1 Introduction

When reasoning about formal mathematics it is very common to use the concept of a (co)representable functor. A (co)representable functor is a method of describing the set of morphisms into (out of) an object in a category, and how to compose these morphisms. Saying that an object is a (co)representation of a functor also uniquely specifies that object up to isomorphism and encodes some other useful information about the object. I will be more precise about this later.

One example of a corepresentation of a functor is the polynomial ring $R[X]$ over a commutative ring R. This is a corepresentation of the functor $S \mapsto \mathrm{Hom}(R, S) \times S$. This means that for any commutative ring S there is an equivalence of sets $R[X] \cong \mathrm{Hom}(R, S) \times S$. There is an evaluation map *eval* taking a morphism $\mathrm{Hom}(R, S)$ and an element of S to return a morphism $\mathrm{Hom}(R[X], S)$. This gives one direction of the isomorphism $R[X] \cong \mathrm{Hom}(R, S) \times S$. The other direction of this isomorphism is given by sending $f \in \mathrm{Hom}(R[X], S)$ to the pair $(f \circ C, f(X)$, where C is the canonical map $R \to R[X]$. The fact that this is an isomorphism encodes the fact that $eval(f, s)(X) = s$ and $eval(f, s)(C(r)) = f(r)$. It also encodes the fact that any two morphisms $f, g \in \mathrm{Hom}(R[X], S)$ are equal if $f(X) = g(X)$ and $f \circ C = g \circ C$. These functions and equalities give convenient ways of defining maps in and out of the polynomial ring and of proving they are equal. They also uniquely define the polynomial ring as an object in the category of commutative rings. These equalities are all "obvious" to most mathematicians, including mathematicians who know nothing about category theory, and most of the equalities that follow from the equalities above are of the kind that would not require a proof in a paper. They can be, however, difficult to prove in a proof assistant to somebody who does not know some techniques for proving equations like this in a proof assistant.

In general saying an object is a (co)representation of a partiular functor gives a method of defing morphisms into or out of this object and of proving equalities of these morphisms. It also characterises the object within the category, giving some notion of what the object "is". (I don't really know what this means but it is probably important for a bunch of reasons).

Knowing that an object in a category is a representation of a functor F is not very useful unless we also know something about the functor F. For example, F might be the product of functors, it might be a dependent product of functors, or it it could be a Hom-functor, or a composition of functors. We will define a syntax for writing down representable functors that covers most examples found in Lean.

The notion of representable functor also has applications to the problem of transferring proofs across isomorphisms. As an example, if M and N are monoids, and $M^\times$ are the units of the monoid M, the two groups $M^\times \times N^\times$ and $(M \times N)^\times$ are isomorphic. However in order to transfer properties about this it is important to know more about how this isomorphism behaves. Both of these groups have natural maps into the monoids M and N. Often, when we want to prove a property of the group $M^\times \times N^\times$ also holds for the group $(M \times N)^\times$, this property may be not just a property of the group structure, but a property of the group structure and the maps into M and N say. The isomorphism $M^\times \times N^\times \cong$

$(M \times N)^\times$ commutes with the maps into M and N, so in fact it is stronger than an isomorphism of groups, it is an isomorphism of groups with maps into M and N. This follows from the fact that these two groups are representations of the same functor. The group $M^\times \times N^\times$ is a representation of the functor $G \mapsto \text{Hom}_{Mon}(G, M) \times \text{Hom}_{Mon}(G, N)$, i.e. $\text{Hom}_{Grp}(G, M^\times \times N^\times) \cong \text{Hom}_{Mon}(G, M) \times \text{Hom}_{Mon}(G, N)$. Substituting $G = M^\times \times N^\times$, and applying this isomorphism to the identity map of $M^\times \times N^\times$, gives the canonical maps $M^\times \times N^\times \to M$ and $M^\times \times N^\times \to N$. Therefore, by understanding representable functors, we have a better way of understanding how to transfer proofs across isomorphisms that preserve more structure than just the group structure.

Reasoning about representable functors requires every example of a representable functor to be explicitly instantiated as such by the user. The

```
{ring} tactic in Lean would be useless without
also having a \verbatim{comm_semiring} typeclass where every example of a commutative
given as an instance of this typeclass. To be able to use the nice properties of repr
it is necessary to encode exactly what properties we want to use and to provide a con
the user to be able to prove instances of this, and should preferably not require the
much category theory in order to be able to use it.


\section{Definition of Representable Functor}

An object $X$ in a category $C$ is a representation of a functor $F : C^{op} \rightar

\begin{equation}
  \text{Hom}(-, X) \cong F
\end{equation}

Precisely what this means is that for any object $Y$ in $C$ there is an isomorphism $
that this isomorphism satisfies the following equations.

For any $Y$ and $Z$ in $C$, and any $f \in \text{Hom}(Z, Y)$ and $g$ in $\text{Hom}(Y
\begin{equation}
  \phi_Z(f ; g) = F(f)(\phi_Y(g))
\end{equation}

This can also be expressed as the commutativity of the following square

% https://tikzcd.yichuanshen.de/#N4Igdg9gJgpgziAXAbVABwnAlgFyxMJZABgBpiBdUkANwEMAbAVx
\begin{tikzcd}
  {\text{Hom}(Y, X)} \arrow[d, "f;"'] \arrow[r, "\phi_Y"] & F(Y) \arrow[d, "F(f)"] \\
  {\text{Hom}(Z, X)} \arrow[r, "\phi_Z"']                  & F(Z)
\end{tikzcd}


This above equation is equivalent to assuming naturality of $\phi^{-1}$ which is
For any $Y$ and $Z$ in $C$, and any $f \in \text{Hom}(Z, Y)$ and $y \in F(Y)$, we hav
```

```
\begin{equation}
  \phi_Z^{-1}(F(f)(y)) = f ; \phi_Y^{-1}(y)
\end{equation}

This can be expressed as the commutativity of the following square

% https://tikzcd.yichuanshen.de/#N4Igdg9gJgpgziAXAbVABwnAlgFyxMJZARgBoAGAXVJADcBDAGwF
\begin{tikzcd}
  F(Y) \arrow[r, "\phi^{-1}_Y"] \arrow[d, "F(f)"'] & {\text{Hom}(Y, X)} \arrow[d, "f;
  F(X) \arrow[r, "\phi^{-1}_Z"']                    & {\text{Hom}(Z, X)}
\end{tikzcd}

One useful characterisation of a representation of a functor is the following. $X$ is
whenever there is the following

\begin{itemize}
  \item An element $c$ of $F(X)$
  \item For every object $Y$ of $\mathcal{X}$, a map $r_Y : F(X) \to \mathcal{X}(Y, X
  \item For any object $Y$ of $\mathcal{X}$, and every element $y \in F(Y)$, $F(r_Y(y
  \item For any two morphisms $f, g : \mathcal{X}(Y, X)$, if $F(f)(c) = F(g)(c)$ then
\end{itemize}

The above functions are enough to define a natural isomorphism between $\text{Hom}(-,
With $f \in \text{Hom}(Y,X)$, we can define $\phi_Y(f)$ to be $F(f)(c)$ and
 for any $y \in F(Y)$, we can define $\phi_Y^{-1}(y)$ to be $r_Y(y)$.

A corepresentation of a functor $F : C \rightarrow Set$ is defined in a similar way.
An object $X$ is a corepresentation of $F$ if there is a natural isomorphism

\begin{equation}
  \text{Hom}(X, -) \cong F
\end{equation}

This can be characterised by the following data about $F$

\begin{itemize}
  \item An element $u$ of $F(X)$
  \item For every object $Y$ of $\mathcal{C}$, a map $e_Y : F(Y) \to \mathcal{C}(X, Y
  \item For any object $Y$ of $\mathcal{C}$, and every element $f : F(Y)$, $F(e_Y(f))
  \item For any two morphisms $f, g : \mathcal{C}(X, Y)$, if $F(f)(u) = F(g)(u)$ then
\end{itemize}


There are many examples of representations or corepresentations of functors in mathem

If $C$ is a category, and $X$ and $Y$ are objects of $C$, then the product, $X \times
of the functor $Hom(-, X) \times Hom(-, Y)$, meaning for any object $Z$ of $C$, the s
$\text{Hom}(Z, X \times Y)$ is isomorphic to the set $\text{Hom}(Z, X) \times \text{H
```

```
In the category of sets, the set with one element, $1$, is the terminal object. This r
of the constant functor $1 : Set \to Set$, i.e. the set of morphisms from any set $X$
one element.

In the category of sets, the empty set, $\emptyset$, is the initial object. This mean
of the constant functor $1 : Set \to Set$, i.e. the set of morphisms from $\emptyset$
exactly one element.

In the category of commutative rings, the polynomial ring $R[X]$ is a representation o
$\text{Hom}(R, -) \times \text{Forget}$, where $\text{Forget}$ is the forgetful
functor from commutative rings to sets. This means that for any commutative ring $S$ a
$R[X] \to S$ can be constructed by taking a ring homomorphism $R \to S$ and an elemen

In the category


\section{Examples of Equality Proofs Using UMP}

Using universal properties provides a convenient way of defining and proving equaliti
categories.

\section{Examples of Isomorphisms made easier with UMP}


\section{Inheritance of UMP}

\subsection{Grothendieck Construction, Categories of Elements and Pi Categories}

\subsection{Monoidal Closed Categories}

\subsection{Maybe Something about Adjoints Preserving (co)limits}

\subsection{Something About Parametricity}

\section{Comparison with UMP of Inductive Types}

\section{The Isomorphism Problem - Transferring across Isomorphisms}

There is a problem in formal proof about how to transfer proofs and structure across
```