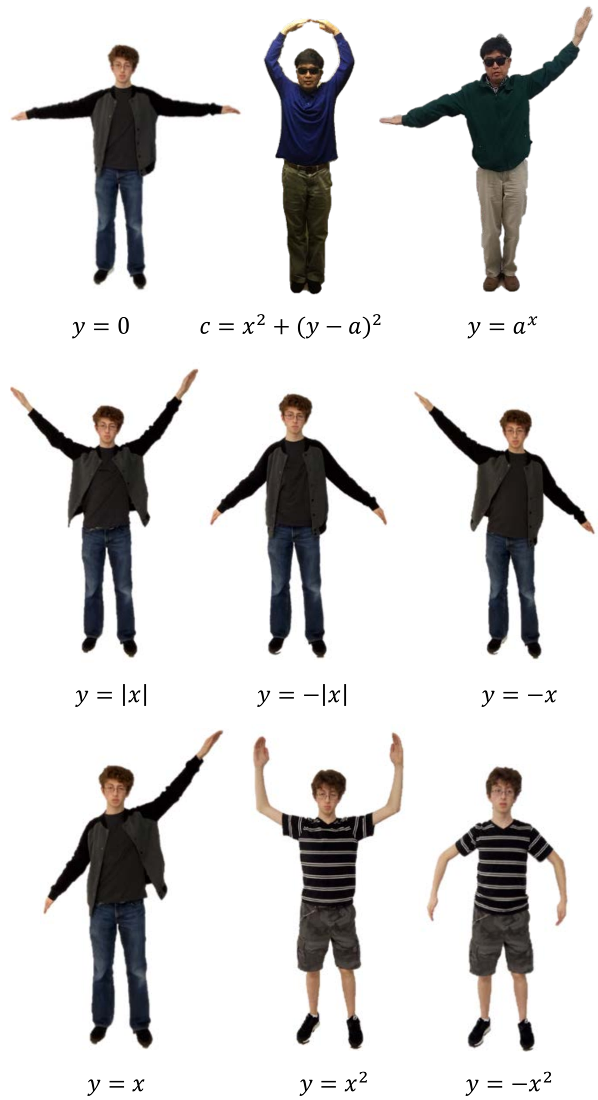


FIGURE II
NINE EQUATIONS AND POSES

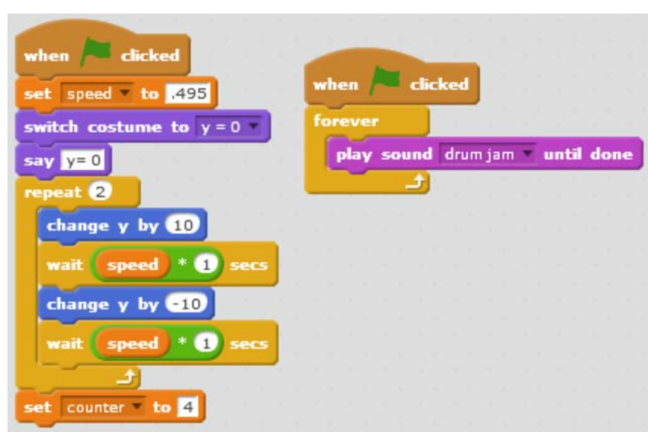


FIGURE IV
SCRATCH CODE FOR THE "A" PART IN THE PSEUDOCODE

```
Play music concurrently;                            ⎫
Set delay as 0.495; // t_unit                       ⎬ A
Up-down moves using y = 0 for 4 times;              ⎭

Set counter = 4;                                    ⎫
Repeat until counter < 2                            ⎪
    Repeat twice the following                      ⎪
        Show y = x;                                 ⎪
        Wait delay*counter seconds;                 ⎪
        Show y = -x;                                ⎪
        Wait delay*counter seconds;                 ⎪
        Show y = |x|;                               ⎪
        Wait delay*counter seconds;                 ⎬ B
        Show y = -|x|;                              ⎪
        Wait delay*counter seconds;                 ⎪
        Show y = x^2;                               ⎪
        Wait delay*counter seconds;                 ⎪
        Show y = -x^2;                              ⎪
        Wait delay*counter seconds;                 ⎪
    End of Loop;                                    ⎪
    Decrease counter by 2;                          ⎭
End of Loop;

Choose a number (either 1 or 2) randomly;           ⎫
If the number is 1                                  ⎪
    Show "circle";                                  ⎪
Else                                                ⎬ C
    Show y=a^x;                                     ⎪
Play cheering and clapping sound;                   ⎪
Stop all;                                           ⎭
```

FIGURE III
PSEUDOCODE FOR THE MATHDANCE

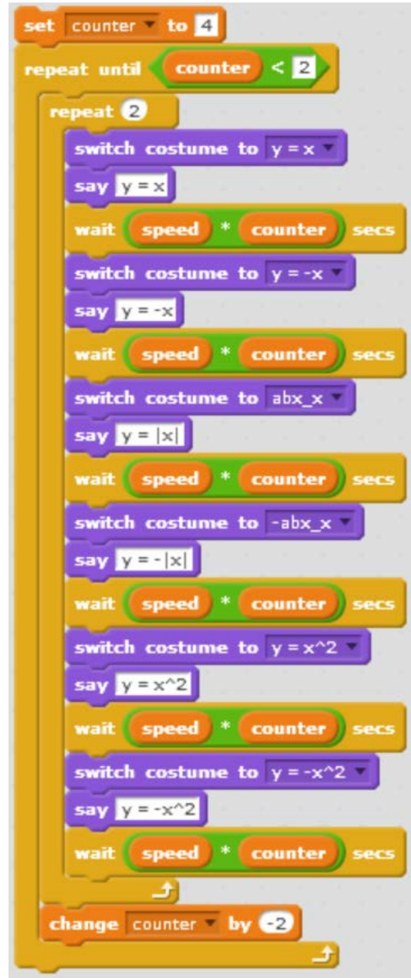


FIGURE V
SCRATCH CODE FOR THE "B" PART IN THE PSEUDOCODE

FIGURE VI
SCRATCH CODE FOR THE "C" PART IN THE PSEUDOCODE

For the second group, a different set of sixteen 7^{th} grade students came from the same school. They were given the same setup as the 1^{st} group. The 2^{nd} group followed the same agenda shown in FIGURE I. In addition, they performed the actual MathDance 3 times as shown in purple in FIGURE I. When they danced, we first projected our pre-made dance animation as well as choreography note on the screen as shown in FIGURE VII. FIGURE VIII shows a snapshot of the actual MathDance activity.

Our first goal of this research project was to show that $CS+PA^2$ approach improves math and CS learning. The 2^{nd} goal was to show that those students who learn and practice the MathDance would do better on a mathematics and computer programming test than those students who do not.

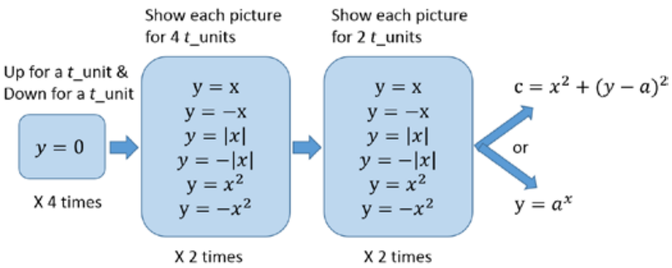


FIGURE VII
CHOREOGRAPHY NOTE GIVEN TO STUDENTS FOR THE MATHDANCE

FIGURE VIII
A SNAPSHOT OF ACTUAL MATHDANCE SHOWING Y=X POSE

ANALYSIS OF ASSESSMENT DATA

After grading the tests, we compared their pre and post-test scores, and after both groups had been graded, we compared both group's post-test scores. The only difference between the experiments between Groups 1 and 2 was that Group 2 had slightly less students (18 vs 16), and we stopped Group 2 during the experiment to repeat the dance 3 times as explained in the previous section.

Group 1, the group that did not perform the dance, improved their scores from the pre-test to the post-test by 104.8% (from an average score of 1.17 to an average score of 2.39 out of 5) as graphed in FIGURE IX. By a paired t-test, the p-value of the two tests was less than 0.0001, which, being under 0.05, is considered extremely significant. There were 18 students in this group and none got a score of zero on the post-test. Though not statistically significant, the Group 1 students were more interested in STEM classes and STEM fields as shown in FIGURE X and XI.

Group 2, the group that did repeat the dance, improved their scores from the pretest to the posttest by 136.4% (from an average score of 1.38 to an average score of 3.25) as graphed in FIGURE IX. By a paired t-test, the p-value of the two tests was 0.0003, which is considered extremely significant. There were 16 students in this group and none scored zeros on the post test. Though not statistically significant, the Group 2 students also increased interest in STEM classes and STEM fields as shown in FIGURE X and XI.

An unpaired t-test (because of the difference in sample size) to check whether the actual MathDance in class helped learning or not shows a p-value of .0935. While this is not statistically significant, it is close. It is likely that this is due to the small sample sizes. It would have been ideal to have at least 25 students in each group.

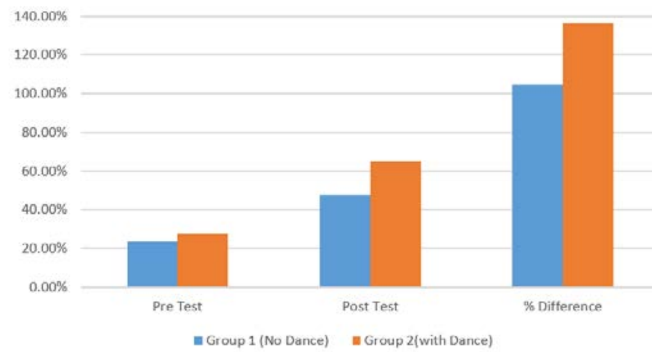


FIGURE IX
COMPARISONS OF GROUP 1 AND GROUP 2 MATH & CS TEST DATA

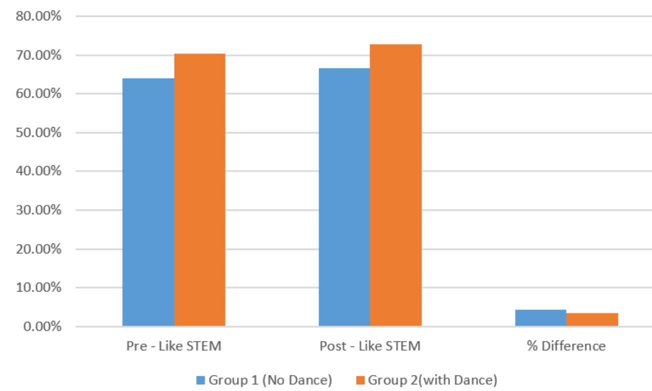


FIGURE X
COMPARISONS OF GROUP 1 AND GROUP 2 STEM LIKENESS DATA

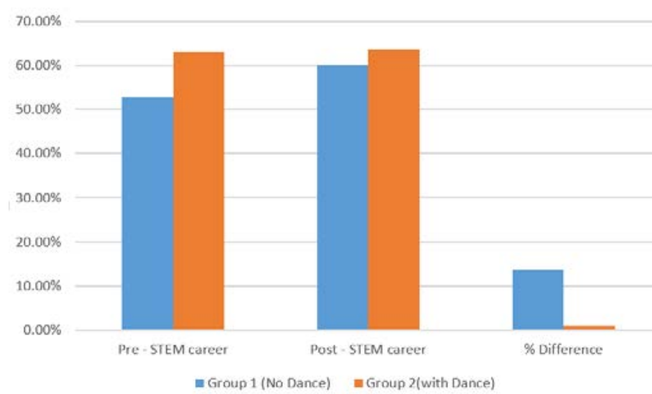


FIGURE XI
COMPARISONS OF GROUP 1 AND GROUP 2 STEM CAREER DATA

SUMMARY AND CONCLUSIONS

MathDance is an innovative example program of $CS+PA^2$ in which K-12 students learn mathematical dance in classrooms, and then learn to animate photographs of themselves dancing using the Scratch programming language to learn computer science. It is an integrated, interdisciplinary, and synergistic approach to combine math, computer science, technology, and physical education and can be an effective model for a STEM initiative emphasizing on computer science projects. Coding played an important role -like glue - to integrate multiple subjects. The MathDance program was not just for learning coding, but coding was used to learn computer science concepts, math, technology, animation, and physical activity. As we expected, the assessment results show that students' knowledge in math and computer science significantly increased in a group with actual dance (Group 2) as well as a group without actual dance (Group 1). We believe this is due to the animation component that uses the visual learning mode. Though not statistically significant, both Group 1 and Group 2 students increased interest in STEM classes and STEM fields.

Regarding the effect of kinesthetic learning style, students who learned and practiced the MathDance physically (Group 2) did better on a mathematics and computer programming test than those students who did not. Future work will include continuous experiments of this MathDance program with larger sample sizes.

Future experiments would include more collection and analysis of qualitative data as well instrumented data within Scratch.

ACKNOWLEDGMENT

This research is supported by HHMI (Howard Hughes Medical Institute) CRE (Classroom-based Research Experience) grant (No. 52008705). We thank teachers from University Prep Science & Math Middle School in Detroit: Mrs. Brienn Frederick (MA, SCL College and Career Counselor) and Mr. Neal Maclellan (Teacher). We also thank Dr. Lior Shamir, PI of the HHMI CRE grant and Dr. Sibrina Collins, Director of Marburger STEM center at LTU who gave opening and closing remarks. Mirit Shamir, Shannan Palonis, and Elmer Santos assisted the programs.

REFERENCES

[1] Kang, C. 2017. "Tech firms add $300 Million to Trump Administration's Computer Science push." The New York Times. September 26.

[2] Loewus, L. 2016. "More States Mandate High Schools Count Computer Science as Math or Science." Education Week. September 13.

[3] Sawchuk,S. 2017. "Virginia Becomes First State to Require Computer Science Instruction" Education Week. November 21.

[4] Passey, D. 2017. "Computer science (CS) in the compulsory education curriculum: Implications for future research." *Education and Information Technologies*. March 2017, Volume 22, Issue 2, pp 421–443.

[5] Donnelly JE, Lambourne K. 2011. Classroom-based physical activity, cognition, and academic achievement. *Preventive Medicine*. 52 (Suppl 1):S36-S42.

[6] Erwin, H., Fedewa, A., Beighle, A. & Ahn, S. 2012. "A Quantitative Review of Physical Activity, Health, and Learning Outcomes Associated With Classroom-Based Physical Activity Interventions." Journal of Applied School Psychology, Vol. 28, Issue 1.

[7] Singh, A., Uijtdewilligen L., Twisk, J., et al. 2012. "Physical Activity and Performance at School A Systematic Review of the Literature Including a Methodological Quality Assessment." *Arch Pediatr Adolesc Med*. 2012; 166 (1):49–55.

[8] Jensen, E. 2005. Teaching with the Brain in Mind, 2nd Edition, Chapter 4.

[9] Barbe, W., Swassing, R., and Milone, M. 1979. "Teaching through modality strengths: concepts practices." Columbus, Ohio: Zaner-Bloser.

[10] Fleming, N.D; (1995), "I'm different; not dumb. Modes of presentation (VARK) in the tertiary classroom.", in Zelmer,A., (ed.) *Research and Development in Higher Education, Proceedings of the 1995 Annual Conference of the Higher Education and Research Development Society of Australasia (HERDSA)*, Volume 18, pp. 308-313.

[11] Gerofsky, S. 2013. "Learning Mathematics Through Dance." *Proceedings of Bridges 2013: Mathematics, Music, Art, Architecture, Culture*. pp 337-344.

[12] Resnick, M., Maloney, J., and Hernández, A., et al. November 2009. "Scratch: Programming for All". *Communications of the ACM*. 52 (11): 60–67.

[13] RJ Fogarty, 2009. "How to integrate the curricula" *Corwin Press.*

[14] Gehrke, N. 1991. "Explorations of Teachers' Development of Integrative Curriculums." *Journal of Curriculum Supervision*. 6/2 (1991): 107-112.

AUTHOR INFORMATION

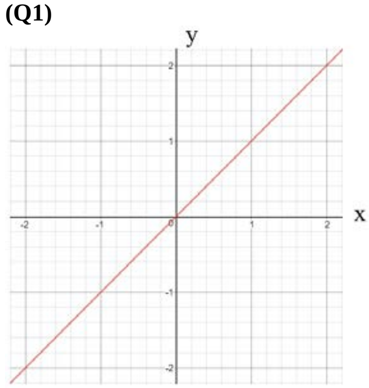
ChanJin "CJ" Chung, Professor; Founder & Director of Robofest, Department of Mathematics and Computer Science, Lawrence Technological University.

Mark Kocherovsky, Computer Science student & Robotics Lab Assistant, Lawrence Technological University.

APPENDIX A: PRE ASSESSMENT QUESTIONS

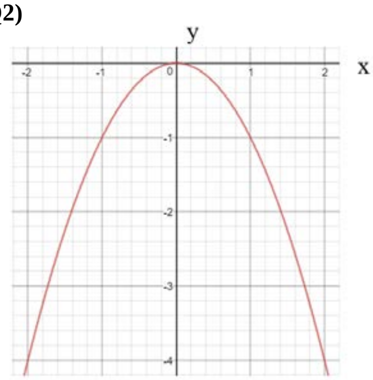
Choose the equation of each function graphed on the left.

(Q1)



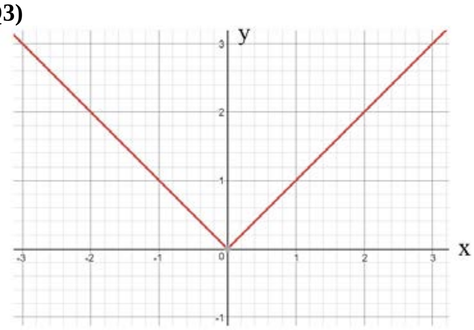
a. $y = x^2$
b. $y = -x$
c. $y = x$
d. $y = -x^2$
e. I don't know

(Q2)



a. $y = 0$
b. $y = -x^2$
c. $y = |x|$
d. $y = x^2$
e. I don't know

(Q3)



a. $y = x$
b. $y = |x|$
c. $y = x^2$
d. $y = -|x|$
e. I don't know

(Q4)

How many steps will be moved according to this program?

```
Repeat the following Begin-end block 2 times:
     Begin:
          Repeat the following Begin-end block 3 times:
               Begin:
                    Move 5 steps.
               End
     End
```

(Q5)

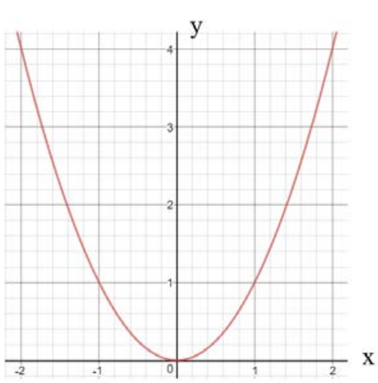
The current temperature t is 28 degrees in Fahrenheit. How many steps will be moved according to this program?

```
If t > 32 then:
     Move 4 steps.
Otherwise:
     Move 10 steps.
```

APPENDIX B: POST ASSESSMENT QUESTIONS

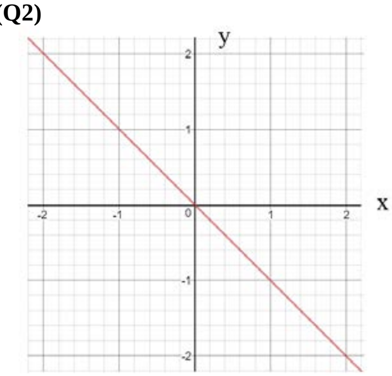
Choose the equation of each function graphed on the left.

(Q1)



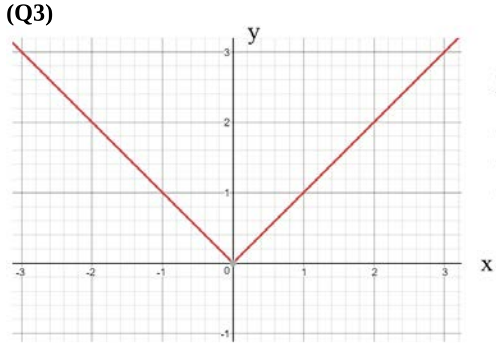
a. $y = -x$
b. $y = x^2$
c. $y = |x|$
d. $y = 0$
e. I don't know

(Q2)



a. $y = -|x|$
b. $y = x$
c. $y = x^2$
d. $y = -x$
e. I don't know

(Q3)



a. $y = x$
b. $y = |x|$
c. $y = -|x|$
d. $y = -x^2$
e. I don't know

(Q4)

How many steps will be moved according to this program?

```
Repeat the following Begin-end block 3 times:
     Begin:
          Repeat the following Begin-end block 2 times:
               Begin:
                    Move 4 steps.
               End
     End
```

(Q5)

The current temperature t is 75 degrees in Fahrenheit. How many steps will be moved according to this program?

```
If t > 32 then:
     Move 4 steps.
Otherwise:
     Move 10 steps.
```

APPENDIX C: COMMON ASSESSMENT QUESTIONS

Do you like Science, Computer Science, Technology, Engineering, and/or mathematics (STEM) related classes? Please circle a number that best represents how you feel on a scale of 0 to 9.

Do **not** like STEM classes 0 1 2 3 4 5 6 7 8 9 Love STEM classes

Are you interested in a future career involving Science, Computer Science, Technology, Engineering, and/or mathematics (STEM)? Please circle a number that best represents how you feel on a scale of 0 to 9.

Not interested at all 0 1 2 3 4 5 6 7 8 9 Very much interested