

Introduction to Programming

Matrix Multiplication

The following program will add two “square arrays.” Note the change from the code we worked on in class.

```
# add 2 square matrices
def add(a,b):
    lena=len(a)
    x=[lena*[0] for i in range(lena)]
    for i in range(lena):
        for j in range(lena):
            x[i][j]=a[i][j]+b[i][j]
    return x

x=[[1,1],[1,1]]
y=[[1,1],[3,4]]
c=add(x,y)
print(c)
```

Write a function `mult(a,b)` that will “multiply” the two square matrices `a` and `b` as we discussed in class.

If we write `c=mult(a,b)` then `c[i][j]` will be the dot product of row `i` of matrix `a` with column `j` of matrix `b`.

Recall that if we have 2 lists `a` and `b` of the same size, then we can define `dot` as follows:

```
z=[1,2,3]
v=[4,5,6]

def dot(a,b):
    lena=len(a)
    g=0
    for i in range(lena):
        g=g+a[i]*b[i]
    return g

print(dot(z,v))
```

Problem

1. Ask your user for an integer `n`.
2. Create two `nXn` matrices (two dimensional lists) filled with random integers. Call one `A` and the other `B`.
3. Print `A` and `B`.
4. Multiply (by using the matrix multiplication algorithm) `A` and `B` to get matrix `C`.
5. Print matrix `C`.
6. Go back to step 1 and repeat until the user enters “done”.

Test out your function on matrices of various sizes. The values in the matrices should be generated by using `randint()`. I would suggest using small random numbers so that it will be easier for you to check.