Weave a semi-literate program into Text, Markdown, etc. format

```
sub MAIN($input-file,
         Bool :l(:$line-numbers)  = True;
         Str :f(:$format) is copy = 'markdown';
         Str :o(:$output-file);
    ) {
    my Str $extension;
    my Str @options;
    my Bool $no-output-file = False;

    $format .= trim;
    given $format {
        when  /:i markdown / {
            $format    = 'MarkDown2';
            $extension = 'md';
        };
        when  /:i [plain]? t[e]?xt / {
            $format    = 'Text';
            $extension = 'txt';
        }
        when  /:i html / {
            $format    = 'HTML2';
            $extension = 'html';
        }

        when /:i pdf / {
            $format = 'PDF';
            $extension = '.pdf';
            @options = "--save-as=$output-file" if $output-file;
            $no-output-file = True;
        }

        default {
            $extension = $format;
        }

    }

    my Str $f = "Pod::To::$format";
    try require ::($f);
    if ::($f) ~~ Failure {
        die "$format is not a supported output format"
    }

    my Str $woven = weave($input-file, :$format, :$line-numbers);

    my ($pod-file, $fh) = tempfile(suffix =>  '.p6');

    $pod-file.IO.spurt: $woven;

    my $output-file-handle = $output-file              ??
                                    open(:w, $output-file) !!
                                    $*OUT
                                unless $no-output-file;

    run $*EXECUTABLE,
        "--doc=$format",
        $pod-file,
        @options,
        :out($output-file-handle);

}


```