1 Administrative Topics

- We go over the homework.
- We take the quiz.
2 Project 4 Advice

- Pseudo-randomly generated numbers: In this project, we need pseudo-randomly generated numbers. Every time we want another number, we must run the appropriate function (e.g. `random.random()`). Numbers are chosen from different distributions, depending on which function we call. To generate numbers that are centered around a particular number use `random.gauss()`. To generate random integers from a uniform distribution, use `random.randint`. To generate floats from a uniform distribution, use `random.random()` or `random.uniform(a,b).

- Here is the layout of `penguins.py`.

```python
def initPopulation():
    # ...
    # return a new population list

def simulateYear(pop, elNinoProb, stdRho, elNinoRho, probFemale, maxCapacity):
    # ...
    # return a new population list

def runSimulation(N, initPopSize, probFemale, elNinoProb, stdRho, elNinoRho, maxCapacity, minViable):
    # Loop over N years,
    # Call simulateYear, saving the return value in an
    # appropriate variable
    # Print the population size for each year
    # (or write it to a file)

def computeCPD(simulationResults, N):
    # ...
    # return a new list

def main(argv):
    # Store the command line arguments in variables
    # so we can use them when we call runSimulation
    # Loop over the number of simulations you need to do
    # Call runSimulation, append its results to a list (results)

    # Call CEPD, passing in results, saving the return value of
    # CEPD in a variable
    # Output the variable
```

2