

107: ~ について調べる

↑ up

- issued: 2020-06-07
- 分類: D (要調査)
- status: Open

概要

The language report では、unzip は次のように定義されている。この ~ はなんだったかな？

```
unzip      :: [(a,b)] -> ([a],[b])
unzip      = foldr (\(a,b) ~(as,bs) -> (a:as,b:bs)) ([],[])
```

調査ログ

2021-11-01 (Mon)

Haskell 2010 3.17.2 Informal Semantics of Pattern Matching^{*1} に、irrefutable pattern (不可反駁パターン) と refutable pattern (可反駁パターン) に関する記載がある。以下はそこに書かれている内容の一部を訳したもの。この記述のあとに、refutable / irrefutable パターンマッチングの例もある (のでみておこう)。

irrefutable pattern のマッチングは non-strict であり、マッチ対象の値が \perp であってもマッチする。refutable pattern のマッチングは strict であり、マッチ対象が \perp のときにはマッチは発散 (diverge) する。

irrefutable パターンは以下：

- 変数
- ワイルドカード
- $N \text{ apat}$ で、 N が newtype 定義される構築子であり、かつ apat が irrefutable
- var@apat で apat が irrefutable
- $\sim \text{apat}$

irrefutable pattern については、「Haskell 入門」の p.59 に「反駁不可能パターン ~」というそのものずばりのセクションがある。

まず、refutable な例：

unno@unno-FMVD70GN7G ~/work/bissues/107

^{*1} <https://www.haskell.org/onlinereport/haskell2010/haskellch3.html#x8-220003>

```

$ cat refutablepat.hs
main = print $ case undefined of (_, _) -> True
unno@unno-FMVD70GN7G ~/work/bissues/107
$ runhaskell refutablepat.hs
refutablepat.hs: Prelude.undefined
CallStack (from HasCallStack):
  error, called at libraries/base/GHC/Err.hs:78:14 in base:GHC.Err
  undefined, called at refutablepat.hs:1:16 in main:Main
unno@unno-FMVD70GN7G ~/work/bissues/107
$ ~/prj/bunny/compiler/bin/bunny testrun refutablepat.hs
/home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/refutablepat --xno-implicit-prelude /home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/refutablepat --xlibrary-path /home/unno/prj/bunny
error: Prelude.undefined

```

これは期待通りの動作といえる。

つぎに、irrefutable な例：

```

unno@unno-FMVD70GN7G ~/work/bissues/107
$ cat irrefutablepat.hs
main = print $ case undefined of ~(_, _) -> True
unno@unno-FMVD70GN7G ~/work/bissues/107
$ runhaskell irrefutablepat.hs
True
unno@unno-FMVD70GN7G ~/work/bissues/107
$ ~/prj/bunny/compiler/bin/bunny testrun irrefutablepat.hs
/home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/irrefutablepat --xno-implicit-prelude /home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/irrefutablepat --xlibrary-path /home/unno/prj/bunny
bunnyc: removeInfix :LazyPat (TupleExp [Just WildcardPat,Just WildcardPat])
CallStack (from HasCallStack):
  error, called at src/Rename.hs:133:21 in main:Rename
CallStack (from -prof):
  Rename.scanDecls.removeInfix (src/Rename.hs:(103,5)-(133,53))
  Rename.scanDecls.scanValueDecl2 (src/Rename.hs:(35,5)-(88,33))
  Rename.scanDecls (src/Rename.hs:(28,1)-(322,64))
  Rename.renExp (src/Rename.hs:(861,1)-(1024,66))
  Rename.renRhs (src/Rename.hs:(810,1)-(835,52))
  Rename.renDecls.renDecl (src/Rename.hs:(631,5)-(645,50))
  Rename.renDecls (src/Rename.hs:(583,1)-(645,50))
  Semant.renProg (src/Semant.hs:(21,1)-(52,11))

```

```
Main.doCompile(...) (app/Main.hs:83:7-70)
Main.doCompile (app/Main.hs:(77,1)-(114,33))
Main.main (app/Main.hs:(117,1)-(132,42))
testrun: failed to compile irrefutablepat.hs
```

こちらは、`~pat` のリネームが未対応でコンパイルがこけた。

つぎに、「Haskell 入門」にはない `irrefutable` の例で `var` パターンを試してみる：

```
$ cat irrefutable2.hs
main = print $ case undefined of x -> True
unno@unno-FMVD70GN7G ~/work/bissues/107
$ runhaskell irrefutable2.hs
True
unno@unno-FMVD70GN7G ~/work/bissues/107
$ ~/prj/bunny/compiler/bin/bunny testrun irrefutable2.hs
/home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/irrefutable2 --xno-implicit-prelude /home/unno/pr
/home/unno/prj/bunny/compiler/bin/bunnyc -d ./jout/irrefutable2 --xlibrary-path /home/unno/prj/bunny
True
```

これも期待通り。

今後は、まず、仕様にある `irrefutable` パターンのすべてのケース（上に箇条書きで書いてあるもの）について、`undefined` に対してもマッチすること、および、マッチしない値を与えたときにエラーすることを確認するテストプログラム群をまず書いてから継続調査しよう。

※ `irrefutable pattern` が `fail` する例

```
unno@unno-FMVD70GN7G ~/work/bissues/107
$ cat irrefutablepat_fail.hs
main = print $ case (1, 1) of ~(x, 0) -> x
unno@unno-FMVD70GN7G ~/work/bissues/107
$ runhaskell irrefutablepat_fail.hs
irrefutablepat_fail.hs: irrefutablepat_fail.hs:1:16-42: Non-exhaustive patterns in (x, 0)
```