

Activity 5 – Introduction to Proof

Algorithms

- (1) Here's a bit of sage code that defines a function that implements the division algorithm.

```
def division_alg(n, d):  
    """  
    Determines q and r such that  $n=qd+r$  and  $0 \leq r < d$   
    """  
    r=n  
    q=0  
    while r>=d:  
        r=r-d  
        q=q+1  
    return q,r
```

Notice that sage – which is closely related to python – uses indentation to group statements together into blocks. For instance, there are two lines inside the loop created by the ‘`while r>=d:`’ line, which are indented more than those around them. This makes the structure of the code visually obvious. As another example, notice that all of the lines following `def division_alg(n, d):` are tabbed in. They are all part of the definition of the function.

Type these lines into a cell in CoCalc. (Remember that the enter key provides line breaks only – when you’re finished typing hit shift-enter to execute the code.) BTW, the line between triple quotes is the documentation string for the function if you want to

save some typing you can leave it out. While you're typing notice how the interface takes care of the indentation for you.

When you finally hit shift-enter it seems that nothing has happened, but internally, sage now has a new function! Try out your newly created function in a new cell by typing something like

```
division_alg(23,7)
```

(but pick your own inputs.)

- (2) Put print statements into the previous function so that it creates a trace table while it is running.
- (3) There is a somewhat obvious way to break this function – use $d = 0$ as an input. Make some changes to handle that situation. (Look for the ‘If’ sample code under ‘Control’ in CoCalc.)
- (4) Could you modify the function above so that it dealt properly with signed inputs (i.e. positive and negative values of n and d)?
- (5) Now that you've gained some familiarity, try implementing the Euclidean algorithm. Note that sage already contains a function called ‘gcd’ so use another name. You can test your function against the internal one.
- (6) Notice that the internally defined ‘gcd’ function is agnostic about which input is larger (‘gcd(25,15)’ and ‘gcd(15,25)’ both work and return 5.) Does your function behave this way? If not, can you fix it?