Solution Stack-based Programming

Instructions:

Solutions should be placed in a separate folder with the name "Assignment02".
Answers to the Exercise 1 submit in **one** .txt file, clearly marked which answer corresponds to which question.
Answers to the Exercises 2 and 3 submit in **two** .ps files named, accordingly, "catalan.ps" and "stars.ps".

Exercise 1 (1.5 points)

Answer the following questions about Postscript:

- What kinds of stacks does PostScript manage and what are their roles?

  Answer:

  1. *Operand stack - the most important, since it's used for all computations*
  2. *Dictionary stack - holds sets of local variables to be used by procedures we define*
  3. *Execution stack - hidden from the user; used to manage running procedures*
  4. *Graphics state stack - makes easy for a user to work in different coordinate systems*

- What is the way of defining a procedure in the PostScript program?

  Answer:

  Procedures are defined by binding names to executable objects, in a way "key value def".

- What are the roles of a dictionary and the dictionary stack in the PostScript program execution?

  Answer:

  By defining our own dictionary, and pushing it to the dictionary stack, we make sure that the names we use do not conflict with any other similar names used by other procedures. The dictionary stack therefore serves the same purpose as the run-time stack in most programming languages.

Exercise 2 (2 points)

Define a procedure in PostScript that will calculate and print the first `n` Catalan numbers, where `n` is an argument on the stack. Catalan numbers are calculated based on the formula $C_n = \frac{(2n)!}{(n+1)!n!}$. The call to the procedure should look like `n catalan`. The output should be similar to the one shown in Figure 1 for `n = 17`. Please use the provided template which contains the skeleton of the code, as it will make it easier for you (and us) to check your solution. Try to define sub-procedures whenever it makes sense.

Answer:

Catalan numbers - solution.

C (n = 0) = 1.0
C (n = 1) = 1.0
C (n = 2) = 2.0
C (n = 3) = 5.0
C (n = 4) = 14.0
C (n = 5) = 42.0
C (n = 6) = 132.0
C (n = 7) = 429.0
C (n = 8) = 1430.0
C (n = 9) = 4862.0
C (n = 10) = 16796.0
C (n = 11) = 58786.0
C (n = 12) = 208012.0
C (n = 13) = 742900.0
C (n = 14) = 2.67444e+06
C (n = 15) = 9.69485e+06
C (n = 16) = 3.53577e+07
C (n = 17) = 1.29645e+08

Figure 1: Catalan numbers

Exercise 3 (2.5 points)

Define a procedure in PostScript that will draw five-point stars given the following arguments on the stack:

`x` x coordinate of the center of stars

`y` y coordinate of the center of stars

`radius` distance from the center of stars to the points of the largest star

`n` number of stars

The procedure should draw `n` stars with the same center, where each star is smaller than the previous one. The decrease in the distance between the center of a star and the star's points should be by $\frac{\texttt{radius}}{\texttt{n}}$. The largest star should be filled with the black color. The each following star should be filled with the brighter nuance of the grey color. The nuance of the grey color of the `i`-th star should be $\frac{\texttt{i}}{\texttt{n}}$. Since we consider the largest star to have the index of `0`, its nuance of the grey color is $\frac{\texttt{0}}{\texttt{n}}$, that is black.

The output should be similar to the one shown in Figure 2, that is the result of the execution

```
300 400 200 10 draw
```

Please use the provided template which contains this code, as it will make it easier for you (and us) to check your solution. Try to define sub-procedures whenever it makes sense.

Answer:

Star - solution.

Figure 2: Stars