`qolistings.sty` Package

Peter Brookes Chambers

November 6, 2020

Contents

1 Introduction

This package defines and sets up several styles for the `listings` package, as well as a few useful supplimentary commands. This package requires four other packages: `listings`, `mdframed`, `xcolor`, and `forarray`. This package does not provide much functionality, and mostly exists for convenience. Each style uses specifically named colours, namely `ForegroundColour`, `ForegroundColour_3`, `ForegroundColour_5`, `BackgroundColour`, `Accent1`, and `Accent2`. As such, loading the `officecolours` package (or changing the scheme used by the `officecolours` package) will change the colours used in these styles. Thus, code listings should always fit in with the rest of a document.

However, the `officecolours` package is not necessary for the `qolistings` package, and so some default colours are defined and used **if and only if** no colour already has the names listed above. To this end, a new command is also defined; `\colorprovide`. This acts identically to the `\colorlet` command provided in the `xcolor` package, except that it will not overwrite an existing color. For example, `\colorlet{blue}{red}` will overwrite the colour `blue` with the value of `red`. However, `\colorprovide{blue}{red}` will change nothing, since a colour with the name `blue` already exists. In other words, `\colorprovide` is to `colorlet` as `\providecolor` is to `\definecolor`.

2 Styles

2.1 main

The first style defined is `main`, on which the subsequent styles are mostly based.

```
IMPORT numpy as np

DEF getA(order, scheme = "central"):
    # Returns the $a$ array for a given order. By default, it returns the values of $a$ for the
    # central difference scheme, but this can be changed by using
    #
    #       scheme = "forward"
    # or
    #       scheme = "backward"
    FROM numpy.linalg IMPORT solve

    IF order % 2 == 1:
        RAISE ValueError("\n\n\tArgument \"order\" must be divisible by 2.\n\n")

    aSize = INT(order / 2)
    # Let $M a = P$

    IF scheme.lower() == "central":
        M = np.zeros([aSize, aSize])

        FOR i IN RANGE(aSize):

            FOR j IN RANGE(aSize):
                M[i, j] = (j + 1) ** (2 * i + 1)

        P = np.zeros([aSize])
        P[0] = 1 / 2
        a = solve(M, P)
    ELSE:
        M = np.zeros([order, order])

        FOR i IN RANGE(order):

```

```
            FOR j IN RANGE(order):
                M[i, j] = (j + 1) ** (i + 1)

        P = np.zeros([order])
        P[0] = 1
        a = solve(M, P)
        PRINT(r"$\int \limits _0 ^{\infty} \text{e}^{-x^2} \text{d} x$")
    RETURN a
```

```
IMPORT numpy as np

DEF getA(order, scheme = "central"):
    # Returns the a array for a given order. By default, it returns the values of a for the
    # central difference scheme, but this can be changed by using
    #
    #       scheme = "forward"
    # or
    #       scheme = "backward"
    FROM numpy.linalg IMPORT solve

    IF order % 2 == 1:
        RAISE ValueError("\n\n\tArgument \"order\" must be divisible by 2.\n\n")

    aSize = INT(order / 2)
    # Let Ma = P

    IF scheme.lower() == "central":
        M = np.zeros([aSize, aSize])

        FOR i IN RANGE(aSize):

            FOR j IN RANGE(aSize):
                M[i, j] = (j + 1) ** (2 * i + 1)

        P = np.zeros([aSize])
        P[0] = 1 / 2
        a = solve(M, P)
    ELSE:
        M = np.zeros([order, order])

        FOR i IN RANGE(order):

            FOR j IN RANGE(order):
                M[i, j] = (j + 1) ** (i + 1)

        P = np.zeros([order])
        P[0] = 1
        a = solve(M, P)
        PRINT(r"$\int_0^{\infty} \text{e}^{-x^2}\text{d}x$")
    RETURN a
```

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur

auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Nam dui ligula, fringilla a, euismod sodales, sollicitudin vel, wisi. Morbi auctor lorem non justo. Nam lacus libero, pretium at, lobortis vitae, ultricies et, tellus. Donec aliquet, tortor sed accumsan bibendum, erat ligula aliquet magna, vitae ornare odio metus a mi. Morbi ac orci et nisl hendrerit mollis. Suspendisse ut massa. Cras nec ante. Pellentesque a nulla. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Aliquam tincidunt urna. Nulla ullamcorper vestibulum turpis. Pellentesque cursus luctus mauris.