A Euro-style game for the λ-calculus

Andrew McKenzie
University of Kansas

One of the most daunting aspects to learning formal semantics is the lambda-calculus. Devised in the 1930s to provide a simple method of computation, the λ-calculus proved very helpful for modeling how languages combine the meanings of linguistic expressions to create the meanings of entire propositions. As a result, it forms the foundation of modern formal semantic investigation.

However, its mathematical formalism requires a degree of abstraction that comes with a steep learning curve. If the student does not surmount this curve, their difficulty only compounds as they progress onto more complicated uses of the calculus, like applying functions as arguments to other functions. As a result, great attention must be paid to ensuring student mastery of the calculus, both in its nuts and bolts and in its creative usage.

That said, the semantics course cannot get bogged down in lambda's. Linguistics students have little interest in a glorified math course. We as instructors must shepherd them quickly to a stage where they can confidently apply the calculus to actual examples of natural language, and answer questions they did not know could be asked. Students need practice in handling the abstraction so that they do not have to think about how the calculus works when they try to apply it.

To that end, I have devised a board game, called *f(x)* [read: "f of x"]. This game has the object of gamifying part of the process of mastering the λ-calculus. In it, students gain experience manipulating the pieces of the calculus on unpredictable problem sets, in a popular game format that adds an element of ludic distraction from the task at hand.

1. Teaching the λ-calculus

At its heart, the λ-calculus is relatively simple: It indicates arguments that you plug into an expresssion to end up with a new expression (Alama & Korbmacher 2021). For instance, the expression *λx. x is red* says 'to get a complete expression, replace *x* with something to say that it is red.' If you plug in simple argument 'the cat', you get 'the cat is red'. This proposition will be true or false depending on the facts of the world, namely the color of the cat. However, that truth-judgment is separate from how the proposition was built.

In semantics, it becomes a little more complex because the expressions are typed. Instead of a simple expression like *λx. x is red*, we can limit *x* to entities (which we say are of type *e*). This is a bit like a selection constraint on what you can plug in. Another element of complexity comes when you plug one function into another function. For instance, the expression *λf. the cat f* indicates 'to get a complete expression, replace f with something to say what the cat is doing'. Whatever *f* is, it has to be a function of a type that the cat can plug into, as we saw above (in this case, type ⟨*e,t*⟩). Plugging in a function argument is trickier than a simple argument for a number of reasons, but mastering that skill is crucial for understanding the composition of most functional items: Determiners, quantifiers, aspect, and so on.

Teaching this calculus can turn into a slog of explanations and exercises, sapping the energy out of a class. To avoid that I had previously flipped the semantics course, having students learn concepts using videos, then spending class time on problem solving and applications. Freeing up class time this way gave more time for games.

2. Gamifying the λ-calculus

Making a game has several key motivations. The main one for me was improving the quality of problem sets. Training students on tailored problems is very helpful for beginning stages, but does not lead students to a confident mastery of the λ-calculus, which also requires understanding how to choose the right parts of it for the problem at hand. I figured that instead of spending countless hours trying to concoct problems that worked or didn't work, it would be better to have problems come about without my intervention. This naturally leads to cards or computing. I did not want to learn how to make a video game, so cards were my choice. I had experimented with this kind of card drawing in the past for other exercises. For instance, you draw a modal force and a modal base and have to make a sentence with it. However, these clumsy attempts seemed like half-hearted gimmicks that did not inspire participation.

Also, I found that with the λ-calculus specifically, it could help students if they handled cards. Working with the calculus requires manipulating formal objects in abstract space. Some students struggle with operations like that. What better way to help them get a handle on things than by literally manipulating actual objects in real space?

Finally, I thought it would be fun for me to come up with a semantics game that I would like to play. As a fan of gaming, I am interested in crafting my own game, but lack the time or commitment to make a full-fledged professional effort. Fitting a smaller-scale game into work seemed like a perfect way to unite my wants and my needs.

3. The game itself, *f(x)*

The game is called *f(x)* (pronounced "f of x"), which represents a standard function application that anyone familiar with the λ-calculus will recognize. It also lets you know what the game is about. While there are a number of rules of semantic composition (Heim & Kratzer 1998), this game only involves functional application.

The game involves a board (Figure 1) and pieces. I used Adobe Illustrator to design the board, with little touches like textures, tree marks, foliage, and city streets (Figure 2). I strongly wanted the game to look and play like a 'real' game, rather than something swiftly cobbled together for a linguistics class. To that end, I also printed the boards onto thick 12 x 18-inch paper for a sturdier backing, and supplied colored wooden crafting tokens for players to use as pieces.

I also added a role-playing conceit. Many board games have one—in *Monopoly* one is a real estate investor; in *Clue/Cluedo*, one is a houseguest trying to solve a murder. More recent `Eurogames' typically root their themes in history or fantasy, from the railroad tycoons of *Ticket to Ride* to the role-playing fantasy of *Gloomhaven*. In this respect, *f(x)* is a bit like a Eurogame, though its theme is not terribly deep or too lavishly executed. In *f(x)*, players are trying to return home from the fair, but the short route is washed out and they have to take a long route. This route is designed to resemble a λ.

As the game begins, I hand out rulesheets and walk through a bit of play, to give people the idea. However, most of the game is learned on the fly, as it is not a complicated process.

Players begin by setting a piece on the start space. Then, they each draw three expression cards (example in Figure 2), which they leave face up so others can see. Each expression card has an expression with its semantic type. The table is set up with three λ-expressions, with one argument each (yellow card), and one value space (in blue) to fill in (Figure 2). The basic actions of play involve trying to resolve the λ-expressions on the board using the expression cards.

On each turn, a player begins with an **act,** where the player chooses two actions among the following choices: **Apply** an expression card, **switch** two arguments or two values, **exchange** expression cards, or **trade** expression cards with another player. The board lists these steps to remind the players without having to refer to the rulesheet.

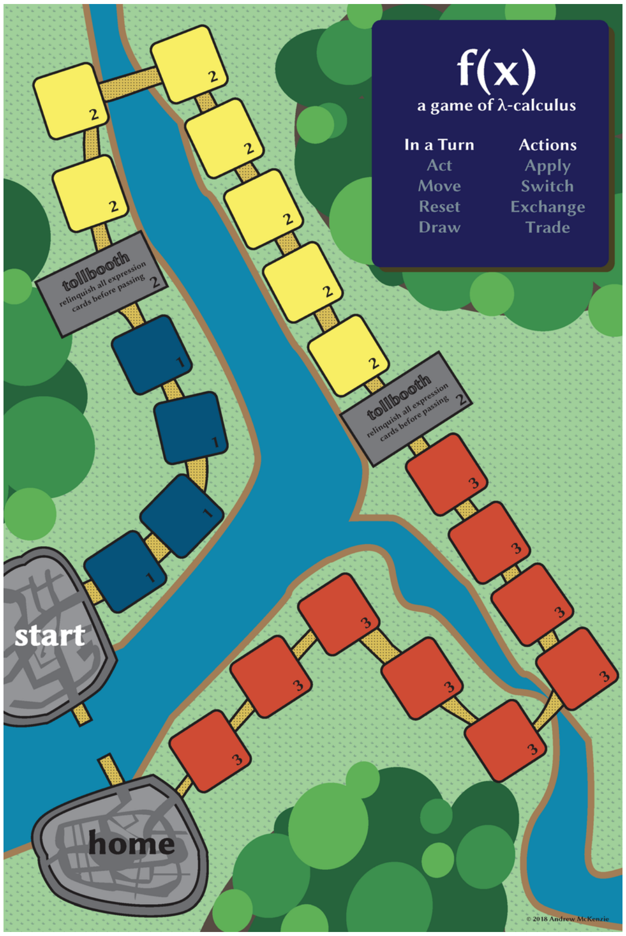

Figure 1: The gameboard. The shape of the path makes a λ

Figure 2: gameboard detail

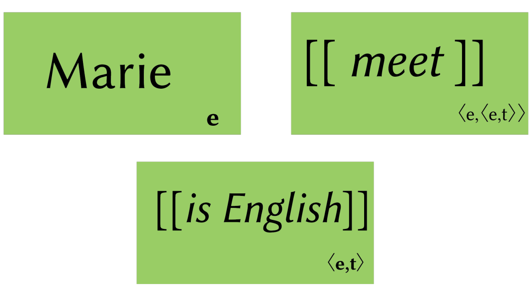

Figure 3: expression cards dealt to player

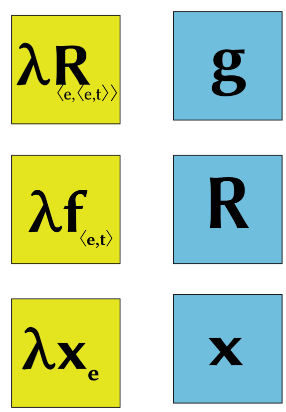

Figure 4: Argument cards and value cards

Applying an expression card involves placing an expression on an appropriate value spot. In this example, a player could apply "Marie" on to the x, because it is of the right type (in this case, *e*). In that case, they can remove the λx. as one would in an actual assignment, and discard it (Step 1 of Figure 5).

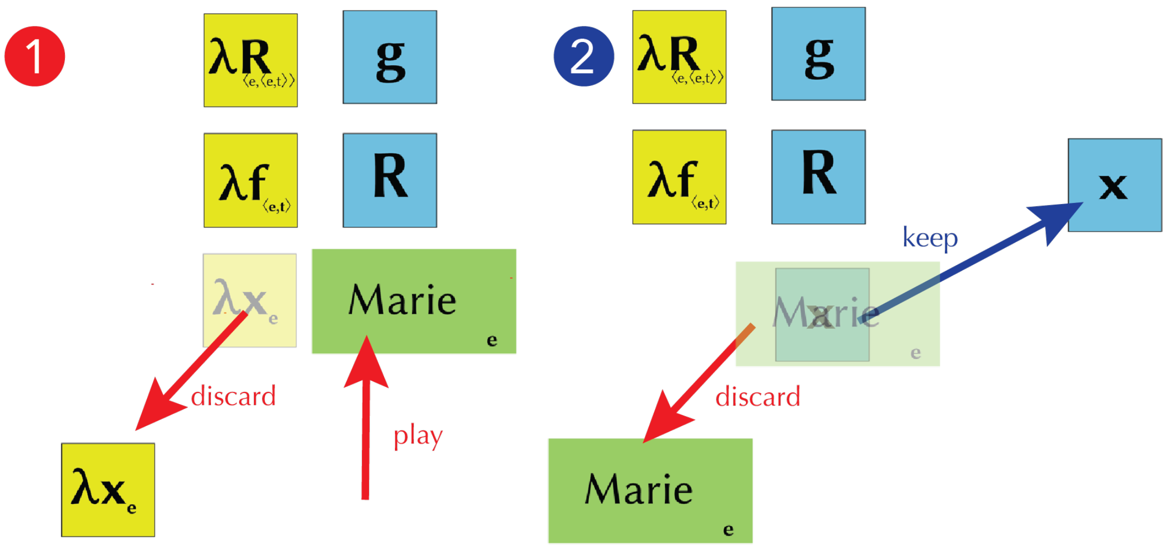

Figure 5: Applying a card and saturating the value card

Since the value card is now completely saturated, the player can remove that and keep it for a point (Step 2 of Figure 5).

For their second move, the player cannot place ⟦*is English*⟧ on g. The type is correct but the variable is not; plugging into λf replaces f in the value expression. They cannot place ⟦ *meet* ⟧ on R, because R is not linked to λR. The player would thus have to switch the value cards g and R or the argument cards λf and λR so the arguments and values can be saturated (Figure 6).

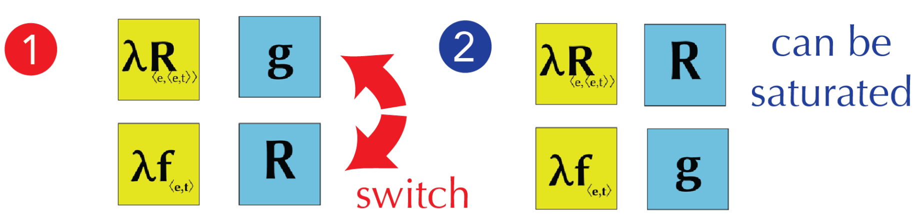

Figure 6: Switching value cards to allow for saturation

Trading a card involves trading a green expression card from your hand to another player, whose hand is visible. The other player must consent to the trade.

Exchanging involves replacing a blue value card on the board with the next one in the deck. The player takes a risk in an exchange because they do not know whether it will help. Figure 7 exemplifies an exchange that does not pan out.

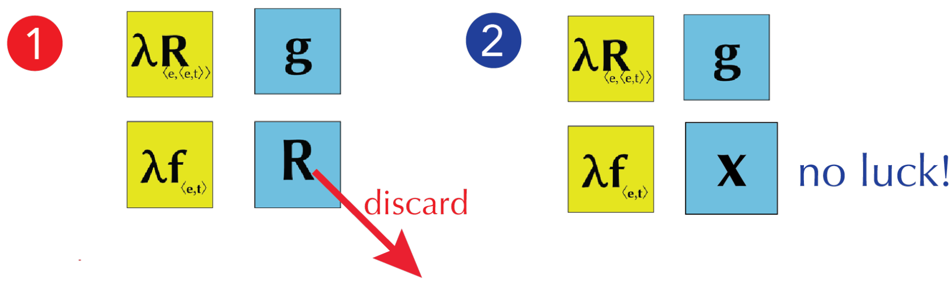

Figure 7: Exchanging value cards

Players can also be antagonistic and exchange a card on the board to make it harder for the next player to saturate anything.

After this second choice, the player moves their piece, and replaces any value and argument cards on the board (so there are always three expressions). They then draw cards to refill their hand up to three.

As the game progresses, the game leads players to place bigger expressions on the board. This way, the game starts simple and gets more complex as players gain experience. After a short while, the player in the lead reaches a "toll booth" where they relinquish the cards from their hand, and only take one back, ending their turn. This slows down players from running away with the game, but also opens the next part of the board up. As we can see in Figure 1, the board spaces have a number on them; this tells the players what size expression to put on the board. Early in the game, the number 1 indicates they should put one argument and a one-place value card. Players landing on a "2" spot should put two arguments a two-place value card. Later, when they land on "3" spots, they will put three arguments and a three-place value card.

After a while, a player with the hand in (Step 1 of figure 8) might see a board like the following (Step 2):

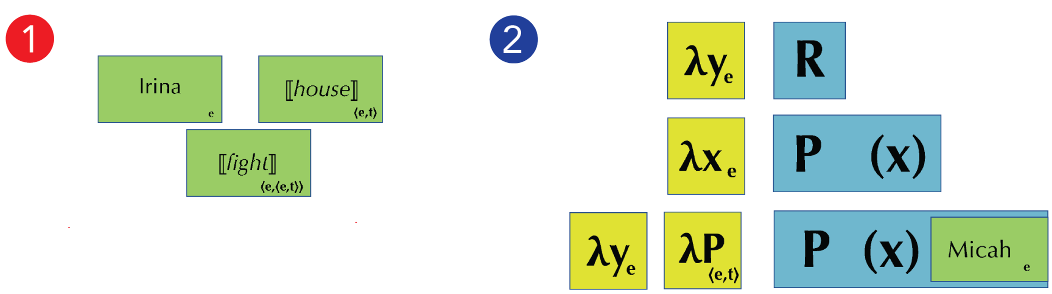

Figure 8: A player's hand (1) and board setup (2) later in the game

Here, a player has a number of choices they can make. They can play Irina on λx... (x). They cannot play λy, because the variable of that type on the corresponding value card is (x). They cannot play λP because it is on the inside of the expression— only the leftmost lambda argument can be saturated. The player can switch λy with λP, which correlates to the process of currying/schönfinkelization. However, this won't help them here, because the type of λP is $\langle e,t \rangle$, and that doesn't fit with the P on the blue value condition, which has two arguments of type *e*, so it is $\langle e,\langle e,t \rangle\rangle$. They can switch λP with the λx above it, and that would allow a saturation, and then play $[\![\textit{house}]\!]$ for their second play. There are other less valuable plays as well. In this case, there is no blue value card that this player can close out. Whoever saturates the last remaining argument gets the card as detailed above. The card is worth one point for each argument.

As far as exchanging goes, the player can replace the blue value card R, or the one with P(x). However, the value card with "Micah" played on it cannot be exchanged. This restriction adds a bit of strategy for plays, as a player can block a later exchange by playing an argument, while risking setting it up for someone else to get the card.

The game itself takes about 20-30 minutes to play. In a 75-minute class, that usually allows for two rounds, sometimes three. After a few turns, the players pick up the rules and things move quickly. They soon get in the habit of applying and manipulating λ-expressions at a rapid pace (Figure 9).

Figure 9: Students playing *f(x)*

4. *f(x)* as a Euro-style game

I designed *f(x)* as a "Euro-style" or "Eurogame." This type of boardgame emerged in West Germany but has become immensely popular throughout the Western world in the last 25 years (Woods 2012). Much of the modern resurgence of board-game culture has largely been driven by it. Some of the earliest games to break this genre open are now widely known beyond gaming circles: *The Settlers of Catan*, *Carcassonne*, *Ticket to Ride*, and so on. As a result, college students are usually familiar with the style if they know board games at all. Eurogames are known for characteristics that aim to enhance player participation from start to finish, and that enhancement is very helpful for a classroom environment.

Characteristic 1: *Player conflict is reduced.* In some games the goal is to win by making other players lose. *Monopoly* is an infamous example, a source of many conflicts even in families. There is no real conflict in *f(x)*. There is no way to steal cards or send other players back to the start, etc. Players can still be competitive, strategizing to stymie their opponents, or they can cooperate. I find that generally, players cooperate, trading cards and helping each other figure out what to plug in.

Characteristic 2: *Skill is promoted over randomness.* The role of randomness (or near-randomness) in gameplay can vary. In dice or card games (*Yahtzee*, *Uno*, etc.), randomness is a key feature. In *Exploding Kittens*, a player is eliminated by drawing a particular explosion card from the top of the deck. Randomness can keep things spicy, but it runs the risk of killing the joy of accomplishment a player gets from doing well, or leading to victory that seems cheap. In a game that is building a skill, it crucially helps to reward skill. The only random element in *f(x)* is the drawing of cards to set on the board and in the players' hand. It is up to players to deal with this, though. Movement is determined by how the player decides to act during their turn. The luck-of-the-draw factor is mitigated by the ability for players to exchange or trade the cards that come up.

Characteristic 3: *The game is relatively close throughout.* One of the greatest risks of board games is failing to maintain player interest. Some games essentially require one player to run the others off the board one by one (*Risk, Monopoly*), making the outcome clear well before the end. Many other games eliminate players as play progresses (*Clue/Cluedo*, *Exploding Kittens*), leaving them to sit around or wander off. Some games have turns that take so long that players can forget they even are playing (*Axis & Allies*). Flagging interest is obviously detrimental to a classroom environment, and *f(x)* is designed to avoid it.

The game ends when one player reaches the home, but that does not mean that player wins. At the end, everyone tallies up their won argument cards with where they are at on the board, and the most points wins. This keeps the score under suspense, which reduces the motivation to give up. Also, the game is set up to allow for catching up. When one player gets ahead, they trigger larger expressions for the board, which allow for faster movement, and the trailing players get first crack at them. Also, players in first place have to pay a toll to open the road for everyone else, and this reduces their scoring ability for a couple of rounds. In the future I plan to allow an extra move as an action players can take, so they can simply choose to race faster around the board and let others mess with the cards.

5. Pedagogical results

At its heart, using *f(x)* devotes an entire class session to doing exercises with the λ-calculus. Instead of printing out staid problem sets of mind-numbing abstraction, this exercise provides unpredictable expressions with no indication of what methods to use to tackle them or repair them. I did not conduct an actual study measuring results. However, participation remained steady throughout the class, and some students had positive comments. Later on, some left positive notes about this game (and others) in end-of-course evaluations. They also were very excited later in the semester when we played another game related to aktionsart, which I took as a sign that the previous game (*f(x)*) had been a hit.

Most importantly, I found subsequent to playing *f(x),* that I had to spend a lot less time correcting students on the nuts and bolts of functional application than in previous semesters. They were notably better when it came to lining up arguments and ruling out structures because they could not be saturated. These skills are crucial to analysis in semantics and increasingly in syntactic theory, and pay off the rest of the semester and beyond.

I also got the chance to observe leadership skills. The game puts students in the habit of teaching each other how the calculus works–sometimes just to enforce the rules, sometimes out of a cooperative spirit (Figure 10). Certain students grabbed the torch and were in a position to lead. Since the 'teaching' was ensconced in a gaming format, leadership did not seem to inspire as much reticence as it would in ordinary class environments, where students often hesitate to 'lord over' their peers, or not come off a know-it-all.

Figure 10: Students teaching each other with *f(x)*

6. Summary

This chapter discusses *f(x)*, a Euro-style game designed to help semantics students master functional application with the λ-calculus. It offers a fairly randomized problem set that offers students a more realistic approach to using the formal tools in language analysis, because they have no idea if what they are about to see even works, but they have to figure that out. The gameplay trains students to manipulate the pieces of the calculus and build the skills of distinguishing the parts that compose and the parts that do not, in an environment that lets them strategize, cooperate, and share their knowledge with each other. It also allowed me as an instructor an outlet for creativity that was not too onerous or time-consuming.

Acknowledgement.
My thanks go to the students who learned with this game and offered advice for its improvement. The students in these photos freely consented in advance to their photos being taken during gameplay, and to the eventual use of those photos in publications about the game or publicity for it.

References

Alama, Jesse and Johannes Korbmacher. 2021. The Lambda Calculus. *The Stanford Encyclopedia of Philosophy,* Edward N. Zalta ed. (online, viewed Nov 29, 2022)
https://plato.stanford.edu/archives/sum2021/entries/lambda-calculus

Heim, Irene, and Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Malden, MA: Blackwell.

Woods, Stewart. 2012. *Eurogaming*: *The Design, Culture and Play of Modern European Board Games.* Jefferson, NC: McFarland & Company.