Journal of Jewish Early Childhood Education

Vol. 1 No. 1 | 2026

Buber on Coding

Toward a Conceptual Framework for a Relational Pedagogy of Technology in Jewish Early Childhood Education

Marina Umaschi Bers

Boston College

This article is part of the inaugural issue of the Journal of Jewish Early Childhood Education.

Buber on Coding: Toward a Conceptual Framework for a Relational Pedagogy of Technology in Jewish Early Childhood Education

Marina Umaschi Bers

Kindergarten children are sitting on a rug in a big circle. It is the beginning of the morning routine. In the middle of the rug, there is a small robot. Children are excited and wiggle. The teacher calls out: "Who wants to tell the robot to move?" "Me, me, me, me…." dozens of voices respond while raising their hands as high as possible. "Go ahead, Ilana", says the teacher. "Robot, move!" yields the little girl. But there is no response from the robot. "Can someone else try"? "Please, please, please, robot move!" says Robby. Nothing happens. "Robot, walk now" commands Itai. No movement happens. "Do you think this robot understands English?" asks the teacher. Students laugh and Shiri, in a loud voice, says "Robot, yala!", but the robot doesn't move. "Robot, lech!" orders Nathan. Nothing. One by one, children give it a try with different words in both English and Hebrew. All of them are unsuccessful at commanding the robot to move. Softly, the teacher interrupts the different attempts and explains that the robot doesn't understand any human language. It only understands computer language, code. She then explains to a group of open-eyed five-year-olds that in this class, they will learn how to program robots.

This teacher has set up a coding playground, in which learning about and with technology happens through playful collaborative exploration (Bers, 2012; 2020). Children are not only learning about coding, or robotics, but they are also exploring what makes us human: the use of languages, both human and artificial. In a coding playground, children learn new skills while having fun. They develop curiosity and persistence. They engage in problem solving. However, most importantly, they encounter each other.

The instrumental relationship with the technology, in this case learning how to program a robot to move, is a stepping stone towards the emotional and social relationships with other humans. Although the robot in this vignette might be described in Martin Buber's terms as an *It*, children are using it in the process of forming *I–Thou* encounters. This essay advances a theoretical argument grounded in the philosophy of Buber (1923, 1970) that suggests that the teaching of coding, robotics, and computational thinking can become a vehicle for cultivating a relational pedagogy. Furthermore, the paper explores two theoretical questions: How might Buber's distinction between *I–Thou* and *I–It* relationships illuminate the pedagogical possibilities

of coding playgrounds? and What Jewish values can inform the implementation of coding playgrounds to meet the needs of Jewish education?

Dialogic Encounters

Martin Buber's philosophy of dialogue distinguishes between two fundamental modes of relationships: *I–It*, in which the other is treated as an object to use or experience, and *I–Thou*, in which both partners engage in a mutual, present, and authentic encounter. Both of these forms of encounters characterize the human existence. The *I–Thou* relationship refers to the mutual, holistic existence of two beings who engage in a dialogue fully present, while the *I–It* relationship is nearly the opposite. The beings do not actually "meet". Therefore, the *I–It* relationship is not a dialogue but a monologue, it captures a transactional, instrumental encounter.

Drawing from Buber's insights and previous work focused on the teaching of computer science as a humanistic endeavor (Bers, 2025), this paper situates the teaching of coding not merely as a technical skill but as a context for enabling dialogic encounters. Most specifically, it introduces the notion of a coding playground as a pedagogical space where the instrumental use of technology (*I–It*) can evolve toward meaningful relationships with others (*I–Thou*). Through this lens, computer programming becomes another language through which children can explore identity, community, and ethical responsibility.

Education, for Buber, is a relational act that invites presence and mutuality (Buber, 1970). This paper illustrates how Buber's relational pedagogy can be implemented in Jewish early childhood settings by integrating the teaching of coding, computational thinking, and robotics to address the needs of growing up Jewish in the diaspora: building a sense of belonging and relationships, exploring Jewish values, rituals, holidays and traditions, connecting with prayer in meaningful ways and reinforcing the Hebrew language.

Technology Education in Early Childhood

The inclusion of technology in schools situates children in two different theoretical camps: children as consumers of technology and children as producers of technology. Within the second group, the effort is on teaching children the tools to become creators of new apps, programmers of robots and trainers of artificial intelligence bots. In short, teaching them how to think in computational terms and how to learn the logic behind computer programming languages.

The last decades brought an increased global demand for computer science education starting in preschool and kindergarten (Bers, 2019; Modan, 2019; Noh & Lee, 2020). Early childhood settings all over the world are now teaching computational thinking, computer programming and robotics (Sano 2019; Seow et al. 2019; Toikkanen & Leinonen 2017). At the writing of this paper, over thirty states in the U.S. are teaching computer science (Code.org, 2024) and in 2023 the European Union called on all member countries to make computer science a required subject. Furthermore, the 2022 OECD working paper (Bers et al, 2022) advocates for the inclusion of computational thinking in early childhood education, suggesting that when introduced in a playful and developmentally appropriate way, it can complement foundational skills like literacy and numeracy.

Research shows that educational interventions that begin in early childhood are associated with lower costs and durable impacts (Cunha & Heckman, 2007; Heckman & Masterov, 2007). In addition, early experiences have an impact in achieving later academic success (National Research Council, 2011; Shonkoff, J, & Phillips, D.A. 2000), in combating gender-based stereotypes in STEM fields (Madill et al., 2007; Markert, 1996) and in increasing interest in engineering, and computer science (Sullivan & Bers, 2013, 2016, 2018; Metz, 2007).

The current spread of computer science education responds to the economic demands of a growing workforce needing technical expertise. However, learning to code goes beyond preparation for the future: it fosters a set of higher order thinking skills (Kafai & Burke, 2014; Tissenbaum et al 2021; Kwon et al 2021) that scholars refer to as computational thinking (Wing, 2006; 2011; Guzdial, 2008; Yadav, 2011). In the process of learning how to use a programming language, one learns to think in logical ways using sequencing, modularization, and abstraction while developing an analytical ability.

In the early years, abstraction can be challenging, thus the importance of using concrete tools and objects to explore abstract concepts, both through computational media and low-tech or unplugged activities (Papert, 1980; Bers, 2008). For example, computational thinking can be promoted in everyday activities such as: sorting LEGOs (using the concept of “hashing” to sort by color, shape, and size), cooking a meal (using “parallel processing” to manage cooking different types of food at different temperatures for different amounts of time), and looking up a name in an alphabetical list (linear: starting at the beginning of the list; binary: starting at the middle of the list).

As children engage in these types of activities, they learn how to break one big problem into smaller steps, how to plan and test different strategies (Bers, 2021). Problem solving is hard and requires focus and attention, as well as learning how to manage frustration when things do not work as expected, how to stay on task and be open-minded to trying different potential solutions, and how to work in a persistent way. Bers describes how the coding playground can intentionally become a learning environment to promote some of these character traits (Bers, 2022) and how the activity of programming can support the development of a palette of virtues (Bers, 2021a).

The book *Beyond Coding: How Children Learn Human Values through Computer Programming* (Bers, 2022), uses the metaphor of a palette of virtues with ten universal character traits that have been observed in different coding playgrounds across cultures and religions: Curiosity, perseverance, patience, open-mindedness, optimism, honesty, fairness, generosity, gratitude, and forgiveness. These are not the only values, just like colors in a painter's palette, new ones can be added. In a coding playground in which teachers become intentional about their palette of virtues and how to put those virtues into practice in their own classrooms, children have the opportunity to use programming languages to encounter the other.

For example, some teachers might focus on turn-taking, taking care of materials, and collaborative work, and others might pay attention to teaching how to be patient when trying to problem-solve. Some might use mindfulness for helping children work through the frustration of debugging, and others might engage children in writing "thank you" notes to acknowledge those who help each other. An intentional palette of virtues can guide teachers in making pedagogical choices to allocate time, distribute materials and set up classroom practices with a sense of purpose. It can help structure the development of technical skills with personal growth.

Coding Playgrounds as Relational Spaces for Encounters

When viewed through a conceptual lens, the teaching of coding is not only about logic and computation but also about communication and intention. A coding playground is a place where children learn by playing, by engaging with others, by taking risks and by developing both social and emotional skills, alongside cognitive ones. In contrast with a coding playpen which is limited and engages children as consumers of technology, in the coding playground children have the opportunity to learn a programming language to create computational projects to express themselves (Bers, 2020). In the process, they are likely to encounter powerful ideas from

different disciplines, and to think about their own thinking (Papert, 1980). However, something else can happen.

In a coding playground children learn the power of languages to create or destroy worlds. The pioneering Reggio Emilia school system in Italy promotes the use of "hundred languages of children" (Edwards et al, 1993). Coding playgrounds offer one more language: computer programming languages. Like visual arts, dramatic play, music, and physical activity that can be used by children to express themselves (Bredekamp, 1991), programming languages present children with the unique affordances of a dynamic, interactive media for self-expression (Ben-Ari et al, 2025)

However, the power of languages goes beyond self-expression. It engages people in relational practices. In a coding playground, children can encounter each other through the use of technology. The Positive Technological Development (PTD) framework (Bers, 2012) informs the design of coding playgrounds that can serve as relational spaces for encounters and identifies six developmental behaviors that can be observed: content creation, creativity, communication, collaboration, community building, and choices of conduct. Within this conceptual framework, the teaching of coding, computational thinking and robotics, may create opportunities for children to reflect on their relationships, choices, and responsibilities.

This does not imply that coding causes moral or character development. Rather, this paper suggests that, given the increasing demands of a technological age that requires the teaching of computational skills, we have an opportunity to set up intentional coding playgrounds to facilitate what Buber described as the movement from *I–It* to *I–Thou* relationships—from instrumental manipulation toward relational encounters. Coding playgrounds invite to see technology as an instrument for connection, responsibility, and shared creativity. These goals resonate with Jewish pedagogical aims, emphasizing not only what children know but who they can become.

Jewish Values in the Coding Playground

Jewish early childhood programs aim to instill a positive Jewish identity by fostering a sense of belonging through stories, symbols, rituals, and language (Pomson & Wertheimer, 2017), thus new technologies can serve as another language of expression to strengthen this sense of belonging. Furthermore, they are an opportunity to learn about Jewish content, traditions and practices in innovative and playful ways (Kress, 2003).

Jewish education has long integrated learning, ethical practice, and creativity. Furthermore, it has embraced a balance between tradition and innovation (Feldman, 1992; Reisman, 1979; Bloomberg, 2007). By introducing coding playgrounds as relational spaces that promote *I–Thou* relationships through the *It* of technologies, educators can frame the teaching of new technologies not as distractions, but as extensions of timeless Jewish ideals and as a pathway for identity formation, community building and sacred creativity. Coding can become another language through which children can express themselves Jewishly. And the palette of virtues informing the pedagogical decision-making, can be intentionally created so children can thrive not only as coders or digital creators, but as *mensches*—kind, thoughtful, and responsible individuals who use their gifts to make a better world.

In this paper, I suggest four core Jewish values that can be added to a palette of virtues that is intentionally designed for Jewish educational settings.

- **B'tzelem Elohim**: Understanding the activity of coding as co-creation. Just as humans are created in the divine image, with the capacity to create, children express their uniqueness through technological making. Furthermore, as each human being is unique, so must be the projects created by children in the coding playground. For example, children can express what being Jewish means to them, through their technological creations. They might program a robotic play to reenact the story of Purim or design a digital art project that reflects their understanding of Shabbat. Just as God is described as the ultimate Creator in Genesis, so too are humans endowed with the power to shape, imagine, and innovate. By engaging young children in making their own creative coding projects, each one of them brings a unique voice and vision to their creations, reflecting the uniqueness of every human being. The idea of being in God's image carries with it ethical responsibility and invites moral reflection on how to use creative power in the coding playground.
- **Tikkun Olam**: Computer science seeks to use technology to solve human problems, and in the coding playground this vision of tikkun olam—repairing the world—becomes a lived practice. Children learn that coding is not only about making something work but about making something better—for themselves, their peers, and their communities. When they design a robot to help with classroom organization, program a story that promotes kindness, or create a game that teaches empathy, they engage in *tikkun* through

creative technological action. In the age of Artificial Intelligence, this value takes on new ethical urgency, as technology increasingly shapes human choices and relationships. The coding playground thus becomes an early moral laboratory where learners explore fairness, bias, and inclusion, discovering that with creative power comes responsibility. By framing coding as a form of tikkun, educators help children see themselves as ethical co-creators of a shared future, where debugging, redesigning, and reimagining technology become acts of repairing the world.

- **Klal Yisrael**: In the coding playground, learning is inherently social: children exchange ideas, offer feedback, and problem-solve together, discovering that creativity and innovation thrive through cooperation. This mirrors the communal nature of Jewish learning, where dialogue and interpretation are central acts of connection. Whether children are coding animated stories about Jewish holidays or designing games that teach Hebrew—the language that binds generations of Jewish learning (Sarna, 2004)—they strengthen both their peer relationships and their connection to Jewish culture and tradition. The coding playground thus becomes a microcosm of Klal Yisrael, a living community where individual voices contribute to a shared creation.
- **Teshuvah**: Often translated as "return" or "repentance", teshuvah is not merely about correcting mistakes but about engaging in a deep process of reflection, learning, and transformation. In the context of the coding playground, teshuvah finds a natural home in the iterative process of designing, debugging, and revising. When children write code, they inevitably encounter errors, yet rather than viewing these as failures, the process invites them to pause, reflect, and try again. Debugging thus becomes an exercise in humility and persistence, an acknowledgment that imperfection is part of creation and that growth emerges through engagement and repair. Each revision of a program represents a small act of renewal—a return to one's intentions and an opportunity to align action with purpose.

By embedding these four values, and others, the coding playground can transform from a site of technical learning into a site of relational growth grounded on Jewish values.

Coding Playgrounds in Jewish settings

Different pedagogical approaches can be used to introduce computer science education in the continuum starting with "direct instruction by the teacher" and ending with "construction of

knowledge by the learners" (Kirschner et al, 2006). The approach described in this paper is based on Papert's constructionist methodology which positions that children learn best when provided with opportunities and technological tools to become active designers, makers and coders of their own personally meaningful projects (Papert, 1980). Grounded on Piaget's work, Papert paid special attention to the role of computers and computation as powerful tools for learning. Inspired by this work, Bers developed the notion of coding playgrounds (Bers, 2018; 2020).

Next, this paper introduces a variety of projects implemented in different coding playgrounds in Jewish early childhood settings. The projects chosen to describe here, are intended as illustrations of what is possible. Furthermore, links to resources for implementing these projects are included and references to research implementation of these programs are provided. Children learn the new language of computation with the goal to not only foster computational thinking, robotics knowledge and coding skills, but also to strengthen Jewish identity, explore Jewish values, learn about Jewish holidays, experience meaningful Jewish prayers and connect to the Hebrew language. But most importantly, throughout all of these experiences, children are invited to use the *It*, the technology, to create *I–Thou* relationships.

Mi Ani? Who am I?

Scholarship argues that the primary task of contemporary religious education "is not so much to transmit faith but rather to facilitate the formation of personal identity as a core aspect of contemporary socialization processes" (Vermeer, 2009, p. 201). Moreover, the focus on Jewish identity has expanded from examining only "a person's active involvement in religious and cultural-communal practices and activities" to looking at an individual's "self-perception and self-definition as a Jew" (Horowitz, 2003, p.iv). It is in this context, that the *Mi Ani? Who Am I?* project was created integrating robotics with an exploration of Jewish identity.

The project was implemented in a pluralistic American Jewish day school kindergarten in the Boston area in a kindergarten classroom. Guided by the principle, "Livnot U'Lehibanot" (to build and to be built), the goal of this project was to engage children in a several months robotics curriculum as they explored their connection to Judaism. As they learned how to program and build their robots, they also build themselves as Jewish individuals. It is beyond the scope of this paper to report results of this experience but research was conducted to evaluate learning outcomes (Bers et al, 2013).

An important challenge for the Jewish minority and diaspora is how to maintain a sense of identity from generation to generation. One way to address this challenge is by engaging children in the exploration of their identity grounded in the minority culture, yet integrated with the majority mainstream culture.This project addressed this issue by inviting young children to create robotic creatures as alter-egos and to program their behaviors to travel across the secular and Jewish calendar. Children had to negotiate between choosing meaningful events in both calendars, expressing their own perception of themselves as kindergarten students in a Jewish day school in America.

For example, a child programmed her robot to travel across a large timeline in the classroom and to turn on its lights when reaching the month of December to symbolize the lighting of the Hanukah candles. A boy programmed his robot to shake and then spun to represent the overwhelming experience of tasting a bitter lemon during their science exploration of citrus fruits during October. Another child programmed a robot to move in a circular pattern during the Jewish holiday of Purim, reflecting the tradition of sharing joy. And yet another created a program to turn the robot's lights on and off in May for her birthday to represent her eyes "lighting up" in excitement. And a child program its robot to move back and forth to symbolize the rolling of the challah dough that his mother did every shabbat. All of these robots performed different actions when travelling in a timeline and encountering both Jewish and secular events that were especially meaningful for the children.

Research found that by programming their robots to perform actions, children not only developed computational thinking and robotic knowledge, but they explored their personal relationship with Judaism. Children's robotic projects expressed their understanding of themselves as actively engaging in Jewish practices such as lighting Hanukkah candles, rolling out matzah dough for Passover, or singing Hebrew songs. Robots allowed children to express themselves not only as "being" Jewish, but also as "doing" Jewish things. Furthermore, as robots traveled through the academic calendar, children encountered opportunities to express different dimensions of their identity: as members of a classroom engaged in a particular curricular activity (i.e., citrus fruits explorations), as members of the Jewish community celebrating holidays (i.e.,

Hanukah), and as individuals (i.e., birthday celebrations). The *Mi Ani? Who am I?* curricular unit described here, can be found free at: https://sites.bc.edu/devtech/beit-midrash/mi-ani/

Mi Ani? Who am I? was implemented several years in a row in different kindergarten classrooms. It success led to the development of a series of units integrating coding, robotics and Jewish studies in early childhood, which are described next.

Limudei Code-Esh

Funded by the David Lear Sulman Computing, Science, and Engineering Fund for Jewish Day Schools, the *Limudei Code-Esh* curriculum integrates coding and robotics with Judaic Studies. Over a period of several years and in collaboration with Jewish educators representing three different religious denominations in the Greater Boston area, the DevTech Research Group developed six free curricula units focused on Jewish holidays: Sukkot, Chanukkah, Tu B'shevat, Purim, Pesach, and Yom Ha'atzmaut. These units are designed for children in kindergarten through third grade using either the KIBO robotics platform described in the introductory vignette in this paper, or the free ScratchJr programming language. However, they can be adapted to any other coding or robotics environment or platform.

ScratchJr is an introductory programming language that enables young children (ages 5-7) to create their own interactive stories and games (Bers & Resnick, 2016). On tablets, children snap together graphical programming blocks to make characters move, jump, dance, and sing. Children can modify characters in the paint editor, add their own voices and sounds, even insert photos of themselves -- then use the programming blocks to make their characters come to life.

Each *Limudei Code-Esh* curriculum unit is designed for a total of 20 hours, but can be extended or shortened and implemented in any formal or informal learning setting around the world. The units of study are designed to complement instruction and to inspire a love of Jewish holiday celebration as well as knowledge and skills about computer science. Each of the units culminates with a final project in which children create animations or robotic plays to showcase their learning about the specific holidays, while engaging in problem solving and collaboration alongside developing technical skills and computational thinking.

For example, as a final project for the Tu-Bshevat curriculum, students make a large two-dimensional tree of life with tablets with ScratchJr animated projects laid down around a tree constructed with arts and crafts. While some children are invited to create interactive roots for the tree, others can work on leaves, and fruit. Students get a chance to use their programming skills in ScratchJr to make the tree come to life, while exploring the holiday of Tu-Bshevat. Combining customizable characters and appear and disappear blocks, students can make their leaves change in color to represent the seasons, or with grow and shrink blocks, could code the fruit to grow in size to express the tree's vitality. For the Chanukkah curriculum, children program their chanukiahs to light up as the week progresses and their dreidels to turn playfully. In the Pesach curriculum, students retell the story of Pesach through a multi-tablet ScratchJr program. Using the theatre mode in ScratchJr, multiple tablets are lined up next to each other, each telling one of the aspects of the holiday. The final project in the Sukkot curriculum invites students to program their KIBO robots to carry rings of gratitude into their own KIBO-sized Sukkah. KIBO's art stages allow students to customize the look of the KIBO. In the Purim curriculum, students finish the unit with a Purim parade led by KIBOs dressed as characters from the Purim story. Using different motion blocks, the KIBOs can be programmed to march in an orderly line or can spin and dance through their parade. And lastly, in the final project in the Yom Ha'atzmaut curriculum, students are tasked to program their KIBO robots to celebrate the independence of Israel with dance and music.

Each of the *Limudei Code-Esh* curricular units integrates technical skills with concepts related to specific Jewish holidays. See table 1 for a description of the Jewish concepts taught in each unit. The chosen concepts resulted from intentional deliberations between Jewish Studies teachers belonging to schools serving three different Jewish denominations, as well as a pluralistic school. During the planning of the curriculum units, several meetings occurred with the goal of having teachers identify key content areas that were taught across the spectrum of Jewish values, traditions and practices in the context of today's children's experiences. At the same time, in collaboration with the DevTech research group, a matrix indicating technical knowledge needed at three different levels (beginners, intermediate and advanced) was developed and projects were brainstormed. All of these curricular units can be freely accessed here: https://sites.bc.edu/devtech/beit-midrash/limudei-code-esh/. In addition, a recorded webinar

provides pedagogical guidelines for making sure *Limude Code-esh* units are taught in the context of coding playground.

Table 1

Holiday	Computational Level	Technology Used	Jewish Holidays Concepts
Tu B'shevat	Beginners	ScratchJr	Tu B'shevat as another new year, the rituals of Tu B'shevat, the seven species, trees representing humans, trees representing the Torah, preserving the environment for future generations, recycling and composting, cycles of the year, planting and appreciating trees, Tu B'shevat as a time for personal growth, and tikkun olam
Chanukkah	Intermediate	ScratchJr	preserve Jewish identity, battle against the Greeks, Maccabees, miracles, festival of lights, and hanukkiyah.
Pesach	Advanced	ScratchJr	chametz/matzah, freedom vs. slavery, The number four in the Haggadah, the four children and us, and retelling the story.
Sukkot	Beginner	KIBO robotics	impermanence/permanence, nature, sukkah, rejoice, the four species of plant, mitzvot, and gratitude/forgiveness
Purim	Intermediate	KIBO robotics	perseverance, power of community, hidden miracles, minhagim of Purim, and mitzvot.
Yom Ha'atzmaut	Advanced	KIBO robotics	Israel, State, Independence, Celebration, Symbols, and Key Figures

The goal of the *Limudei Code-Esh* project was to develop curriculum units and to explore how *I–Thou* relationships could be formed, not only amongst students experiencing these units, but also between teachers of different Jewish denominations working together towards a shared goal: to develop a shared language about key concepts to teach with technology for each of the Jewish holidays. So far, no research was conducted to investigate the impact of the units on student learning, and only anecdotes about their implementation were gathered. A different approach was taken in the *Con-science* project described below.

Con-science: Creative Robotic Prayers

This project involved families at a Jewish Day School in Argentina and it happened during the Yamim Noraim, the period of ten days between the Jewish New Year, Rosh Hashanah, and the Day of Atonement, Yom Kippur. Parents and children worked together on making creative prayers using LEGO-based robotic materials and research was conducted (Bers & Urrea, 2000). The school's Rabbi led the project alongside the research team. While the rabbi focused on leading activities about Jewish values that are core during the these holidays, such as Teshuvah, and on helping them discuss how those abstract values could be put into practice, the research team provided technical training by creating an intergenerational coding playground. Arts and crafts alongside Jewish texts, gears, motors, sensors, and LEGOs of all kinds and shapes, could be found in the classroom. It was decided to implement a robotics activity engaging parents and not only students, given the importance of educational programs which not only strengthen Jewish identity but also build a supportive community among families. By participating in these shared experiences, the goal was for families to deepen their connection to Jewish traditions and to one another.

For example, a father and his son decided to use LEGO bricks to build a large Maguen David (Star of David). They constructed a geometric structure with two interlocking triangles and a motor. Then, they programmed it with the LEGO Mindstorms robotics system to rotate in different directions, while playing the sound of the shofar and turning on and off lights on top of the structure. When interviewed, they explained that they wanted to represent the cyclical continuity of the Jewish people through the rotation of the Maguen David. Guided by the notion of "Le Dor va Dor" (from generation to generation) they explored how an ancient symbol could be used to express a dynamic concept through programming.

Another family, a child with his mother, chose to focus on the value of teshuvah, a Hebrew word meaning "return" or "repentance". It refers to the spiritual process of turning back to God, correcting one's mistakes, and striving to become a better person. Teshuvah is especially emphasized during the High Holidays—Rosh Hashanah and Yom Kippur—but is relevant year-round. It reflects Judaism's belief in human potential for growth and moral responsibility. The concept of Teshuvah is abstract. However, this family chose a very concrete artifact to symbolize it. They built and programmed a robotic LEGO structure to serve as a conveyor belt. Using cardboard, they created little boxes and invited people to hide inside, pieces of papers in which they wrote thoughts and actions of the past year, which they thought needed to be re-examined. As the conveyor belt was moved by a motor, it transported the little boxes with the past actions inside. As the conveyor belt reached a point, people could choose to have the boxes turn left or right on the conveyor belt. If they chose left, the little boxes were dropped into a "good" bucket and if they chose right, into a "bad bucket". All of the actions dropped into this bucket required teshuvah. This project was both conceptually and technically sophisticated. It displayed creativity and technical skill, as well as an understanding of the abstract concept of teshuvah. In addition, it promoted dialogue about the abstract categories of "good and bad" and how to engage in the process of reflection and fixing what needs to be fixed.

At the culmination of the workshop, each family was invited to write a creative prayer about their learning experience. Those creative prayers were then collected and copied and were included as an addendum in the prayer book utilized during that weeks' prayer in the synagogue next to the school. Both the robotic projects and the creative prayers reflecting on the learning experience, served to help families go beyond traditional, fixed liturgical formulas and involved personal, imaginative ways of connecting with the Divine, a sense of purpose, transcendence and belonging. Narrative alongside robotics was purposefully used to enrich the coding playground. Language plays an important role when working with coding. While the *Con-science* project and all of the curriculum units developed at DevTech include a narrative component, the Coding as Another Language (CAL) curriculum was explicitly designed to put together powerful ideas of computer programing and computational thinking in conversation with powerful ideas of literacy. In Jewish early childhood education, the CAL curriculum can be used in either English or Hebrew, or any other of the multiple languages it has been translated to.

Coding as Another Language

The Coding as Another Language (CAL) curriculum recognizes computer programming not only as a tool to solve problems, but as a literacy through which learners can tell stories and express themselves. Understanding coding as another language for expression facilitates this process. That is, introducing programming as an activity that utilizes a symbolic system of representation with communicative and expressive functions (Bers, 2019).

Although there are significant differences between programming languages and natural languages, CAL's focus is on shared practices: the creation of expressive compositional projects, either through coding or through writing, to share with others. It consists of twenty-four lessons of 30-45 minutes (18 total hours of instruction) that progressively take pre-kindergartners to second graders in an exploration of coding, computational thinking and literacy in an integrated way (Bers et al, 2023). CAL is structured around six powerful ideas from both computer science (algorithms, design process, representation, debugging, control structures, modularity, and hardware/software) and literacy (sequencing, the writing process, alphabet and letter-sound correspondence, editing and audience awareness, literary devices, phonological awareness, and tools of communication and language) respectively. Lessons invite children to explore and discover these powerful ideas in unplugged ways. Each grade level invites children to design and program two open-ended ScratchJr projects based on a fiction story with a strong sequence of events. Children re-create the stories making ScratchJr animations and sometimes invent new endings. In the context of Jewish early childhood settings, the books have been chosen to reflect Jewish themes. The culmination of each CAL unit involves a community presentation in which students share their interactive narratives and celebrate their hard work.

Each CAL lesson follows a similar structure: warm up games to playfully introduce or reinforce computational and literacy concepts, design activities to solidify skills, structured challenges to practice, free explorations for students to tinker and expand skills, expressive explorations to promote creativity, off-screen unplugged games involving social interactions and movement, writing activities and technology circles to share and reflect. Individual, small group and whole classroom activities are interspersed in the curriculum involving a variety of playful, developmentally appropriate activities.

Each lesson on the CAL curriculum is designed to promote one or more of the six behaviors (content creation, collaboration, creativity, communication, choices of conduct and community building) identified by the Positive Technological Development Framework (PTD)

for designing coding playgrounds. In addition, CAL positions the teaching of creative programming as a pathway for character development using the metaphor of a palette of virtues. When we learn a new language, we also must learn how to use it in responsible ways. Each lesson identifies a specific human value, virtue or character strength to be developed and/or practiced in the proposed activities. For example, first grade's "Lesson 15: Need for Speed" teaches the ScratchJr "speed block". Students create a program where multiple characters compete in a race and are given different starting points and speeds. Not only do children code their own races, but they also explore the concept of fairness and what makes a fair race.

In Jewish early childhood education, teachers do not only serve as facilitators of meaningful play and inquiry to promote cognitive, emotional, and social growth. Most importantly, they model Jewish values, integrating these principles into daily routines and interactions. In the same way that artists use their color palettes to create different paintings, CAL supports teachers to identify each virtue they want to promote in their lessons. The CAL curriculum has been translated to Hebrew by a team of Israeli colleagues and has been implemented and evaluated in Israeli kindergartens (Bers et al, 2023a). Although significant research has been done with CAL throughout the world, it is beyond the scope of this paper to describe results. Those can be found in many publications (Zhanxia et al, 2025; Levinson et al, 2025). CAL is freely accessible to be used by anyone around the world and it can be found here: https://sites.bc.edu/codingasanotherlanguage/curricula/scratchjr-curricula-hebrew/

Conclusion

This paper has proposed a conceptual framework for understanding the teaching of coding and technology in Jewish early childhood education through the lens of Martin Buber's philosophy of dialogue. At its core, this framework positions technology not as an instrument of efficiency but as a medium for human encounters—a bridge from the *I–It* to the *I–Thou*. Furthermore, this framework proposes a pedagogy that exposes that learning involves not only using human languages, but also creating new artificial languages, such as programming languages.

In the coding playground, children learn both to command machines but also to communicate, collaborate, and create meaning with one another. Through playful exploration, programming becomes a language of relationship, reflection, and ethical engagement. Grounded in the Positive Technological Development framework and the metaphor of a palette of virtues,

the coding playground intentionally integrates moral and character formation with technical learning. It positions the acquisition of computational skills alongside the cultivation of virtues such as curiosity, perseverance, generosity, and gratitude.

In Jewish educational contexts, these virtues resonate deeply with enduring values that guide both learning and life: *B'tzelem Elohim*—recognizing human creativity as an act of co-creation with the Divine; *Tikkun Olam*—understanding technology as a tool for repair and ethical responsibility; *Klal Yisrael*—fostering community, belonging, and shared purpose; and *Teshuvah*—embracing reflection, growth, and renewal.

Through projects such as *Mi Ani? Who Am I?*, *Limudei Code-Esh*, *Con-science*, and *Coding as Another Language*, we witness how Jewish education can merge ancient wisdom with contemporary tools. Children not only learn to code but to express their Jewish identity, celebrate their traditions, and connect to their communities through technological creation. These experiences exemplify how Buber's vision of relational education—rooted in presence, dialogue, and responsibility—can flourish even in the digital age.

Ultimately, teaching coding in Jewish early childhood settings is not simply about preparing children for a technological future; it is about guiding them to become thoughtful, compassionate, and creative individuals, *mensches*, who use technology in the service of connection, meaning, and the sacred task of making the world a better place. In this way, the *coding playground* becomes both a site of learning and a sanctuary of encounter, where the human and the technological, the creative and the spiritual, meet in dialogue.

Martin Buber proposed that human life involves a continuous oscillation between *I–Thou* and *I–It* encounters. He argued that the growing dominance of a purely analytical and materialistic perspective posed the danger of fostering *I–It* relationships. As technology quickly enters all domains of our lives, especially education, Buber's warning becomes even more urgent. As we train teachers to convert their classrooms into coding playgrounds, we also invite them to intentionally think about the palette of virtues they silently bring into their classrooms to nurture *I–Thou* relationships to flourish. It is the Jewish palette of virtues the one that should guide the pedagogical choices in the coding playground.

References

Ben-Ari, A., Levinson, T. G., Blake-West, J., Bers, M. U., & Rosenberg-Kima, R. B. (2025). From avoidance to ownership: Preschool teachers learn to teach code, a case study. *Educational Technology Research and Development*. https://doi.org/10.1007/s11423-025-10548-0

Bers, M. U. (2012). *Designing digital experiences for positive youth development: From playpen to playground*. Oxford University Press.

Bers, M. U. (2019). Coding as another language: A pedagogical approach for teaching computer science in early childhood. *Journal of Computers in Education, 6*(4), 499–528. https://doi.org/10.1007/s40692-019-00147-3

Bers, M. U. (2020). *Coding as a playground: Programming and computational thinking in the early childhood classroom* (2nd ed.). Routledge.

Bers, M. U. (2021a). Coding, robotics and socio-emotional learning: Developing a palette of virtues. *Pixel-Bit: Revista de Medios y Educación, 62*, 309–322.

Bers, M. U. (2021). *Teaching computational thinking and coding to young children*. IGI Global. https://doi.org/10.4018/978-1-7998-7308-2

Bers, M. U. (2022). *Beyond coding: How children learn human values through programming*. MIT Press.

Bers, M. U. (2025). Computer science education as a humanistic endeavor: A model for designing technology-rich formative experiences. *Education and Information Technologies*. https://doi.org/10.1007/s10639-025-13623-4

Bers, M. U., Blake-West, J., Kapoor, M. G., Levinson, T., Relkin, E., Unahalekhaka, A., & Yang, Z. (2023). Coding as another language: Research-based curriculum for early childhood computer science. *Early Childhood Research Quarterly, 64*, 394–404. https://doi.org/10.1016/j.ecresq.2023.05.002

Bers, M. U., Doyle-Lynch, A., & Chau, C. (2012). Positive technological development: The multifaceted nature of youth technology use toward improving self and society. In C. C. Ching & B. J. Foley (Eds.), *Constructing the self in a digital world* (pp. 110–136). Cambridge University Press.

Bers, M. U., Levinson, T., Yang, Z., Rosenberg-Kima, R. B., Ben-Ari, A., Jacob, S., Dubash, P., Warschauer, M., Gimenez, C., Gonzalez, P., & Gonzalez, H. (2023a). Coding as another language: An international comparative study of learning computer science and computational thinking in kindergarten. In P. Blikstein, J. van Aalst, & K. Brennan (Eds.), *Proceedings of the 17th International Conference of the Learning Sciences* (pp. 1659–1665). https://2023.isls.org/proceedings/

Bers, M. U., & Resnick, M. (2016). *The official ScratchJr book: Help your kids learn to code*. No Starch Press.

Bers, M. U., Strawhacker, A., & Sullivan, A. (2022). The state of the field of computational thinking in early childhood education. *OECD Education Working Papers, No. 274*. https://doi.org/10.1787/3354387a-en

Bers, M. U., & Urrea, C. (2000). Technological prayers: Parents and children exploring robotics and values. In A. Druin & J. Hendler (Eds.), *Robots for kids: Exploring new technologies for learning experiences* (pp. 194–217). Morgan Kaufmann.

Bers, M., Matas, J., & Libman, N. (2013). Livnot u'lehibanot: To build and to be built: Making robots in kindergarten to explore Jewish identity. *Diaspora, Indigenous, and Minority Education, 7*(3), 164–179.

Bloomberg, L. D. (2007). Teaching Jewish values in early childhood: A developmental approach. Coalition for the Advancement of Jewish Education.

Bredekamp, S. (1991). Redeveloping early childhood education: A response to Kessler. *Early Childhood Research Quarterly, 6*, 199–209.

Buber, M. (1970). *I and thou* (W. Kaufmann, Trans.). Charles Scribner's Sons. (Original work published 1923)

Code.org. (2024). *State of computer science education*. https://advocacy.code.org/stateofcs/

Cunha, F., & Heckman, J. J. (2007). The technology of skill formation. *American Economic Review, 97*(2), 31–47. https://doi.org/10.1257/aer.97.2.31

Edwards, C., Gandini, L., & Forman, G. (Eds.). (1993). *The hundred languages of children: The Reggio Emilia approach to early childhood education*. Ablex.

Feldman, J. (1992). *The world of the Jewish child*. Behrman House.

Guzdial, M. (2008). Education: Paving the way for computational thinking. *Communications of the ACM, 51*(8), 25–27. https://doi.org/10.1145/1378704.1378719

Heckman, J. J., & Masterov, D. V. (2007). The productivity argument for investing in young children. *Applied Economic Perspectives and Policy, 29*(3), 446–493. https://doi.org/10.2139/ssrn.987876

Horowitz, B. (2003). *Connections and journeys: Assessing critical opportunities for enhancing Jewish identity*. UJA-Federation of New York.

Kafai, Y. B., & Burke, Q. (2014). Constructionist gaming: Understanding the benefits of making games for learning. *Educational Psychologist, 49*(1), 1–13. https://doi.org/10.1080/00461520.2013.878318

Kress, J. S. (2003). Developing Jewish children's spirituality through play. *Religious Education, 98*(1), 5–23.

Kwon, M., Lee, H., & Lee, J. (2021). Computational thinking in early childhood education: A systematic review. *Computers & Education, 161*, 104084. https://doi.org/10.1016/j.compedu.2020.104084

Lee, I., Martin, F., & Apone, K. (2011). Computational thinking for youth in practice. *ACM Inroads, 2*(1), 32–37. https://doi.org/10.1145/1929887.1929903

Levinson, T., Carocca, P. F., & Bers, M. U. (2025). Coding as another language: An early childhood programming curriculum in Argentina. *Computer Science Education*, 1–21.

Madill, R., & McKinley, D. (2007). Gender and technology: A critical review of the literature. *Gender and Education, 19*(3), 269–283.

Markert, R. (1996). Gender differences in computer science education: A review of the literature. *Journal of Educational Computing Research, 14*(3), 235–247.

Metz, S. S. (2007). Attracting the engineering of 2020 today. In R. Burke & M. Mattis (Eds.), *Women and minorities in science, technology, engineering and mathematics: Upping the numbers* (pp. 184–209). Edward Elgar.

Modan, G. (2019). Teaching computational thinking in early childhood education: A review of the literature. *Early Education and Development, 30*(2), 174–191.

National Research Council. (2011). *Early childhood assessment: Why, what, and how*. National Academies Press.

Noh, S. D., & Lee, S. M. (2020). Computational thinking in early childhood education: A systematic review. *Computers & Education, 148*, 103798.

Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books.

Pomson, A., & Wertheimer, D. (2017). The role of Jewish early childhood education in fostering a sense of belonging. *Journal of Jewish Education, 83*(1), 5–23.

Reisman, T. (1979). *Roots and wings: A sourcebook of Jewish values*. Union of American Hebrew Congregations.

Sarna, J. D. (2004). *American Judaism: A history*. Yale University Press.

Seow, P., Tan, L., & Tan, C. (2019). Computational thinking in early childhood education: A case study. *Journal of Educational Computing Research, 57*(4), 1011–1032.

Shonkoff, J. P., & Phillips, D. A. (2000). *From neurons to neighborhoods: The science of early childhood development*. National Academy Press.

Sullivan, A., & Bers, M. U. (2013). The role of coding in early childhood education. *Early Childhood Education Journal, 41*(6), 445–452.

Tissenbaum, M., & Sheldon, J. (2021). Computational thinking in early childhood education: A review of the literature. *Computers & Education, 161*, 104084.

Toikkanen, T., & Leinonen, M. (2017). Teaching computational thinking in early childhood education: A case study. *Journal of Educational Computing Research, 55*(3), 1–19.

Vermeer, P. (2009). Religious education and socialization. *Religious Education, 104*(2), 201–213.

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35.

Wing, J. M. (2011). Research notebook: Computational thinking—What and why? *The Link Magazine, 6*, 20–23.

Yadav, A., Hong, H., & Stephenson, C. (2016). Computational thinking for all. *TechTrends, 60*, 565–568.

Zhanxia, Y., Blake-West, J., Dandan, Y., & Bers, M. U. (2025). The impact of a block-based visual programming curriculum. *Learning and Instruction, 95*. https://doi.org/10.1016/j.learninstruc.2024.102041