[SQUEAKING]

[RUSTLING]

[CLICKING]

IAN BALL: So I wanted to start by playing the beauty contest again. Last week-- for two reasons. Last week, I kind of messed it up, and I described a mixture of the game. We actually played a more complicated game that I play in my other class, so I want to set the record straight here. And I also want to see how the results maybe have changed since the first time we played.

So if you haven't made an account yet, you can log in here and put in the password or put in the class code, try to join this game now. And let me say a little bit about how it works. So remember, this is the beauty contest. You're going to guess a number between 1 and 100. The computer is going to collect everyone's guesses and compute the average of those guesses. And the winner is going to be the person whose guess is closest to 2/3 of the average of the class's guesses.

So let's start, play this, and see what happens. So did we learn from last week? Let's see. So in this case, the average choice was 18. And the winning choice was 12. So I think last week, what was it the first time? Maybe 25 was the winning bid. So what you see is, again, we're shifting a little bit. I was actually expecting it might shift a bit more. No one went to close to 0.

If we really were ultra, ultra rational, there's an argument that everyone should guess 0. And we'll see the formal argument for that later in the course. But if you're not sure if everyone else is rational, then maybe this is a pretty reasonable performance. Again, we had a few people going a little crazy and guessing 100. I think that's probably not a good idea. So let's put this down and get started.

So let me start with just a very brief review of what we did last class, or I'll say yesterday, but I guess last class. So last class, we talked about individual decision making. This class, we're going to start talking about interactive decision making, where we have multiple players making decisions. And the best decision for each of them depends upon the decisions that other players are making. So just some notation to remember from last class-- it's really just two pieces you need to remember.

From last class, we said that there's some set z of consequences or outcomes. And we defined a Von Neumann-Morgenstern utility that assigns some real number to every single consequence or outcome. I'll use those two terms interchangeably. And then we defined capital U to be the expected utility. So what this does is it assigns to every lottery, every lottery over consequences, the expected utility that the agent gets from that lottery. And remember a lottery would be something like this.

So if there are m consequences, then a lottery is a vector p that specifies m different probabilities that sum to 1. And the interpretation is that p1 is the probability, under this lottery, that outcome that the first consequence is realized, all the way up to pm, which is the probability that the mth consequence is realized.

And I realized last week, I think I used p in two different ways, both as a probability and as a vector. So here this is a vector. Maybe I'll put a hat on it here. But I may not use hats throughout the course. But keep in mind that p will usually be a vector. But these little p's are numbers. These are probabilities inside the vector.

So now today, the plan is to move to interactive decision making. And let me just give an outline of the three topics we're going to cover. The first is what we're going to define what we call an extensive form game. This is one way of representing a game.

Well, what is a game? Well, think of a game as a multi-agent decision problem. Yesterday, we talked about the decision problem faced by an individual. Today, when we think about multiple individuals making choices that interact, we're going to call that a game. An extensive form game is going to be a very detailed representation of the game that specifies all the details that might possibly matter about the game.

Then later, we're going to talk about a strategic form game, which you can think of as a summary of the extensive form game. It isolates or identifies the essential elements of the game for strategic analysis, and it throws out some of the details that are not explicitly relevant when we're analyzing the game. And the key way that we jump from an extensive form game to a strategic form game is the notion of a strategy. And I think this is the single most important definition in the course.

So after we talk about extensive form games, we'll define what a strategy is, and then we'll move to strategic form games. I think most points that people lose throughout this course on exams are because they haven't thought carefully about what a strategy is and what the right definition of a strategy is. So I'd really encourage you to focus on this definition.

So let's-- maybe I'll start a new board here, actually. So I want to start with two examples of extensive form games. And then motivated by those examples, I'll present the general theory. But I think if I do the general theory first, it's just a lot of notation, and it's not very concrete.

So let's start with two really simple examples. The first is an investment game. And this will actually only have one strategic player. Games don't have to have multiple players. They're usually more interesting when there's multiple players. But it's allowed for there to just be one player. So let's think that there is a single player. We'll call them player 1. And we're going to represent this game using what's called a decision tree, which you may have seen in some other courses.

So this one player is choosing whether to make a safe investment or a risky investment. So they have two actions, safe and risky. And the safe investment is always going to pay the same amount of money. But the risky investment may pay a lot of money, or it may pay a little money. They're not sure. So the way we're going to represent that is, under the risky investment, there's actually going to be a move by a player that we'll call Nature.

Now, Nature is a special kind of player. They're not a strategic player, like everyone else. But we often find it helpful to think of them as a player when we represent this game. And here, we think of Nature as choosing what the payoff of this asset is going to be if they make the risky investment.

And in this example, there's two possibilities. The payoff is only $3. The monetary payout is either $3 or $1. And I have to tell you the probabilities of these two outcomes. So whenever Nature chooses something, Nature chooses that from some distribution, from some lottery. And I'm going to indicate the probabilities like this, 1/2 and 1/2.

Now up here, I don't really need to use Nature, but maybe I will, just to be really precise. So here Nature is always going to choose a payout of $2. So maybe if I really want to be precise, I'll put a 1 here. And this is a one, not a 7. Sometimes I get a little crazy with the hat. So this says, if I choose the safe investment, I'm certain to get $2. If I choose the risky investment, then there's a 1/2 chance that I get $3 and a 1/2 chance that I get $1. But I would argue that this is not a complete specification of the game. What's missing here, if I want to specify this problem? What have I not told you? Yeah?

AUDIENCE: Preferences.

IAN BALL: Preferences, right? And it's attempting to say, oh, well, all I want to do is maximize expected monetary amount. But as we talked about last class, some people are risk-averse. Some people are risk-loving. We have to capture preferences. So what we need to write down is what my utility is from each of these outcomes. So we're going to specify my utility from $2, my utility from $3, and my utility from $1. And there's a lot of choices we could make. But maybe we'll say, 1.5, 2, and 0.9, just as some example.

So it's important to understand that the game specifies the consequences but also the payoffs from those consequences. And it can be particularly tempting when you have monetary payouts, so when the consequences are money, to get confused about that distinction. But there's still a crucial distinction between the consequence, which here is measured in dollars, and the utility the player gets from that consequence, which is measured in utils. So this here is a Von Neumann-Morgenstern utility, like we talked about last class.

So that's a very simple game. There's one strategic player together with Nature. Let's now do a game that actually has two players. So let's really get into game theory. And I'm going to call this the BoS game. So historically, this game is called The Battle of the Sexes. I think today I'd like to try to avoid that terminology. So I'll just call it BoS. Think of it as Boston. But if you read old game theory books, they'll often call this the Battle of the Sexes game.

So what's the idea of this game, or at least the version we're going to tell? You have two friends, and they're choosing between two activities. So let's say since it's Boston, let's say going to a Celtics game or going to a Red Sox game. And the idea is they have to choose which activity to do. And they're good friends. So if they go to different activities, they don't have any fun. If you go to the Celtics game, and I go to the Red Sox game, we're each all alone, and we're both sad.

So we'd both like to go to the same activity. We'd like to be together at that activity. But one of us would rather that we be together at the Celtics game because they're more of a basketball fan. And the other person would rather that they be together at a Red Sox game. So why I think this game is so important, and why it's gotten a lot of attention, is it features both conflict-- maybe conflict and cooperation.

It has a conflict of interest because there's one player who'd like that you both go to the Celtics game, and there's the other player who'd like you both to go to the Red Sox game. So you have some conflict of interest between where you go, but you also-- and maybe cooperation, or coordination is maybe a better term. Let me say coordination.

But there's also a coordination aspect to it because you'd like to go to the same activity. You'd like to be at the same place. And this combination of conflict and coordination is present in a lot of situations far beyond this very simple game. So let's draw this out. We're going to say player 1 is first going to choose whether to go to the Celtics game or the Red Sox game. And I'll call this C and R. And let me just be clear, when I say Celtics game or Red Sox game, that's not the formal sense of a game we're talking about here. That's game as in an athletic game. So player 1 is choosing between C and R. And then player 2 is choosing also between C and R.

Now, the outcomes are kind of already immediate from the way I've written it. What is the outcome here? The outcome here is that we watch the Celtics game together. And the outcome here is that player 1 watches the Red Sox game by yourself, and player 2 watches the Celtics game by yourself. But again, if we want to fully specify the game, we have to say what payoffs are. So we're going to write down payoffs here. And what I'm going to assume is that player 1 is the player who prefers the Celtics. So if they both go to the Celtics game, the payoffs are 4, 3.

Now, notice a key difference here. With a single-player game, I only had to specify one utility at each of these consequences. But now that I have two players, I have to specify each player's utility from that consequence. And the convention we're going to use is that if we have players 1 and 2, the first player's payoff is going to come first, and the second player's payoff is going to come second. So this is how we specify everyone's payoffs from this consequence. If we had more players, we'd keep having commas, and we'd keep adding more numbers.

Now the other one we'll write down is R, R. So this is, again, where we've coordinated on going to the same sports event, but we've coordinated on the Red Sox game. And player 2 is the one who likes the Red Sox more. So here, this down here, is going to be 3, 4. And then, to keep things simple, we'll make the other payoff 0, 0. So the idea is if you go to the game by yourself, you're so sad, it doesn't even matter who's playing. You're just sitting by yourself, and you're sad. You get a payoff of 0. But here we see the conflict.

So a very simple game-- I don't know. In this game, who would you rather be? Do you have any intuition about what might happen in this game or which player you'd like to be? Who would you rather, be based on the way we've written the game? Yeah?

AUDIENCE: Player 1.

IAN BALL: And tell me why.

AUDIENCE: Because they pick first.

IAN BALL: Because they pick first, so exactly. I'm getting at a crucial thing. What is implicit in the way I've written this, and what you see is important for the analysis, is not just that they pick first, but that player 2 observes the pick that they make. So implicitly, the way I've written this is that player 2 gets to see what player 1 does. And can you walk me through why if player 2 sees what you do, why you think player one is in a really strong position? So what's going to happen? What's your prediction about what would happen?

AUDIENCE: No matter what player 1 does, player 2 is better off if they follow suit.

IAN BALL: Right. So in this case, I'm player 1. I like the Celtics. What I want to do is I want to go to the Celtics game and make sure you know I'm going there. And once player 2 already knows that I'm at the Celtics game, well, they don't really want to be alone, so they're going to follow me to the Celtics game. So in this case, the fact that player 1 moves first, and that move is observed, gives them the power to select which game they end up going to.

But what you can see is if player 2 didn't observe what player one did, then this would be a quite different game. An alternative game would be we move simultaneously. It's always a bit harder to motivate this in the world of instant communication. But in the old days, people would have to agree about a place to meet up, and they might simultaneously choose where they would go to meet up. And you might not observe where the other person had chosen to go. So we need notation for that.

So if we wanted to say the version of the game where player 2 does not observe what player 1 did, we would indicate that using a dotted line here. So the dotted line-- maybe I'll write this over here-- means player 2, P2, does not observe player 1's move.

So you can think of this essentially as a simultaneous move game. Technically, the way we wrote the tree, player 1 is moving first, and player 2 is moving second. But because player 2 doesn't get to see what player 1 does, it's effectively as if they were moving simultaneously. Are there any questions on these games here?

So now what I'd like to do is present the general definition of a game. So these are two examples of a game that we want to capture. And we want to introduce a much more general definition that can cover all the examples that we're going to discuss in this course. So formally-- I'll do it over here. An extensive form game-- well, first we need a word for this decision tree that I drew. Formally, this is what's called a rooted tree.

So let me say an extensive form game is a-- this is a fancy name. All it means is it looks like this. The root is where you start. This is the root. And then it comes out like this. Sometimes people write trees vertically. Sometimes you write them from left to right. It doesn't really matter. The formal notion of a tree is that at any point I look at, there's a unique path back to the root.

So just so we understand what's not allowed, if I had something like this, this would not be allowed because here's my root. If I look at this point, there's two different ways to get back to the root. So this is not a graph theory course. We're not getting into too many technicalities. But basically, there can't be any loops or any cycles in the tree. And that's what we mean by rooted tree.

And the interpretation of the root is that's where things start. So remember, implicitly, when we write an extensive form game, there's some notion of time. We think of time starting at the root. And then as we move forward in time, we move throughout the tree. Let me erase this, since this is not what we allow.

I want to make-- so it's a rooted tree specifying four things. So I want to first specify each of them kind of informally. And then I want to talk formally how they appear on the graph. So there's four things it specifies. And I like using the acronym PAPI to remember it. So this is players, actions, payoffs, information. So let me write that down. Yes.

AUDIENCE: Are these required to be finite trees?

IAN BALL: Good question. We're going to do what's often done in an intro game theory class. And when we formally define it, we'll assume things are finite so we don't have issues. But then we'll go to some examples where things are infinite. So basically, what I would say is, to carefully define infinite games, there are some issues. So we'll only consider infinite games where things are very well-behaved. And we won't formally define exactly what well-behaved is. It's just if you see an infinite game, trust that it's going to be well-behaved.

But if we think of the Keynesian beauty contest that we started with, we required you to choose a number between 1 and 100. It had to be an integer. But what if you could choose a real number between 0 and 100? Now all of a sudden, we have an infinite game. And that could be a realistic model or a useful model. Yeah. Great.

So P is for players. A is for actions. P is for payoffs. And I is for information. So I could define these things, but I think people should have a pretty intuitive sense of what these mean. The players are the agents, the actors, in our game. The actions are the choices that they make. So for instance, in this game, player 1 chose between safe and risky. Over here, player 2 chose between going to the Celtics game and going to the Red Sox game.

Payoffs are what we define at the end. This is the utilities that people get from the consequences of the game. And then information, we're going to have to be a bit more careful about defining this, but it captures what we were initially discussing here, things like does player 2 know what player 1 chose when it's player two's turn to move?

And I want to just put a few synonyms up here because I have a tendency, throughout the course, to use a few different words to mean the same thing. And I just want us to be all on the same page about what those mean. So I might say action, or I might say moves. Those mean the same thing. And I might say payoffs, or I might say utilities. And those mean the same thing.

One really important thing is payoffs always means utilities. So even in this game here, here I would say your payoff is 1.5. What I mean by that is your utility is 1.5. I'm not talking about the monetary payoff. So if I don't use the word monetary, we're always talking about utilities. Why? Well, it's really the utilities that matter. The monetary payoffs are in the background. But everything really comes down to the utilities.

So players-- I think intuitively we all know how we need to specify this, but I just want to be a little bit more formal. Well, the players, the way we specify it, is we put labels on the nodes in the tree. But notice, not every node gets a label. In this game, these nodes don't get a player label. They get a utility label. So really, if we look at this game, we see there's two different kinds of nodes. There's the nodes like this, 1, 2, 3 which are decision nodes. These are nodes in the tree where people take decisions or choose-- maybe I'll say actions, moves. Maybe I'll add one more synonym, decisions.

And then we have nodes like this, which are at the end of the game. And these specify consequences. So let me say the rooted tree-- let me go a little up here-- it has nodes in some set H. Why do we call the nodes capital H? Because H reminds us of history. And with a tree, every node exactly tells us the history of play in the game.

Let's look at this. At his node, well, nothing has happened. So the history is nothing. At this node, the history is player 1 chose C. And at this node the history is player 1 chose C. And then player two chose R. Notice that this gets back to the problem with our example here. If we had two paths that went to the same node, well now, interpreting this as a history becomes unclear, because if we're here, well, did we get here by following that history? Or did we get here by following that history? And that shows we've done something wrong in the way we've specified the game. We want to uniquely specify the history of actions at each node in the game. And that's why we rule that out.

So within the set of nodes H, we'll say Z, which is a subset of H, these are the terminal nodes. And then H minus Z, these are the decision nodes. So I could define this formally. But I think we should know what this means. Z subset of H means there's some subcollection of nodes that are the terminal nodes.

In this game here, the set H contains, well, 1, 2, 3, 4, 5, 6, 7. So in this game, H would have 7 nodes. 4 of them are terminal nodes. So these 4 nodes at the end would be in the set Z, which is a subset of all the 7. So we have 7 histories, 7 nodes altogether. 4 of them are terminal nodes in the set Z. And then you may not have seen this notation before, but H set minus Z means everything that's in H that's not in Z.

So over here in this example, we have 7 nodes altogether. 4 of them are in Z. What's left? What's not in Z? Well, it's these 3 nodes, 1, 2, 3. And these are called decision nodes because these are nodes where players make decisions. Once we get here, there are no decisions to be made. These are just consequences.

It's not a coincidence that we use the term Z for the terminal nodes because Z is exactly playing the role that Z played last week in our individual game. So at the terminal nodes of our game, we have some consequence, which lives in some set Z, and we have Von Neumann-Morgenstern utilities over those consequences. Of course, before, we just had one vNM utility. And now, we have one for every player. So we'll talk about that in a second.

So what are the players? We'll usually label them i equals 1 up to n. So we could have a one player game a two player game. These are the strategic players. And then sometimes we also have Nature. So maybe I'll say and Nature. So Nature is kind of a special player that we sometimes also add to the game if there's uncertainty in the game. So over here, this game had player 1 as a strategic player, and it had Nature. This game had two strategic players player 1 and player 2.

Now, we need the actions, moves, or decisions. Well, crucially, these are, I'll say, labeled on the edges of the tree. And we could be really formal and precise. But I think it's easier just to see what we mean. So let's take this history here. It's player 1's turn to move. At this node, player 1 has two choices or two actions, safe and risky. We represent those two actions with edges coming out of the node, and we label each edge with the corresponding action.

Similarly, over here, when it's player 2's is turn to move at this node, player 2 has two choices, C and R. We represent those with edges. And we label them according to the actions that they are. And if you have a math background, you may think, oh, these labels are kind of arbitrary. It turns out the labels are quite fundamental and something we'll do later. So the labels are not arbitrary. I'll make that point now. Payoffs or utilities-- well, based on what I've said, can anyone guess how we're going to specify the payoffs, utilities in this game? Any thoughts? Shy? Yeah?

AUDIENCE: A set--

IAN BALL: Say again?

AUDIENCE: A set of utilities associated with [INAUDIBLE]?

IAN BALL: Exactly. And the way we're going to write that is we're going to have n utility functions, one for each player. So we're going to have utility function ui from Z to R for i equals 1 through n. What does that mean? It says for each player, i, we're going to specify that player i's Von Neumann-Morgenstern utility function over all of the outcomes. So this is going to assign a real number in R to every single outcome of the game.

Now, notice we've grouped things a little differently. Here I'm grouping things by player. So for player i, this function says, how much utility does that player i get from every single consequence? Well, the way we wrote it on the tree, we actually grouped it by consequences. So associated with this node is u1 of that node, comma, u2 of that node.

So if you want to connect it to the definition, all these numbers here would correspond to the function u1. And all these numbers here would correspond to the function u2. So technically what I put here is u1 of this outcome, comma, u2 of this outcome or consequence, same thing. And these are, I should emphasize, Von Neumann-Morgenstern utilities, just like we said last week, so that when players face uncertainty, they evaluate lotteries over consequences using expected utility.

Great. And then the last thing is information is going to be represented with something called information sets. And that's going to take a new board to talk about. So let me pause here and see, are any questions on what I've done so far?

It's a little tight at the bottom. So now, let's talk about information sets. So for each player i-- so let me just say it in words first. We want to look at the game. And we want to look at all the nodes where player i is called to move. And this is important. So up here, when I said players i equals 1 through n, part of naming those players is also assigning a player to every single node in the tree.

And that means that if I fix a given player i, I can ask, what are the nodes in the tree where that player i is called to act? So which of the decision nodes is player i the one making the decision? So I'll say for each player i, the decision nodes of player i-- and they'd all be connected, but just to give an illustration, maybe the nodes are something like this. There'd be lines in between them, but let's just focus on the nodes for now. The decision nodes of player i are partitioned into information sets. And I'll explain what that means.

A partition is a fancy math word to say, we have a group of things. We have a collection of things. And we're going to put them into groups. We're just going to break them up into groups. So for instance, one way of partitioning this would be that. That's a partition. I have five things. I put two things together in this group, these two things together in this group, and one thing by itself in this group.

So the rules of partition are what you think. Nothing could be in two groups. And nothing can be not in any group. So everything's in exactly one group. And that group could be it by itself, like we have here. That's allowed. But think of you have a pile of candy on the ground, and you just push it into piles on the ground.

What is the interpretation? The interpretation is that player i can only tell which group they're in, which information set they're in, but not which node they're in within that information set. So let me say that. So player I observes the information set. So in this case, we have three information sets, maybe i1, i2, and i3. Whenever the player moves, they know which information set they're in. But they don't know which node inside the information set they're in.

Now, of course, if a node is by itself, then they effectively observe which node they're at because if I know I'm in this information set, then I certainly know which node I'm at. But if I know only that I'm at this information set, I'm not sure whether I'm at this node or I'm at this node.

Now here, that's exactly what we were capturing on this graph. But sometimes it's cleaner, instead of drawing a circle around the two nodes, just to put a dotted line between them. So what we did here is exactly representing this information partition. But we just did it in a slightly more compact way. So often, an alternative thing-- maybe I'll give an example. If you had three nodes, or specifically if you have two nodes, you might draw a dotted line between them instead of a circle. And that just means the same thing. But it means things get a little less messy. And I should say, this is not an information set. This is just me indicating something. So pretend that's not there.

And let's go to the interpretation. So what we said, when these are in the same information set, when it's player 2's turn to move, what do they know? They know that they're either at this history, or they're at this history. So let's say that, in words, what they know is player 1 has chosen something. But they don't know which thing player 1 has chosen. Either player 1 chose to go to the Celtics game or player 1 chose to go to the Red Sox game. They know player 1 chose one of those things, but they don't know which one.

If we erase this information set, now these two nodes are in separate information sets. And if they're in separate information sets, player 2 does observe the choice that player 1 makes because they know either they're at this singleton information set that only contains this node or they're at this singleton information set that only contains this node. So they know exactly what choice player 1 made. And this is the formal way that we're going to model observability. Yes.

AUDIENCE: For simultaneous decision making, do we prefer to use the information segmentation, or do we ever take a decision node and have two players kind of belonging to that decision?

IAN BALL: So we're always going to do the first thing. So we every node always has exactly one player. If we want to capture simultaneity, We always capture it through information sets. It's a great question. So nodes are never going to have two players associated with them, always just one. Good question.

So one final comment is if a player can observe everything, then what partition does that correspond to, if I observe everything, I have no uncertainty about who's played before me. Well, then just everything's in its own group. So that would be I have a circle over this. I a circle over this.

And in that case, we sometimes call these-- maybe I'll give a word for that, a singleton information set. A singleton is just a set with one node inside it. So if I'm ever at a singleton information set, I know exactly which node I'm at. And if a player always knows exactly where they are, then their partition consists only of singletons. None of their information sets contains more than one node. Any questions here? Yes?

AUDIENCE: What about other information that might be relevant for decisions, such as the payoffs?

IAN BALL: Good question. So here what we've assumed is that the payoffs of each consequence are known. If there's some uncertainty about what my payoff, payout is, then we need to model that as different consequences. So just like we did here, basically, when I chose the risky investment, we didn't assign a utility directly to the investment. What we said is, well, we don't know how the investment is going to turn out. And we said, if I get different utilities, then those are really different outcomes or different consequences. So we need to split them up. So I would say, consequences always pin down and specify utilities exactly. If you feel like you're not sure what your utility is, then you should split those consequences up and represent them differently.

AUDIENCE: What about in relation to other players?

IAN BALL: Same thing. So if there's some consequence, if I'm not sure what the other player's payoff is, then we should represent that as two different consequences, where the other player gets different payoffs. Maybe I'll get the same payoff at those two consequences, but the other player will get different payoffs. And we'll see examples of that throughout the course.

So I think this framework is actually much more general than it may at first seem. A lot of these assumptions-- oh, we know exactly what our payoffs are. We know all these things-- can capture much richer environments with the appropriate interpretation of what the information sets are, what Nature is, and what the consequences are. Yeah. And was there a question here or no? Maybe not.

Well, I think there's one problem with this definition. And so we need a little more structure here. The problem with this definition is what if I had two information sets or two nodes-- let's take this-- that are in the same information set. But the actions that I'm allowed to take at those two nodes are different. Now we have a problem because we've implicitly assumed that I know what my actions are. If I didn't know what my actions were, then there'd be no way I could even take the action. So we're in trouble. So we've assumed I know what actions I can choose.

But if we had two nodes where I had different feasible actions, then I would definitely know that I'm at different nodes because I could tell that at one of the nodes, I had a different action available. So we need to impose a restriction on the way we define information sets so that property doesn't-- that never occurs.

So conditions on information sets-- and there's really two main conditions that we need. And the first main condition is capturing what I just said, that the player must have-- so I'll say player i has same feasible actions at all nodes in the same information set.

So just to make it a little more concrete, if we want to put this in the same information set, this is OK here because player 2 is choosing between C and R at this node, and they're choosing between C and R at this node. But if I tried to add something else-- let's say, H, I stay home-- now we have a problem because when the player gets to this node, and he knows he's able to stay home, he knows that he must be at this node. And if he sees that staying home is not feasible, he knows he must be at this node, and therefore, those can't be in the same information set. So let me erase this. But that's all that this says.

Notice if all of a player's information sets are singletons, well, then this is automatically satisfied. If I only have one node in my information set, then of course I have the same actions feasible at every node because there's only one of them. So this is what we might say vacuously true, automatically true with a singleton information set. But it has bite if my information set has multiple nodes.

And then there's one more restriction that we put on and we say that the root-- remember this is the initial node-- is in a singleton information set. The idea would be if we don't know where the game starts, and we don't know where we are, then there must be some prior move that we didn't observe, and we should model that explicitly.

So if you don't know where you start, then often the way we capture that is we say that Nature is initially making a move that determines where we start. But then Nature has a node that starts the game. So the root or the initial node must be in its own information set. And you see that that was satisfied here and here.

A few final points-- let me check the time. So I've been a little sloppy in a few things in that I've been talking a lot about players, but Nature is really different than the other players. So we need to specify some different rules for Nature. So let me say, Nature is different. If you just look up at these trees, can you see any differences when we label a node to be Nature, rather than a strategic player, what's different when it's Nature's turn? Yeah?

AUDIENCE: Nature works on probability.

IAN BALL: Exactly right. So whereas a strategic player chooses an action at an edge, Nature just chooses probabilities. So edges have probabilities, not actions. And that's because Nature is not strategic. These just probabilities are given to us. With strategic players, we can't assume that a player is always going to go to the Celtics game with probability 0.2. That's the whole point what we're trying to figure out, whereas Nature, we do think of it like a lottery, like a roulette wheel. These probabilities are just fixed as part of the game. And all that nature does is it mechanically draws an edge according to those probabilities.

Any other things you notice that are different about Nature? Well, one thing, Nature doesn't have payoffs. When we labeled payoffs at the end of the game, we only specified payoffs for the strategic players in the game, players 1 through n. We didn't label it for Nature. So Nature has no payoffs.

And then the final thing is remember when I talked about information sets, I said, for each player i. What I meant by that is each strategic player i. Nature also doesn't have information sets. Wherever Nature is, they just mechanically draw their probabilities. There's no sense of what Nature knows about other moves. We don't need to worry about that. So we have no information sets. Or an equivalent way of saying it is every info set is a singleton.

I guess this is convenient because I said the root must be a singleton information set. Well, if the root is a move by Nature, technically, I should say Nature has a Singleton information set if I want to be consistent with this definition. But that's nothing about interpretation. It's just we allow Nature to start the game. And then I guess, we often-- it can sometimes get a bit cumbersome to keep writing Nature. So often we just draw a circle to indicate Nature's nodes. So for strategic players, we have a filled-in circle. And for Nature, we have a hollow or empty circle.

Great. So now that we've talked about extensive form games, I'd like to move on to strategies. Unless there's any questions on extensive form games, we're going to move on now. And maybe I'll say this is kind of implicit, but Nature also has no strategies, which we're about to define.

So what is a strategy? Well, I want you to put yourself in the shoes of someone approaching this game. Before the game has started, you're contemplating how you're going to play in this game. And the strategy is the answer to that. The strategy is a complete contingent plan.

So it's a plan of action. You analyze it at the beginning. You're an outsider, or you're one of the players, but you're kind of evaluating and assessing the game. And you look at all the places where you might be called upon to move, and you plan ahead. You make a plan for what actions you're going to take at every node.

It's a complete plan. You have to specify what you're going to do at every possible contingency that arises. So I'm very often asked in section and office hours, do I really have to specify this contingency here? I've never been asked that question, and the answer has been no. It's always been yes.

So if you're unsure, do I really need to specify this as part of a strategy? Unless you're different than every other student I've had before, the answer is yes, you have to specify it. So when in doubt, specify it. It turns out that's going to make your life-- that kind of makes your life easier. But I think we'll see that as we go throughout the course. But I'll just say that maybe my answer is yes, you have to include it, everywhere, even when you think you don't need to.

And contingent is capturing the fact that you're going to learn things over the course of the game. So if, for instance, these two nodes are on different information sets, then player 2 is going to learn what player 1 does. And therefore, their plan is implicitly going to be contingent plan. It's not going to say, I'm going to go to the Celtics game, or I'm going to go to the Red Sox game. It might say something like, I'm going to go to the Celtics game if my friend goes to the Celtics game, and I'm going to go to the Red Sox game if my friend goes to the Red Sox game. And that's what's capturing the contingency in a strategy. So that's the intuitive idea. What is it formally? Formally, it specifies an action at every information set.

So often a good way to write a strategy is to put out some spaces for each of the information sets a player has. So let's say a player has four information sets. Maybe this is information set 1, information set 2, information set 3, and information set 4. And the way you specify a strategy is to specify an action here-- maybe I'll call it A1-- an action here, an action here, an action here.

So how many information sets I have are going to be determined by the structure of the extensive form game. But once I have the extensive form game written down, I can list out all the information sets for a given player. And then I can specify a strategy by specifying what action the player will take at each of these information sets. So the interpretation of this strategy would be if I find myself in information set 1, I will take action A1. If I find myself at information set I2, I will take action A2, and so on.

And of course, the action that I take at information set I1 must be a feasible action at that information set. If it was infeasible, then it wouldn't make sense. But here's a question. I said this action must be feasible at an information set. But an information set has many nodes. So how does that make sense? Isn't feasibility a property of a node, not an information set? Yeah?

AUDIENCE: One of the properties of information sets was that every node had to have the same feasible actions.

IAN BALL: Exactly. So because every node in information set has the same feasible actions, I can think of the feasible actions at an information set because it's the same whichever node in that information set I look at. And therefore, it's meaningful to say, I'm going to take action A1, which is feasible at this information set. It just means it's feasible at every single node in the information set that I might be at. And again, this is a contingency. We often say, if it-- we'll do this. But if it rains, we'll do this. That's exactly what this is capturing. We'll do this. But if it rains, we'll do this. And if it snows, we'll do this. And if it's sunny, we'll do this. So let's now take-- let's go over here.

So we've defined an extensive form game. We've defined a strategy. I now want to describe what a strategic form game is. That's actually really easy. But then what's harder is showing how we move from an extensive form game to an equivalent strategic form game. So let's first talk about a strategic form game.

And the idea is, well, look, we wrote down this really complicated extensive form game. But once we understand the concept of a strategy, we can really simplify this game a lot. What happens in this game? Each player chooses a strategy. They choose a complete contingent plan. And then based on those complete contingent plans, we can calculate what's going to happen in the game. So the strategic form representation is a way of reducing the game simply to its strategic components.

So what do we specify? We specify the players as before. We specify the strategies. Well, well call them-- we'll have sets. So we have S1 up to Sn. Big S1 is the set of strategies for player 1. So every element, every member of this set, is a particular strategy for player 1. We list all those strategies together and put them together in a set called Big S1. And we might use notation like little s1 is a particular strategy that lives inside this set of strategies.

And then it's often going to be helpful to have notation for profiles of strategies because if we want to understand what's going to happen in a game, it's not enough just to know player 1's strategy. We need to know, what is player 1 doing? What is their strategy? What is player 2's strategy? All the way up to what is player n's strategy?

So we're going to define a set, capital S is equal to S1 times up to Sn. So what is this? This is the set of all tuples or pairs, triples, however many we have, of strategies. So as an example, an element of this might be something like S equals S1 up to Sn. So this is the element symbol. So what I'm saying is S here is a strategy profile.

So this notation will come up throughout the year. So I think it's really important to make sure we understand it. Little s is a profile of strategies. It specifies a strategy for player 1, a strategy for player 2, all the way up to a strategy for player n. So sn is the strategy by player n in the particular strategy profile s. This strategy profile s lives in this set big S This is the collection of all strategy profiles in the game. Yeah?

AUDIENCE: So when we say that, we're saying it's every combination of every strategy any player could think of.

IAN BALL: Exactly right, yeah. So that's what this product notation is. We choose one strategy for player 1. We choose one strategy for player 2, all the way up to n. And any collection of those choices is going to give us the strategy profile. Yeah. And then the last thing is just payoffs.

And payoffs, we need a function ui from S to R for i equals 1 to n. Now, some people are tempted to put an i here because they see an i here. But that would be wrong. If we put an eye here, that would mean that player i's payoff only depends on player i's strategy. If that were the case, we wouldn't need to do game theory at all. That's just an individual decision problem.

What makes it interesting is that the payoff that I get as player i depends not only on what I do, but also on what everyone else does. And that's why there's no i here. And the payoff of player i is a function of the strategy profile. It says, for instance, if there's two players, if player 1 chooses strategy S1 and player 2 chooses strategy S2, what is going to be the payoff for player i from that combination of strategies, or in other words, from that strategy profile?

So strategic form, it's really simple, but you have to make sure you write down the set of strategies correctly, and you have to make sure you write down the payoffs correctly. So what we're now going to describe is how we can move when we're given an extensive form game, how we can reduce it to a strategic form game, reduce it to its strategic form representation. So let me go to a new board over here. Do we have one?

So what's our procedure for going from-- so it's not always the case. But often when we start by considering some strategic situation, we write down all the details in the extensive form to make sure we specify everything exactly right. And then when it's time to analyze the game, it's often more convenient to reduce this extensive form to this more compact and convenient strategic form. But if you try to jump right to the strategic form, you might make a mistake. So it's good to be really careful about how you set up the extensive form and then convert things to the strategic form.

So how do we do it? Well, there's two steps or two hard steps. One step is just look at who the players are. But that's pretty easy. Once we have an extensive form game, we see who all the players are, and the players are going to be the same. So that's easy. The hard thing, or the two hard steps, are that we have to define the strategy sets.

So we'll do an example in a second. But a strategy set would be something like a collection of all vectors that look like this. That would be an example of what the strategy set will be. But to define it correctly, you have to make sure you specify all the information sets correctly and all the possible actions that can be chosen at those information sets.

And then the second step is to compute the payoff functions ui from S to R. That is, once I've defined the strategy sets, I still have to say, if you give me a strategy profile, specifying a complete contingent plan for each of the players, what is the payoff that player i is going to get from that profile?

So why is this hard? It's hard because, from the extensive form, what we know is the payoff each player gets from a consequence, from a terminal node in the game. And now we have to compute what payoff they get from a strategy profile, which is not the same thing as a consequence. So how do we do that? Let me give you a maybe somewhat cryptic formula. And then we'll do an example to see how we do it.

But the key is that a strategy profile s, if we focus on a particular strategy profile s, that is going to induce in the extensive form game a lottery over terminal nodes. It's going to induce a lottery over consequences. So maybe I'll write z of s. And I'll show how that happens. But a strategy profile induces a lottery over consequences.

And once we can compute that lottery over consequences, we can compute each player's expected utility from that lottery over consequences, because remember, we've already defined utilities over consequences. And that expected utility is going to be the new definition of the utility function that we assign to this strategy profile. So we'll do an example. But the step is a strategy profile determines a lottery over consequences, which allows us to compute expected utilities. And those are going to be the payoffs in our strategic form game. So let's do an example. Let me go here. And we'll end with this example.

So we're going to think of the Boston game we did before. But it's a little artificial just to illustrate ideas that Nature is going to flip a coin to determine the order in which we move in this game. So first, Nature is going to flip a coin, say 1/2, 1/2, to give myself some more space.

And if the coin is, say, heads, then player 1 is going to move first, and their move is not going to be observed. So how do we represent this? Player 1 moves first. They choose Celtics or Red Sox. Their move is not observed. So that means player 2-- this is player 2. And we put a dotted line here to indicate player 2 does not observe the move by player 1. And then player 2 chooses Celtics or Red Sox. And you can see, it's going to get pretty messy to write all the payoffs, so for now, I'm not going to write the payoffs. But the payoffs will be the same as in the other game.

And if we do it the other way, let's say if nature flips tails, then player 2 is going to move first. But then their move is going to be observed by player 1 So now we have player 2 here. They're going to choose between Celtics and Red Sox. Now it's player 1's turn to move. We don't draw a dotted line here because player 1 observes what player 2 does. And then we have CR, CR.

So let's make sure we understand the game. Nature is going to flip a coin. These are the rules. If the coin is heads, player 1 decides first, then player 2 decides without having observed what player 1 does. If Nature flips tails, then player 2 goes first, and then player 1 goes after observing what player 2 did. Yes?

AUDIENCE: So here then, does player 2 know if they're going first?

IAN BALL: Great. So let's look at the information sets. So how many information sets does player 2 have in this game? I'll open it up to everyone. 2. We have information set here and an information set here. Remember, this is actually an information set. I just haven't drawn it. So once we write out all the information sets explicitly, we can always answer questions like this. So when I'm here, well, I know that I'm either at one of these two nodes. I know that I'm not here, which means I know the choice that Nature made. I know they flipped heads. But I don't know whether player one chose C or R.

AUDIENCE: But you could, theoretically, even though they're not at the same line?

IAN BALL: A different game would be I put all three of these in the same information set. That would be a different game that I could write down, but that's not the game I've written down here. Exactly. Yeah. Any other questions on this? So now, the hardest step is figuring out strategies.

So I think, especially when you're just getting used to this, whenever you're asked to write down strategies, the first question should be, how many information sets does each player have? Because often when people get strategies wrong, it's because they've gotten the information sets wrong. So let's look at player 1 and player 2. So first, how many information sets does player 1 have? Yeah?

AUDIENCE: Three.

IAN BALL: Three. Well, they have three nodes, 1, 2, and 3. And remember what I said is I'm not going to draw-- or I hope I said this. I'm not going to draw the singleton ones because it's just a little messy. So technically, it's like this. But I'm just not going to draw those. So if you don't see the nodes connected, that means each node is in its own information set.

So player 1 has three information sets. And each of these information sets is what we call a singleton. It has a single node in it. So player 1 has three info sets. And what about player 2? How many information sets does player 2 have? Two, right? Information set 1 and information set 2.

So why is it good to ask how many information sets are first? Because you can think of a strategy as a vector. And the number of places in the vector is going to be the number of information sets. So each information set is going to have a slot in this vector. And then we're going to fill in those slots.

So let's think of strategies for player 1. We're going to have three slots. And strategies for player 2, we're going to have two slots. Why slots? Because a strategy specifies something for each information set. So we're just going to specify a slot wherever we need to put something.

Now, the next thing is it's kind of helpful often to number the information sets. On a problem set, you could either explicitly number them, or maybe you could indicate how you're going to number it so that the TA understands. But it's important that you're clear about which of these slots each information set corresponds to. So maybe I'll label player 1's information sets I, 2I, and 3I. I didn't want to use 1, 2, and 3 because that is confusing with players. And then for player 2, I guess I'll do I and 2I.

So now, let's fill this in. We can just give an example. So we have three information sets. To specify a strategy, we just have to put Celtics or Red Sox here, Celtics or Red Sox here, and Celtics or Red Sox here. So how many strategies are we going to have in total for player 1? Eight strategies. And what about for player 2? How many strategies? Four, right?

So in general, it's a bit more complicated if you have more than two moves, but whenever you have two moves at every information set, the reason we get 8 is this is 2 to the 3. And the reason we get 4 is this is 2 to the 2 because we have two choices here. For each of those two choices, we have two choices here. And for each of those two choices, we have two more. So you can-- it's like a tree. It keeps doubling each time. And eventually, you'll get eight.

So that's a lot to write down. If we really want to specify this game, we have to specify utilities for every strategy profile. So we're not going to do it now. But let's just understand, if there's eight strategies for player 1 and four strategies for player 2, then all together, we have 32 strategy profiles, because for each of the eight strategies of player 1, we can combine it with the strategy for player 2.

And then for each of those profiles, we have to specify two numbers, the payoff for player 1 and the payoff for player 2. So altogether, we need to specify 64 numbers in this really, really small game. And I think this is illustrating why games like chess, no one's been able to even-- we can't even deal with a number of strategies in chess. It's many, many times larger than the number of atoms in the universe, like the atoms in the universe to the atoms of the universe, crazy, crazy numbers. And the reason for that is, well, you can see here, even in the simplest possible game, it's combinatorial. It's exponential.

So let's just do one example to see how we can calculate these payoffs. So let's say player 1 is going to choose-- let's say-- it doesn't really matter. I'll just choose it arbitrarily. Let's do that. And let's say player 2 is choosing RC. So what I'd like to do is I'm going to take this strategy by player 1 and this strategy by player 2, and I want to compute what the payoffs are going to be for each player. So I'm basically filling in two of those 64 numbers that I have to fill in if I wanted to fully specify the game.

So what's often helpful is-- let me start with player 1. And I'm going to go to each of player 1's information sets. And I'm going to highlight the edge that the player would choose at that information set under this strategy. So information set 1 is here. If player 1 finds himself at this contingency, this strategy specifies that that player chooses C. So I'm going to highlight this here. If player 1 finds himself at this contingency, he's going to choose R. And at this contingency, he's going to choose R as well.

Now let's go to player 2. At this contingency-- this is information set 1-- player 2 chooses R. So I'll fill this in with R here. And now, this one is trickier. At this information set, she chooses C. But now what do I fill in? Well what I have to do is I have to fill in C at both nodes. Why? Well, when player 2 gets here, player 2 can't observe which node she's at. She just always does C because that's what the strategy says.

So if it turned out she were at this node, this is what she would choose. If she were at this node, this is what she would choose. She doesn't know which one she's at. She just plays C at that information set. So now that we've filled this in, what we want to do is kind of trace through the graph to try to compute what the induced lottery over consequences is going to be.

So we start here. If Nature chooses heads, which happens with probability 1/2, we go here. Then player 1 chooses C. And then player 2 chooses C. So we're going to get this outcome with probability 1/2. If Nature chooses tails, then player 2 chooses R. And then player 1 chooses R. So we're going to get this outcome with probability 1/2.

So remember over here, I said a strategy profile for the players induces a lottery over consequences. That's exactly what happened over here. We computed a strategy profile. This is the particular strategy profile. What is the lottery it induces? The induced lottery is we get to this history with probability 1/2, which means we both go to the Celtics game after nature flipped the coin heads. And we get to this consequence also with probability 1/2. What is this consequence? Well, it's this full history. Nature chooses tales. Then player 2 chooses Red Sox. And then player 1 chooses Red Sox.

And now if we go back, we can compute the expected payoffs from this strategy profile by looking at 1/2 times the utilities here plus 1/2 times the utilities here. And that's going to give each player's expected payoff from this strategy profile. That gives us two numbers. And we only have 62 to go. Obviously, I'm not going to ask you to do things like that on problem sets. But that's, in principle, how it would work. Any questions on this? Yes?

AUDIENCE: Can I ask again, why are we introducing the concept of the Nature of flipping a coin? Is this a correct transition of the game, in the extensive form game, we were looking at before?

IAN BALL: So I would say, this is a new extensive form game. So I defined this new extensive form game to try to illustrate some of the concepts. So this is not the same as the extensive form game we did before. The reason it's important to introduce moves of Nature is that a lot of games in reality involve randomness.

So the way that we're going to capture randomness more broadly is going to be through nature. This is kind of a silly example, with nature flipping a coin. But in reality, it might be I set my price, the price at my gas station to be this. And then there's a war in the Middle East. And then the price of oil changes. And that's a choice of Nature.

So it's going to be more substantive things. I set my price, and then there's an antitrust case against my competitor, things like that I can't anticipate, and then I form beliefs over, are going to be represented by moves of Nature. But I want to be clear that this extensive form game is not the same as this extensive form game. These are two different games that illustrate different ideas. Any other questions? Yeah?

AUDIENCE: If you wanted to make it so that the players didn't know who was choosing first, would you just put all the nodes for each player into one information set, basically?

IAN BALL: Yeah, you could do that. And that would be legal in this game because we just have to check, are two properties satisfied? Is there a singleton that starts things? Yes. And does the player have the same set of feasible actions at every node in the information set? And that would still be true. I will warn you that there are some weird things you can do with information sets, what we-- so we've imposed a few conditions. There's a few things we haven't imposed.

So a natural assumption is that players don't forget what they previously knew. This is something called perfect recall. You might find it strange if a player takes a move and then later forgets what move she previously took. We haven't ruled that out explicitly. But every game we're going to study, maybe with the exception of the last day of class, will have this property of perfect recall. And without it, you can have some weird things.

On the other hand, is perfect recall really a good assumption? You probably don't remember a lot of things you did when you were a child. So maybe in some cases, imperfect recall is a good model. But that's something that is up to the modeler in this course. Any other questions? So let me stop there, and I'll see everyone on Thursday.