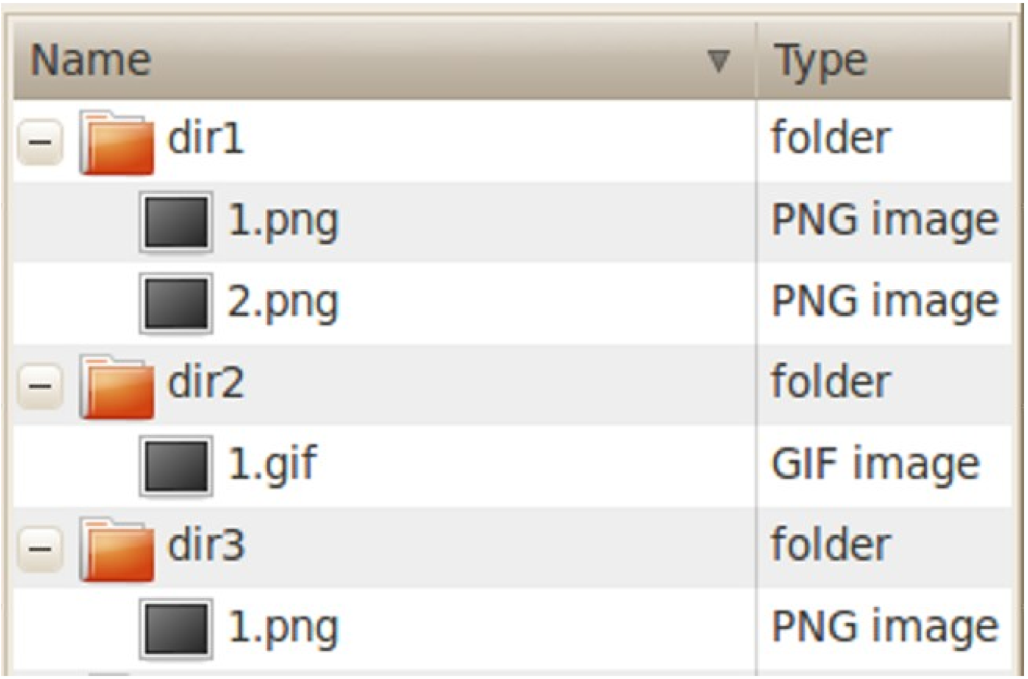
Perl Gotchas

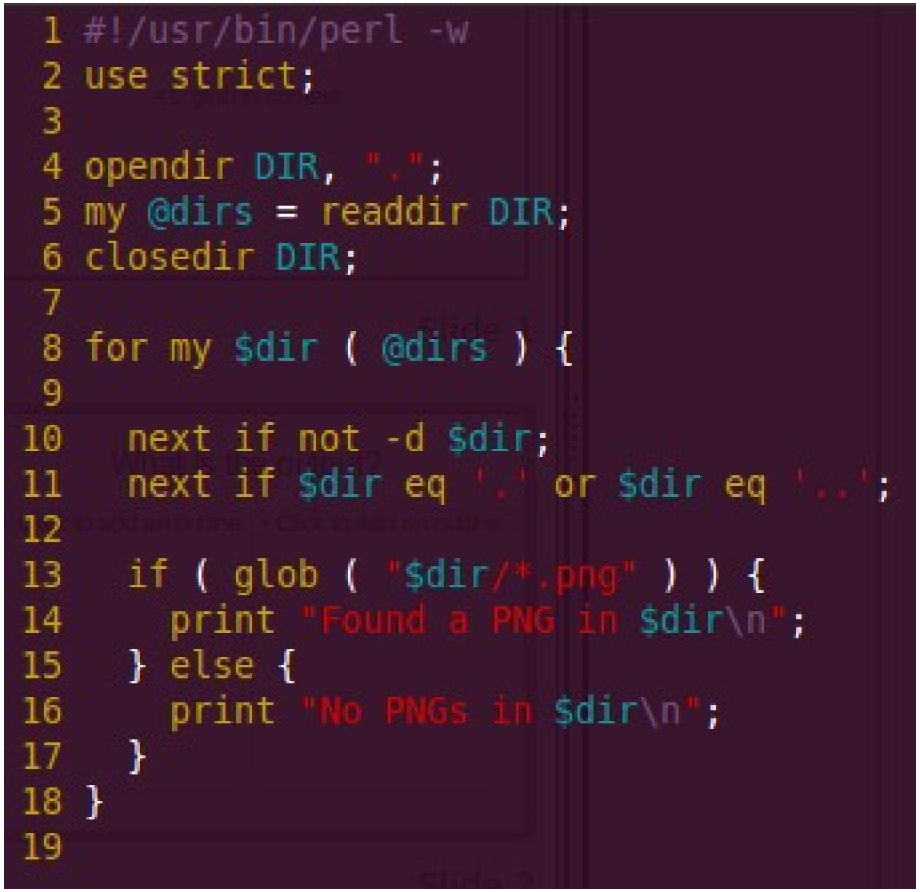
Gotcha #1: `glob` in context

Bryan Smith, Nov 29 2010

What is the output?

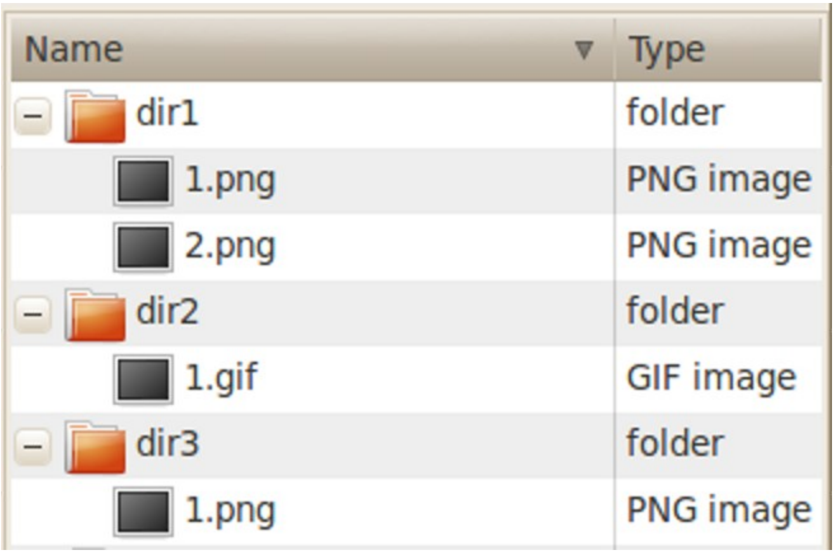


Name	Type
dir1	folder
1.png	PNG image
2.png	PNG image
dir2	folder
1.gif	GIF image
dir3	folder
1.png	PNG image

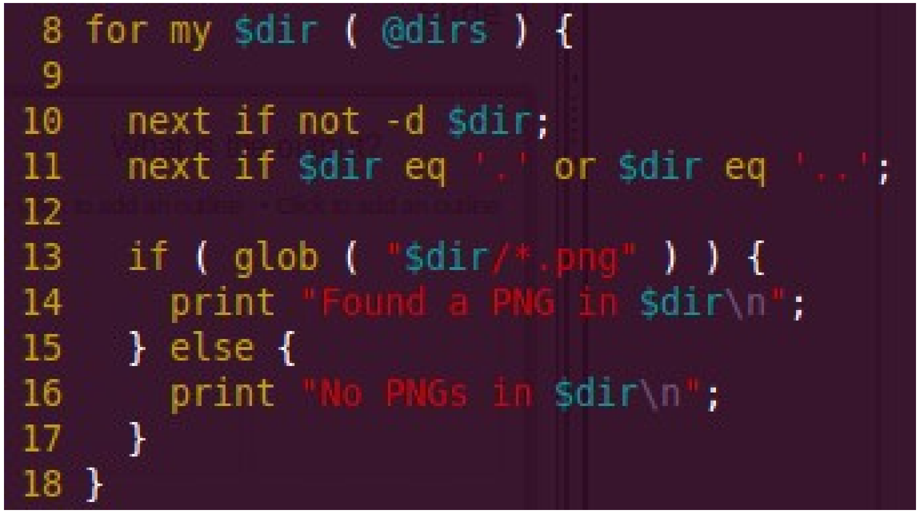


```perl
#!/usr/bin/perl -w
use strict;

opendir DIR, ".";
my @dirs = readdir DIR;
closedir DIR;

for my $dir ( @dirs ) {

  next if not -d $dir;
  next if $dir eq '.' or $dir eq '..';

  if ( glob ( "$dir/*.png" ) ) {
    print "Found a PNG in $dir\n";
  } else {
    print "No PNGs in $dir\n";
  }
}

```

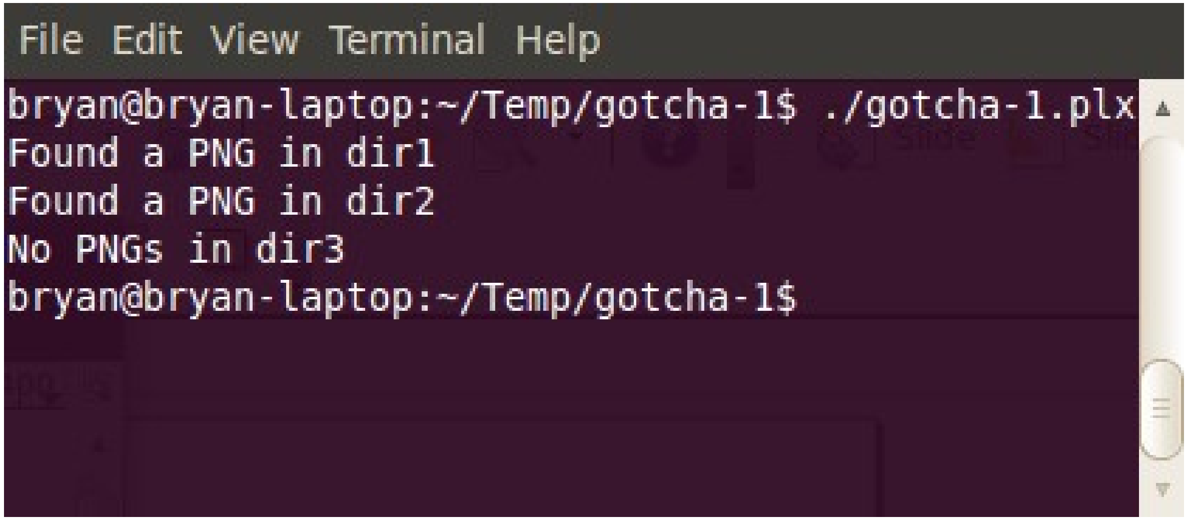
Uh-oh...



Name	Type
dir1	folder
1.png	PNG image
2.png	PNG image
dir2	folder
1.gif	GIF image
dir3	folder
1.png	PNG image



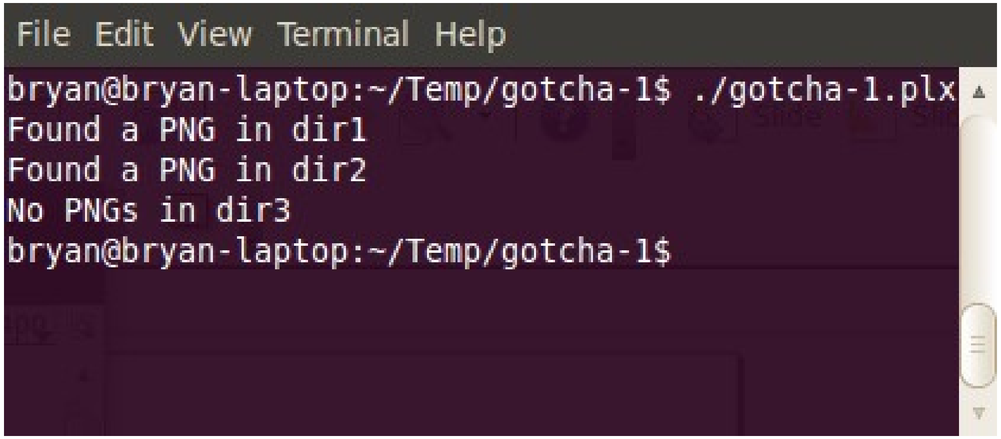
```
for my $dir ( @dirs ) {

  next if not -d $dir;
  next if $dir eq '.' or $dir eq '..';

  if ( glob ( "$dir/*.png" ) ) {
    print "Found a PNG in $dir\n";
  } else {
    print "No PNGs in $dir\n";
  }
}
```



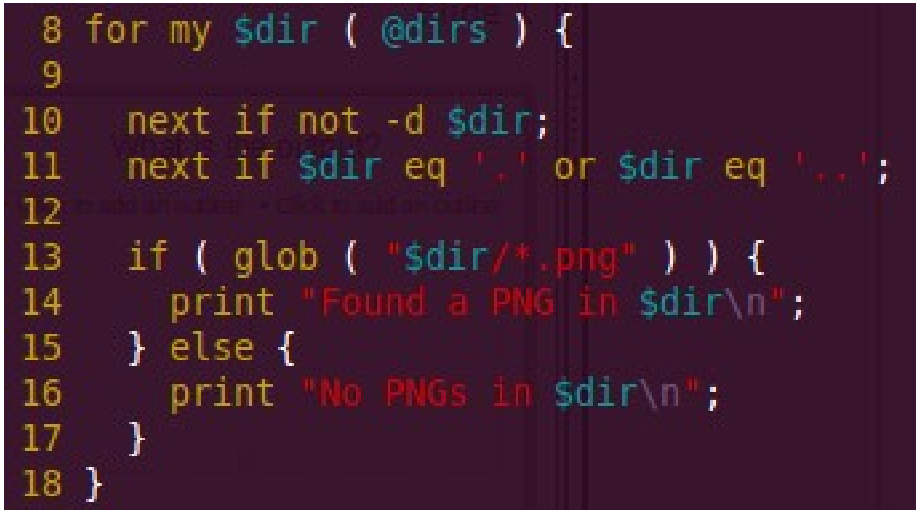
```
bryan@bryan-laptop:~/Temp/gotcha-1$ ./gotcha-1.plx
Found a PNG in dir1
Found a PNG in dir2
No PNGs in dir3
bryan@bryan-laptop:~/Temp/gotcha-1$
```

What happened?

- Expected to see PNGs in dir1 and dir3
- This isn’t a bug... this is glob in a scalar context!



```
bryan@bryan-laptop:~/Temp/gotcha-1$ ./gotcha-1.plx
Found a PNG in dir1
Found a PNG in dir2
No PNGs in dir3
bryan@bryan-laptop:~/Temp/gotcha-1$
```



```
for my $dir ( @dirs ) {

  next if not -d $dir;
  next if $dir eq '.' or $dir eq '..';

  if ( glob ( "$dir/*.png" ) ) {
    print "Found a PNG in $dir\n";
  } else {
    print "No PNGs in $dir\n";
  }
}
```

Scalar vs. list context

- Where an expression is evaluated determines its context
- E.g.,
  - `my $scalar = … # scalar context`
  - `my ( $var1, $var2 ) = … # list context`
  - `my @array = glob ( “$dir/*.png” ) # list context`
  - `if ( glob ( “$dir/*.png” ) ) # scalar context`
- See *Learning Perl* (Schwartz & Phoenix), §3.8 “Scalar and List Context”

If at first you don’t succeed, RTFM

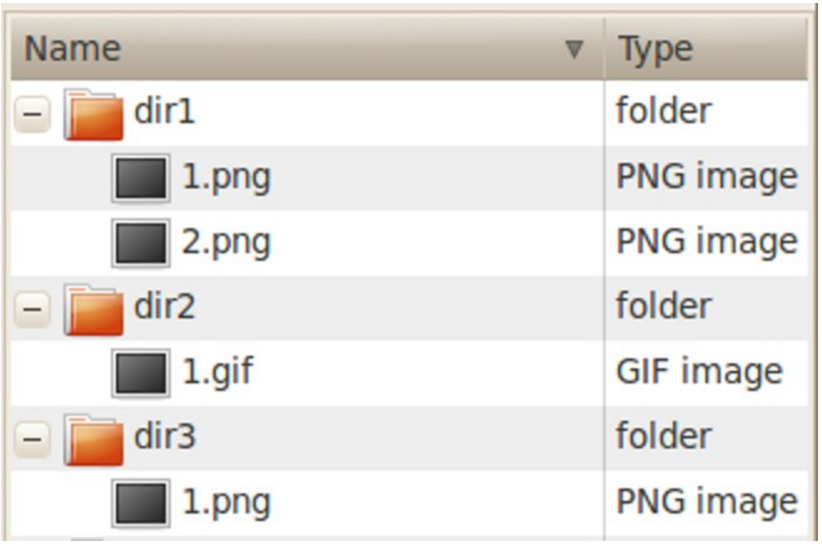
- Running `perldoc -f glob`:

```
In list context, returns a (possibly
empty) list of filename expansions on
the value of EXPR such as the standard
Unix shell /bin/csh would do. In scalar
context, glob iterates through such
filename expansions, returning undef
when the list is exhausted...
```

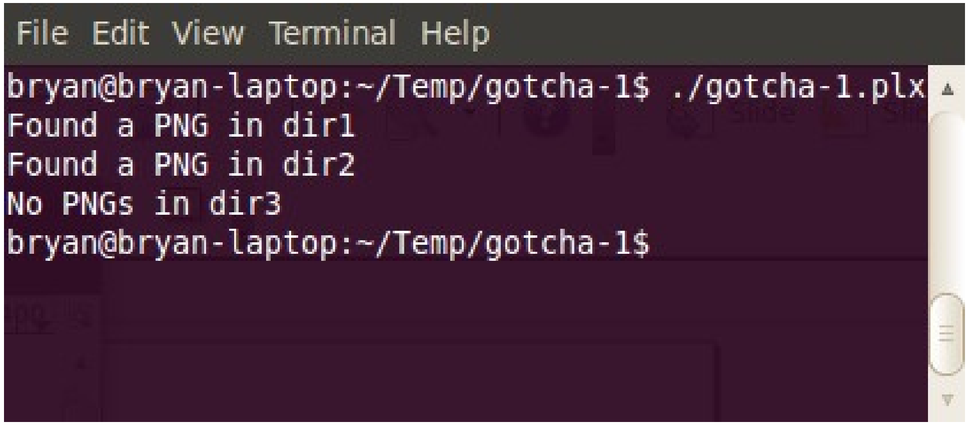
Ah-ha!

> “In scalar context, glob iterates through... returning undef when the list is exhausted...”

- So each iteration, glob ignored the directory parameter and continued iterating files in `dir1/`!



Name	Type
dir1	folder
1.png	PNG image
2.png	PNG image
dir2	folder
1.gif	GIF image
dir3	folder
1.png	PNG image



```
bryan@bryan-laptop:~/Temp/gotcha-1$ ./gotcha-1.plx
Found a PNG in dir1
Found a PNG in dir2
No PNGs in dir3
bryan@bryan-laptop:~/Temp/gotcha-1$
```

Conclusion

- Perl basic: Important to know the difference between list and scalar contexts.
- `glob` behavior not particularly intuitive in scalar context
- `` `perldoc -f <function>` `` is a great resource
  - If not seeing what you expect, look for documentation on your current context