

Solution Stack-based Programming

- Exercises are given every week on the PL page of the SCG website (<http://scg.unibe.ch/teaching/pl>)
- Solutions to each assignment must be sent to **mohammadreza.hazhirpasand@inf.unibe.ch**
- The solutions of the assignments are to be delivered before every Thursday at 5 PM. Solutions handed in later than the specified time will not be accepted. In case of serious reasons send an e-mail to **mohammadreza.hazhirpasand@inf.unibe.ch**

Exercise 1 (4 points)

- What kinds of stacks does PostScript manage and what are their roles? (1 pts)

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? please also define a procedure to calculate the following formula and print the result on the screen : $((x + y) / 2) * 2$ (2 pts)

Answer:

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

Solution: /str 20 string def
320 550 moveto
/ADDFIVE { add 2 div 2 mul } def
9 9 ADDFIVE str cvs show

- Define a procedure to print 10 random numbers (using loops) and each number must be printed in a new line. *hint: “rand” produces random number* (1 pts)

sample output:
684570285
1502883016
252193898
...

Solution: /newLine {
currentpoint exch pop
FS 2 add sub
LM exch moveto
} def

```
/loopit {  
  320 650 moveto  
  10  
  {  
    rand str cvs show  
    newLine  
  } repeat  
} def  
loopit
```

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