GOALS

- **As a biology instructor, I need a cross-platform program (app?) my students can use to**
  a) Learn about the major forces of evolution;
  b) Learn to design experiments;
  c) Learn to interpret graphs and analyze data;
  d) Solidify their knowledge of Mendelian genetics
  e) Export data for more sophisticated analyses (and import scenario?)
  It should be open–source in a cross-platform language

BASIC GENETIC & EVOLUTIONARY CONCEPTS

- **A Species** can be conceived of as a metapopulation, a set of more–or–less isolated subpopulations that are capable of reproducing within & amongst themselves and which are isolated from populations of other species. EVOLVE simulates a metapopulation with one or more subpopulations.

A Population consists of Organisms

- Each population has 1 Gene that affects 1 or more traits (e.g., pigments, proteins, etc.)
  Genes may have 2 or 3 Alleles (more in real world) symbolized by A, B, C
  Alleles have
  Mutation Rates (A–>B, B–>A, A–>C, C–>A, B–>C, C–>B)
  Patterns of inheritance, see below

Ea. Organism has 2 copies of each gene

- Thus, organisms may have one of 3 (or 6) Genotypes.
  Symbolized AA, AB, BB; AC, BC, CC
  Organisms may be Homozygotes (2 copies of same allele) or Heterozygotes (2 different alleles)
  Organisms have Phenotypes: observable traits (e.g., pigments, proteins, etc.) produced by genotype.

Phenotypes of interest in EVOLVE include:

- Survival rate (avg. % young surviving to adulthood)
  Reproductive rate (avg. # young / adult)
  Absolute Fitness (SurvR8 * ReprR8) Values > 1 result in increasing population size
  Relative Fitness (AbsFit/MaxAbsFit) Values 0–1
  Migration rates (% individuals moving to another population)
  Mating preferences (QUERY: model or not? Adds significant complexity.)

Patterns of inheritance are determined by interactions between alleles and include

- Dominant–Recessive: heterozygote has same phenotype as dominant homozygote, recessive phenotype is homozygous recessive.
  E.g., if AA = 5, AB = 5, BB = 10, A is dominant, B is recessive
- Incomplete Dominance: heterozygote is in between phenotypes of homozygotes.
  E.g., If AA = 5, AB = 8, BB = 10, alleles are incompletely dominant.
- Codominance: heterozygote displays phenotypes of both homozygotes.
  E.g., AA = Protein1, BB = Protein 2, AB = Proteins 1 & 2.
- Heterosis / Heterozygote superiority: heterozygote phenotype is more extreme than either homozygote.
  E.g., AA = 5, AB = 13, BB =8
- Underdominance: heterozygote phenotype is less extreme than either homozygote.
  E.g., AA = 5, AB = 3, BB = 10

Hardy–Weinberg Concept

- A population can be modeled as a gene pool made up of all of the alleles in a population.
  Gene Pools have
  Allele Frequencies (p = freq. A, q = freq. B, r = freq. C), by definition, p + q = 1 or p + q + r = 1
  Genotype frequencies
- Mating consists of randomly sampling pairs of alleles without replacement from the gene pool to produce the young of the next generation.
- Allele frequencies will not change and genotype frequencies will be
  $p^2 + 2pq + q^2$ (2–alleles) or
  $p^2 + 2pq + 2pr + 2qr + q^2 + r^2$ (3–alleles)
  This is the null case of no evolution.
- However the above assumes non–overlapping generations and that:
  1. Alleles don't change, or that mutation rates from one allele to another are equal
  2. No differences in allele/genotype survival or reproduction, i.e., there is no natural selection
  3. The population is closed, no differential emigration/immigration of alleles/genotypes, i.e., there is not Gene flow:
  4. The population is infinitely large, i.e., there is no Genetic Drift
  5. Random mating; there are no differences in probability of genotypes mating w/ ea. other, i.e., no sexual selection
  Violation of any of the 5 assumptions leads to changes in allele/genotype frequencies = evolution.

Modeled Organisms

- Diploid (have 2 copies of each gene)
- Have non–overlapping generations; think of insects where adults mate, give birth, lay eggs, set seeds, etc., then die before the young mature.
- Hermaphroditic, facultatively self–fertilizing (if cannot find a mate, will fertilize themselves).

- Life cycle: adults mate, produce zygotes (eggs/seed/young), die. A % of young may emigrate; migration is random between populations. Immigrants are added to resident young. Young undergo selection, those that survive are adults of next generation.

PARAMETERS & CONTROLS

Menus

File

- **New Notebook**
- **Open ...**
- **Save As...**
- **Export...** {basic & advanced lists}
- **Page Setup...**
- **Print...**

Edit

- **Cut** ⌘X
- **Copy** ⌘C
- **Paste** ⌘V

Run

- **Continue, Current Parameters** ⌘^C
- **Continue, Different Parameters** ⌘D
- **New Run, Current Parameters** ⌘N
- **New Run, Change Parameters** ⌘^N

Problem Description: Title: [_____Text box_____] {Title, required}

- **Question: [_____]** {Question}
- **Experimental Design: [_____]** {ExpDesign}
- **Results: [_____] {**ExpResults}
- **Discussion, Further Questions: [_____]** {Discussn}

Experiment Configuration

- **Seed:** {Autofill, but allow user input} **[___Int#___]** (Reuse a number to repeat run)

Alleles ◉ 2 ◎ 3 {hide C allele, genotypes in all displays, parameter lists, results if 2 is ON}

- **Alleles:** (use default or enter your choice of letters & names)
  [A|*Letter*] [*Name*] **[B|*Letter*] [*Name*]** **[C|*Letter*] [*Name*]**

  Genotypes' Phenotypes: {change genotype letters to match user input above. Limit length of Phtype?}
  AA: [Phtype] ***AB: [Phtype]*** ***BB: [Phtype]***
  AC: [Phtype] ***BC: [Phtype]***
  CC: [Phtype]

Initial Population

◉◎ Allele Frequencies (establish equilibrium population)

- **[Int#] Initial Population Size (10–10,000)**
- **Frequency of {Show user–entered allele letters, names; don't show C allele if Alleles 2 is ON}**
  [dec#] ***A Name*** **[dec#]** ***B Name {Calculated}***
  [dec#] ***A Name*** **[dec#]** ***B Name*** **[dec#]** ***C Name {Calculated}***

◎◉ Genotypes (establish non–equilibrium population) {Show user–entered allele letters & Phtypes; don't show C types if *Alleles 2* is ON}

- **[Int#] No.** ***AA Phtype*** **[Int#] No.** ***AB Phtype*** **[Int#] No.** ***BB Phtype***
  [Int#] No. ***AC Phtype*** **[Int#] No.** ***BC Phtype***
  [Int#] No. ***CC Phtype***

Active Evolutionary Forces

☑ Population Size (AKA Genetic Drift) {Use fixed checked box to show is always in operation.}

- ◉ **Fixed [___#___]** ◎ **Variable{if OFF, don't display disclose ▶}**

◎ Fixed ◉ Variable

- **[___#___] Carrying Capacity** (1–10,000)
- **[___#___] Post–Crash Size** (1–9,000)

▼ ◎◉ **Selection** {if OFF, don't display disclose ▶. Show user–entered allele letters; don't show C allele if Alleles 2 is ON. Show user's Phtype if there's room.}

▼ **Input: ◉ Survival, Reproductive Rates ◎ Absolute Fitnesses**

- **Survival Rates** (Avg. %)
  [dec%] *AA* [dec%] AB [dec%] *BB*
  [dec%] *AC* [dec%] *BC*
  [dec%] *CC*
  Reproductive Rates (Avg. # Young, 0–10)
  [dec#] *AA* [dec#] *AB* [dec#] *BB*
  [dec#]*AC* [dec#] *BC*
  [dec#] *CC*
  {Once entered, display calculated values below. Not boldfaced to indicated user can't change.}
  Calculated Absolute Fitness
  [dec#] *AA* [dec#] *AB* [_#_] *BB*
  [dec#] *AC* [_dec#_] *BC*
  [dec#] *CC*
  Calculated Relative Fitnesses
  [dec#] AA [dec#] AB [dec#] BB
  [dec#] *AC* [dec#] *BC*
  [dec#] *CC*

▼ **Input: ◎ Survival, Reproductive Rates ◉ Absolute Fitnesses**

- **Absolute Fitness** (0–10) {show calculated:}
  [dec#] *AA* [dec#] *AB* [_#_] *BB*
  [dec#] *AC* [_#_] *BC*
  [dec#] *CC*
  Calculated Relative Fitnesses {Not boldfaced to indicated user can't change.}
  [dec#] *AA* [dec#] *AB* [dec#] *BB*
  [dec#] *AC* [dec#] *BC*
  [dec#] *CC*

▼ ◎◉ **Mutation** {if OFF, don't display disclose ▶. Use user's allele letters.}

- **Rates** **From–>To** **From–>To** **{can we allow users to enter both decimal & e–notation? Use user's allele letters}**
  [Dec#] *A–>B* **[Dec#] *B–>A***
  [Dec#] *A–>C* **[Dec#] *B–>C***
  [Dec#] *C–>A* **[Dec#] *C–>B***

▼ ◎◉ **Migration (AKA Gene Flow)** {if OFF, don't display disclose ▶}

▼ ◉ **All Genotypes Equal ◎ Genotypes Unequal**

- **[Dec#]% Moving Between Populations**

▼ ◎ **All Genotypes Equal ◉ Genotypes Unequal** {Show user–entered allele letters; don't show C allele if Alleles 2 is ON. Show user's Phtype if there's room.}

- **% Moving Between Populations**
  [Dec#] *AA Phtype* **[Dec#] *AB Phtype*** **[Dec#] *BB Phtype***
  [Dec#] *AC Phtype* **[Dec#] *BC Phtype***
  [Dec#] *CC Phtype*

▼ ◎◉ **Non–Random Mating** (AKA Sexual Selection) {if OFF, don't display disclose ▶}

- ◉ **Non–Random Mating (AKA Sexual Selection)**

  ***AA* Preference for** {Show user–entered allele letters. Arrange as triangular matrix? Room for Phtype names?}
  [dec#]% *AA* [dec#]% *BB* [dec#]% *AB* [dec#]% *AC* [dec#]% *BC* [dec#]% *CC*
  AB Preference for
  [dec#]% *AA* [dec#]% *BB* [dec#]% *AB* [dec#]% *AC* [dec#]% *BC* [dec#]% *CC*
  BB Preference for
  [dec#]% *AA* [dec#]% *BB* [dec#]% *AB* [dec#]% *AC* [dec#]% *BC* [dec#]% *CC*
  AC Preference for
  [dec#]% AA [dec#]% BB [dec#]% AB [dec#]% AC [dec#]% BC [dec#]% CC
  BC Preference for
  [dec#]% AA [dec#]% BB [dec#]% AB [dec#]% AC [dec#]% BC [dec#]% CC
  CC Preference for
  [dec#]% AA [dec#]% BB [dec#]% AB [dec#]% AC [dec#]% BC [dec#]% CC

▼ **This will require some thoughts as to how to enter data. e.g.:**

▼ Could default all to no–preference values, allow user to change, then autocalc remaining values as entry progresses:

- E.g.: initial values w/ 2 alleles, display AA pref as
  [33.3..]% AA [33.3..]% AB [33.3..]% BB
  then user changes BB preference to 50%, display
  [25]% AA [25]% AB [50]% BB
  then user changes AA preference to 0%, display
  [0]% AA [33.3..]% AB [66.6..]% BB

  OR Could show all blank values w/ a [dec#]% remaining box at the end, e.g.:
  [0]% AA [0]% AB [0]% BB [100.0]% left to enter
  User changes BB to 50%, display
  [0]% AA [0]% AB [50]% BB [50.0]% left to enter
  User changes AB to 25%, display
  [0]% AA [25]% AB [50]% BB [25.0]% left to enter

▼ Are the above too complicated? Could also establish fixed scenarios, e.g.,

- 4 levels of inbreeding (e.g., 75%, 50%, 25%, 5% preference for like phenotypes)
  4 levels of outbreeding …

- Gets complicated w/ 3 alleles. Is there a better way of conceptualizing this?

- **Number of Generations: [_Int#_]**
- **Number of Populations: [_Int#_]**

▼ **<Query> How to handle**

- Sequential exp'ts (params change over time, e.g., population bottleneck)
- Parallel exp'ts (vary a parameter in several populations, e.g., genotype fitness changes from 0.1–0.9 by 0.1 or by selected values: 0.05, 0.10, 0.30, 0.50, 0.70, 0.9, 0.95)
- Notebook w/ list of exp'ts
- Instructor customization to hide some parameters, setup initial parameter values

▼ **OUTPUT PARAMETERS & CONTROLS** {very rough}

- **How to handle Ternary plots?** Over time = 4D?
- **{<QUERY> List of possible variables, drag to X–Axis, Y–Axis, Z–Axis}**

▼ **2D Graphs**

▼ **X Axis**

- ◉ **Time** ◎ **Freq. (allele)**

▼ **Y Axis**

- ◉ **Pop. Size** ◉ **Allele Freq** ◉ **Genotype Freq.**
  ◎ **Change** ◎ **Change**
  ◎ **Heterozygosity** ◎ **Mean Fitness** ◎**Genetic Variance**

▼ **3D Rotating Graphs**

▼ **X Axis**

- ◉ **Time** ◎ **Freq. (allele)**
  etc.

▼ **Y Axis**

- ◉ **Pop. Size** ◉ **Freq. Allele (list)** ◉ **Freq. Genotype (list)**
  ◎◉ **Heterozygosity** ◎◉ **Mean Fitness** ◎◉ **Genetic Variance**
  etc.

▼ **Z Axis**

- ◎◉ **Mean Fitness** ◎◉ **Genetic Variance**
  ◎◉ **Histogram: Metapopulation Frequencies**
  ◎◉ **Smoothed: Metapopulation Frequencies**
  etc.

- **Print**
- **Export**
- **Notebook**

▼ **CALCULATIONS**

▼ **Notation & Definitions** The notation below is a combination of traditional notations and my own 'pseudo notation'

- **Allele Frequencies:** (range 0.0–1.0):
  p Frequency of the A allele, p(#) = frequency in generation #, etc
  q Frequency of the B allele
  r Frequency of the C allele
  Note: *p* & *q* are the traditional notations for frequency of 2 alleles; I've added *r* for the 3rd.
  Suggest we use *FreqA, FreqB, FreqC* in coding.

With all of these we could use parentheses to indicate generation, e.g. *FreqA(1)* is frequency of A allele in generation 1
Genotype Frequencies: *FreqAA, FreqBB, FreqAB, FreqCC, FreqAC, FreqBC*
Survival rates: (range 0.0–1.0):
SurvR8AA, SurvR8AB, … SurvR8BC
Reproductive rates: (range 0.0–10.0):
RepR8AA, RepR8AB … RepR8BC
Absolute Fitnesses: (range 0.0–10.0): *W,* or *ω,* is traditional notation for 'Fitness' (sometimes absolute & sometimes relative), with Selection coefficient (*s*) defined as *1 - W*
WAbsAA, WAbsAB, … WAbsBC
Relative Fitnesses: (range 0.0–1.0):
WRelAA, WRelAB … WRelBC
Total population size: (range 0–10,000)
NumPop
Genotype numbers: *NumAA, NumBB, NumAB, NumCC, NumAC, NumBC*
Mutation rates: from *A* to *B* and vice versa {QUERY: use en–dash as separator?}
MuR8A–B), MuR8B–A … MuR8C–B
Migration rates: (0.0–0.95)

MigR8AA, MigR8BB … MigR8BC

Mating rates: {QUERY: use en–dash as separator?}

M8R8AA–AA, M8R8AA–BB … M8R8BC–BC

- **Order of genotypes:** Suggest convention of alphabetical A & B homozygotes 1st, then heterozygote, then CC homozygote & heterozygotes

  Is there a more logical/useful convention? Note use of en–dash instead of hyphe/minus sign.

  - **Genotypes(6):** AA, BB, AB, CC, AC, BC
  - **Mating combinations (21):**

    AA–AA
    AA–BB BB–BB
    AA–AB BB–AB AB–AB
    AA–CC BB–CC CC–CC CC–AB
    AA–AC BB–AC CC–AC AB–AC AC–AC
    AA–BC BB–BC CC–BC AB–BC AC–BC BC–BC

- **For each population, we will need to store the following basic data** (at least; you'll come up with more)**:**
  - **Array of Adults: 6 genotypes x number of generations**
  - **Array of Mates: 21 genotype–genotype combinations x # of generations**
  - **Array of Young: 6 genotypes x number of generations**
  - **Array of Migrating Young: 6 genotypes x number of generations x 2 directions, MigsIn & MigsOut**

    MigsIn would be added to Young array, MigsOut would be subtracted & distributed among the MigsIn of other populations.

- **Patterns of inheritance** = the relationship of genotypes to phenotypes.
  - **There are 5 common patterns of inheritance:**
    - **Dominance–recessiveness = heterozygote resembles one homozygote**
      - E.g.: if

	AA	AB	BB	
WRel	0.5	1.0	1.0	then A & B are incompletely dominant

    - **Incomplete dominance = heterozygote is intermediate to both homozygotes**
      - E.g.: if

	AA	AB	BB	
WRel	0.5	0.75	1.0	then A & B are incompletely dominant

    - **Over–dominance/Heterosis/Heterozygote advantage = heterozygotes are superior to both homozygotes**
      - E.g.: if

	AA	AB	BB	
WRel	0.5	0.75	1.0	then A & B are over–dominant = show heterosis/heterozygote advantage

    - **Under–dominance/Heterozygote disadvantage = heterozygotes are inferior to both homozygotes**
      - e.g.: if

	AA	AB	BB	
WRel	0.5	0.25	1.0	then A & B are under–dominant = show heterozygote disadvantage

    - **Co-dominance = both alleles produce their phenotype in heterozygotes**

      Not relevant w/respect to this simulation b/c will collapse into one of above

      - e.g.: if

	AA	AB	BB	
SurvR8	0.5	0.5	0.3	then A is dominant to B w/respect to high survival
RepR8	3.0	4.0	4.0	then B is dominant to A w/respect to high reproduction
WAbs	1.5	2.0	1.2	Together, they exhibit co–dominance, i.e., heterozygotes reflect both, but collapses to heterozygote advantage w/respect to fitness
WRel	0.75	1.0	0.6	

Formulae

- **Allele & Genotype frequencies**
  - **Allele Frequencies on input**
    - **2–Allele Case:**
      - *p + q = 1*

        Given *FreqA, FreqB = 1 - FreqA*
    - **3–Allele Case:**
      - *p + q + r = 1*

        Given *FreqA, FreqB*, then *FreqC = 1 - (FreqA + FreqB)*
  - **Expected input genotype freq.s** (Hardy–Weinberg equilibrium)
    - **2–Allele Case:**
      - *p^2 + 2pq + q^2 = 1*

        Genotype Freqs.:

        FreqAA = FreqA^2

FreqBB = FreqB^2
*FreqAB = 2*FreqA*FreqB*

3–Allele Case:

- *p^2 + q^2 + r^2+ 2pq + 2pr + 2qr = 1*
  Genotype freq.s:
  FreqAA = FreqA^2
  FreqBB = FreqB^2
  *FreqAB = 2*FreqA*FreqB*
  FreqCC = FreqC^2
  *FreqAC = 2*FreqA*FreqC*
  *FreqBC = 2*FreqB*FreqC*

Allele freq.s given genotype numbers in generation *gen*

2–Allele Case:

- Given *NumAA(gen) + NumAB(gen) + NumBB(gen) = NumPop(gen)*
  *FreqA(gen) = (2 * NumAA(gen) + NumAB(gen)) / (2 * NumPop(gen))*
  *FreqB(gen) = (2 * NumBB(gen) + NumAB(gen)) / (2 * NumPop(gen))*

3–Allele Case:

- Given *NumAA(gen) + NumBB(gen) + NumAB(gen) + NumCC(gen) + NumAC(gen) + NumBC(gen) = NumPop(gen)*
  *FreqA(gen) = (2 * NumAA(gen) + NumAB(gen) + NumAC(gen)) / (2 * NumPop(gen))*
  *FreqB(gen) = (2 * NumBB(gen) + NumAB(gen) + NumBC(gen)) / (2 * NumPop(gen))*
  *FreqC(gen) = (2 * NumCC(gen) + NumAC(gen) + NumBC(gen)) / (2 * NumPop(gen))*

Change in Allele & Genotype Frequencies

By Hardy–Weinberg population concept, allele & genotype frequencies should not change. Thus the frequency in any generation should = the frequency in the previous generation. We can measure rate of evolution by subtracting its frequency in generation x from its frequency in generation x+1.

Change in Allele Freq.

Observed change in allele freq. from generation *gen to gen+1*.

- dFreqA(*gen*) = FreqA(*gen +1*) - FreqA(*gen*)
- dFreqB(*gen*) = FreqB(*gen +1*) - FreqB(*gen*)
- dFreqC(*gen*) = FreqC(*gen +1*) - FreqC(*gen*)

Change in Genotype Freq.

- dFreqAA(*gen*) = FreqAA(*gen +1*) - FreqAA(*gen*)
- :
- dFreqBC(*gen*) = FreqBC(*gen* +1) - FreqBC(*gen*)

Fitnesses

W or ω is a traditional notation for 'Fitness' (which may be either absolute or relative), with Selection coefficient, *s*, defined as s = *1 - W*

Absolute Fitnesses given survival & reproductive rates

- ***Absolute fitness = Survival rate * Reproductive rate***
- ***WAbsAA = SurvR8AA * RepR8AA***
  WAbsBB = SurvR8BB * RepR8BB
  :
  WAbsBC = SurvR8BC * RepR8BC

Relative Fitnesses given absolute fitnesses

- ***Relative fitness = genotype's absolute fitness / maximum of absolute fitnesses***
- ***WRelAA = WAbsAA / Maximum(WAbsAA, WAbsBB, WAbsAB, WAbsCC, WAbsAC, WAbsBC)***
  :
  WRelBC = WAbsBC / Maximum(WAbsAA, WAbsBB, WAbsCC, WAbsAB, WAbsAC, WAbsBC)

Mean Fitness of Population in generation *gen*

Mean Fitness of a population = average fitness of the whole population = fitness of each genotype multiplied by its frequency. The traditional notation is W–bar.

- $$\overline{W} = \sum_{i,j} f_{ij} W_{ij}$$
- ***MeanAbsFitPop(gen) = (FreqAA(gen)*WAbsAA) + (FreqBB(gen)*WAbsBB) + (FreqAB(gen)*WAbsAB) + (FreqCC(gen)*WAbsCC) + (FreqAC(gen)*WAbsAC) + (FreqBC(gen)*WAbsBC)***

Genetic Variance in fitness in generation *gen*

Population's variance in fitness = difference between each genotype's fitness and the population's mean fitness, squared, and multiplied by the genotype's frequency, & summed over all genotypes. Importance: Fisher's Fundamental Theorem states that rate of increase in mean fitness of a population is equal to its genetic variance in fitness.

- $$\mathrm{Var}_G(W) = \sum_{i,j} f_{ij} (W_{ij} - \overline{W})^2$$
- **GenVarFitPop(gen) = ((FreqAA(gen) * (WRelAA - MeanAbsFitPop(gen))^2) + (FreqBB(gen) * (WRelBB - MeanAbsFitPop(gen))^2) + (FreqAB(gen) * (WRelAB - MeanAbsFitPop(gen))^2) + (FreqCC(gen) * (WRelCC -**

MeanAbsFitPop(gen))^2) + (FreqAC(gen) * (WRelAC - MeanAbsFitPop(gen))^2) + (FreqBC(gen) * (WRelBC - MeanAbsFitPop(gen))^2)

Heterozygosity

Expected heterozygosity

Expected heterozygosity is one minus the summed squared frequency of each allele. If *m* is the number of alleles, and *i* is the allele frequency of the ith allele:

- $$H_e = 1 - \sum_{i=1}^{m} (f_i)^2$$
- **HeterozExp(gen) = 1 - (FreqA(gen)^2 + (FreqB(gen)^2 + (FreqC(gen)^2)**

Observed

Natural selection can push populations out of H-W equilibrium, so the expected heterozygosity may not match the observed heterozygosity. The deviation id usually small if mating is random — but we allow for non–random mating.

- **HeterozObs(gen) = FreqAB(gen) + FreqAC(gen) + FreqBC(gen)**

Possible future: Add Wright's F stats

https://en.wikipedia.org/wiki/F-statistics

F_{IT} is the inbreeding coefficient of an individual (I) relative to the total (T) population,

F_{IS} is the inbreeding coefficient of an individual (I) relative to the subpopulation (S)

F_{ST} is the effect of subpopulations (S) compared to the total population (T)

Can also be used in determining effective population size (not useful in monoecious pop?), amount of fixation due to drift …
Most of sources I've seen in cursory search deal only with 2 alleles, formulae for 3 alleles will require research.

- Given ability to have a metapopulation with migration & non–random mating, these might be worthwhile additions. I'm not familiar with these at presentt & will have to investigate them.

Calculating Offspring from mating combinations

Determining ratios of offspring genotypes

I've listed a table that shows the ratios of offspring genotypes expected from the 21 different possible mating combinations. Hard–coding each would be a bit tedious (but perhaps fast?). I suspect one of you can code a Punnett Square object to do it efficiently (outlined below the tables). I'm not sure which approach would be the most efficient.

The order in this table differs from our alphabetical list, but it may suggest a way to more efficiently code the genotypes of the zygotes/fertilized eggs.

- Table of offspring resulting from particular combinations of mated genotypes

	Offspring Ratio				
Mating Comb.	Homoz = Homoz	Homoz ≠ Homoz	Homoz – Heteroz	Heteroz = Heteroz	Heteroz ≠ Heteroz
AA–AA	1AA				
BB–BB	1BB				
CC–CC	1CC				
AA–BB		1AB			
AA–CC		1AC			
BB–CC		1BC			
AA–AB			1AA:1AB		
BB–AB			1BB:1AB		
CC–AB			1AC:1BC		
AA–AC			1AA:1AC		
BB–AC			1AB:1BC		
CC–AC			1CC:1AC		
AA–BC			1AB:1AC		
BB–BC			1BB:1BC		
CC–BC			1CC:1BC		
AB–AB				1AA:1BB:2AB	
AC–AC				1AA:1CC:2AC	
BC–BC				1BB:1CC:2BC	
AB–AC					1AA:1AB:1AC:1BC
AB–BC					1BB:1AB:1AC:1BC
AC–BC					1AB:1CC:1AC:1BC

Punnet Square

You've probably encountered this in HS Biology, but I've outlined the approach below. It's usually displayed as a 3x3 matrix with the alleles of one parent listed in the top row & the other parent's down the left column. The offspring genotypes go into the remaining 2x2 cells, but can be done by inspection.

Given a pair of mated genotypes, e.g.,

- AB–BC, List each pairwise allele combination: AB, AC, BB, BC, ratio is 1:1:1:1
- AA–BC, offspring are AB, AC, AB, AC, ratio is 2AB:2AC or 1:1

Determining number of young.

Once the ratios of genotypes are determined, the number of young of each genotype can be determined by adding the number of young for each parent. E.g.:

- **Given the RepR8s AA = 4, BC = 3,**
  AA–BC would produce 4 + 3 = 7 offspring, randomly to AB & AC genotypes in the Young array.

Mutation

Mutations may occur at rates of 10^{-6}–10^{-8} per cell. Let's assume mutations happen in zygotes/young before they migrate

- **Calculation**
  Multiply the number of young of each genotype by a Gaussian no. = the mutation rate (SD?) for each allele to the other alleles, round the result