

# *Game of Mendel:*

## *A song of blonds and browns*

Relazione terza unità didattica

---

*METHODS IN COMPUTER SCIENCE  
EDUCATION: ANALYSIS*

Andrea Rosati  
1595834

Federica Spini  
1588482

## Revision History

<b>Revision</b>	<b>Date</b>	<b>Author(s)</b>	<b>Description</b>
1.0	28.06.19	AR, FS	Prima stesura documento

## Indice

<b>1</b>	<b>Introduzione</b>	<b>3</b>
1.1	Le leggi di Mendel . . . . .	4
<b>2</b>	<b>Net-Logo</b>	<b>6</b>
2.1	Limiti di Netlogo . . . . .	6
2.1.1	Documentazione di NetLogo . . . . .	7
<b>3</b>	<b>Target e requisiti della la lezione</b>	<b>8</b>
3.1	Prerequisiti didattici interdisciplinari . . . . .	8
3.2	Prerequisiti didattici informatici . . . . .	8
3.3	Obiettivi didattici interdisciplinari . . . . .	8
3.4	Obiettivi didattici informatici . . . . .	9
<b>4</b>	<b>Come svolgere l'esperienza</b>	<b>10</b>
4.1	Sviluppare GOM in classe . . . . .	10
4.2	Task avanzati . . . . .	12
4.3	Discussione in aula . . . . .	14
4.4	La relazione . . . . .	14
4.5	Confronto . . . . .	16
4.6	Valutazione . . . . .	16
<b>5</b>	<b>Altre possibilità per riadattare la lezione</b>	<b>18</b>

## Introduzione

*Game of Mendel: A song of blonds and browns* (abbreviata in **GOM**) è un'attività interdisciplinare tra la biologia e l'informatica, pensata per studenti del quarto o quinto superiore. In questa learning unit i ragazzi utilizzando l'ambiente di sviluppo di [NetLogo](#) per implementare una simulazione di agenti basata sulle leggi della genetica formulate da [Gregor Mendel](#). Successivamente, basandosi da questa esperienza di laboratorio, gli studenti dovranno scrivere una relazione in cui descriveranno e interpreteranno i fenomeni che potranno osservare durante la simulazione da loro programmata.

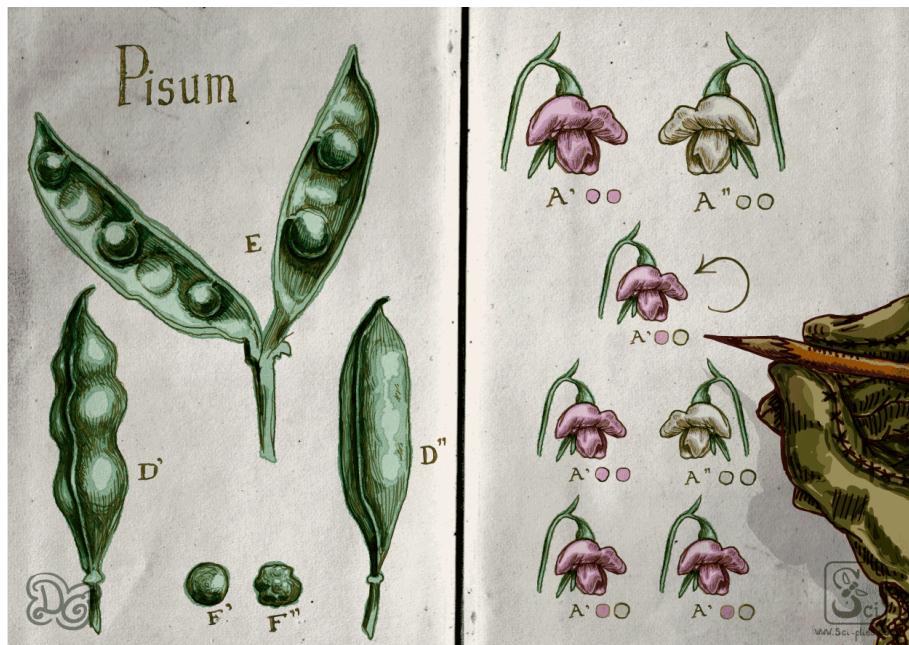


Figura 1: Gregor Mendel (1822-1884)

Ci è sembrato un lavoro interessante da proporre, come alternativa ai classici problemi genetici che prevedono quesiti su un gruppo ristretto di individui. In questo modo sarà possibile osservare il comportamento biologico causato dalla genetica mendeliana su una scala più vasta, in modo da vedere le leggi genetiche come qualcosa che regola anche la popolazione mondiale. Anche consolidare l'apprendimento di NetLogo con un progetto simile potrebbe essere utile, perché gli studenti potrebbero utilizzarlo per altri tipi di simulazioni, in progetti di altre materie, per illustrare un fenomeno in altre ricerche o nella tesina di maturità.

## 1.1 Le leggi di Mendel

**Gregor Johann Mendel** è stato un biologo, matematico e monaco agostiniano ceco di lingua tedesca, considerato il precursore della moderna genetica per le sue osservazioni sui caratteri ereditari. Mendel condusse esperimenti di genetica su piante di pisello odoroso in un orto del monastero di Brno. La sua ricerca consisteva nello scegliere i caratteri della nuova generazione di piselli controllando l'impollinazione e quindi scegliendo quali piante far accoppiare. Una volta raccolto i dati, Mendel ha potuto analizzarli utilizzando un approccio puramente matematico: questo studio gli ha permesso di poter smentire alcune convinzioni dell'epoca e formulare le sue tre leggi.



Prima degli studi di Mendel La teoria più in voga ai tempi era quella della **mescolanza** che affermava che i genitori danno un uguale contributo genetico alla prole e che in questa i fattori ereditari si mescolano. Con i suoi studi Mendel confermò il primo dogma ma confutò il secondo, poiché osservò che i caratteri dei piselli possono essere suddivisi in due categorie, dominanti e recessivi e fra di loro non c'è mescolanza.

Un figlio eredità da ciascun genitore un **allele** per ogni gene, se questi sono uguali il figlio sarà **omozigote** per quel gene e ovviamente avrà quel tratto, nel caso invece che il gene sia **eterozigote** e quindi formato da due alleli diversi il tratto che si manifesterà sarà quello dell'allele dominante.

Traits	Shape of Seeds	Color of Seeds	Color of Pods	Shape of Pods	Plant Height	Position of Flowers	Flower Color
Dominant trait	Round	Yellow	Green	Full	Tall	At leaf junctions	Purple
Recessive trait	Wrinkled	Green	Yellow	Flat, constricted	short	At tips of branches	White

Figura 2: I tratti presi in esame da Mendel

In un individuo dunque, per ogni caratteristica, è possibile parlare del **genotipo** come l'insieme delle caratteristiche genetiche che riguardano quello specifico tratto, sia del **fenotipo** come la manifestazione fisica del tratto. Ad ogni genotipo corrisponde un fenotipo, ma non è vero il contrario: un individuo che presenta un fenotipo dominante potrebbe avere un genotipo sia omozigote che eterozigote. Mendel notò in seguito che i tratti recessivi che non comparivano nella prima generazione di figli potevano ritornare nella seconda. Questo succede perché un figlio può ereditare dai propri genitori solo gli alleli recessivi e quindi può manifestare il tratto recessivo nonostante i genitori manifestino entrambi il tratto dominante perché entrambi hanno un genotipo eterozigote. Dai suoi studi Mendel formulò queste tre leggi genetiche:

### Leggi di Mendel

- Legge della dominanza (o legge dell'omogeneità di fenotipo): gli individui nati dall'incrocio tra due individui omozigoti che differiscono per una coppia allelica, avranno il fenotipo dato dall'allele dominante. Con significato più ampio rispetto al lavoro di Mendel, può essere enunciata come legge dell'uniformità degli ibridi di prima generazione
- Legge della segregazione: durante la generazione della prole, gli alleli associati a uno stesso gene si separano tra di loro, facendo sì che ad ognuno dei due gameti giunga solo uno degli alleli stessi.
- Legge dell'assortimento indipendente: durante la formazione dei gameti, geni diversi si distribuiscono l'uno indipendentemente dall'altro.

## Net-Logo

Net-Logo è un ambiente di sviluppo per programmare agenti dentro un ecosistema simulato. Di tutti i tool visti durante il corso di *methods in computer science education: analysis* Net-logo ci è sembrato quasi una scelta obbligata poiché era l'unico ideato appositamente per programmare simulazioni con un focus quasi totalmente incentrato sugli agenti.

Net-logo nasce nel 1999 con due obiettivi, il primo di dare a professionisti (privi di background informatico) di vari settori uno strumento semplice per creare simulazioni, l'altro di insegnare concetti di informatica a studenti tramite la programmazione di agenti. In questi anni net-logo è stato utilizzato da classi scolastiche che variano dalle elementari alle superiori e da ricercatori universitari per creare simulazioni in moltissimi ambiti fra i quali chimica, biologia, economia, sociologia e fisica; non sorprende che sia stato utilizzato anche per dei paper scientifici pubblicati. Ad oggi Net-logo non soltanto è utilizzato ma anche mantenuto e aggiornato con nuove features, l'ultima release disponibile è di un mese prima della scrittura di questa relazione.

Abbiamo trovato Net.logo un linguaggio interessante e molto particolare. Considerando l'importanza che la scuola dà all'insegnamento del "pensiero computazionale" piuttosto che all'apprendimento della programmazione in sé, potrebbe essere molto stimolante proporre un linguaggio funzionale che funziona con dei criteri diversi rispetto ai classici Python, C e Java. Riuscire a distinguere le turtles, dalle patches e dall'observer è un modo per sviluppare la logica richiesta, scrivere delle procedure di conseguenza costringe ad utilizzare i meccanismi conosciuti, ma sotto una prospettiva diversa. Questo è buono perché la logica dei programmi viene svincolata dall'abitudine dell'utilizzo di una sintassi sempre uguale.



### 2.1 Limiti di Netlogo

Netlogo non ha particolari limiti, soprattutto se utilizzato per fare simulazioni, ovvero ciò per cui è stato ideato. Siamo riusciti senza molti problemi a fare ciò che volevamo, laddove forse l'unica difficoltà è stata l'utilizzo della grafica delle tartarughe; ci sarebbe piaciuto fare qualcosa di più complesso, che non vincolasse la personalizzazione di un solo colore per tartaruga. D'altro canto, per un linguaggio mirato a fare simulazioni su ampia scala, è sufficiente una grafica molto basilare e simbolica, soprattutto se si considera che non è uno strumento

didattico adatto ad un pubblico molto giovane e che quindi non necessita di essere coinvolto tramite l'estetica.

### 2.1.1 Documentazione di NetLogo

Il grave problema di Net-Logo è la documentazione che è davvero confusionaria e dispersiva. Sul sito di Net-Logo c'è un [tutorial](#) per cominciare che si limita a spiegare i comandi base e come connettere i pulsanti dell'interfaccia al codice. Per la spiegazione dei comandi più avanzati esiste una [pagina web](#) che raccoglie e descrive tutti i comandi disponibili in Net-logo, non di tutti sono presenti esempi di codice. Fortunatamente l'IDE offre una collezione di modelli d'esempio con il relativo codice da cui si può imparare come fare le cose più complesse in Net-logo, tuttavia la sintassi specifica da conoscere non è affatto poca, probabilmente se organizzati meglio i manuali, si potrebbe procedere all'utilizzo di Net-logo con molte meno difficoltà.

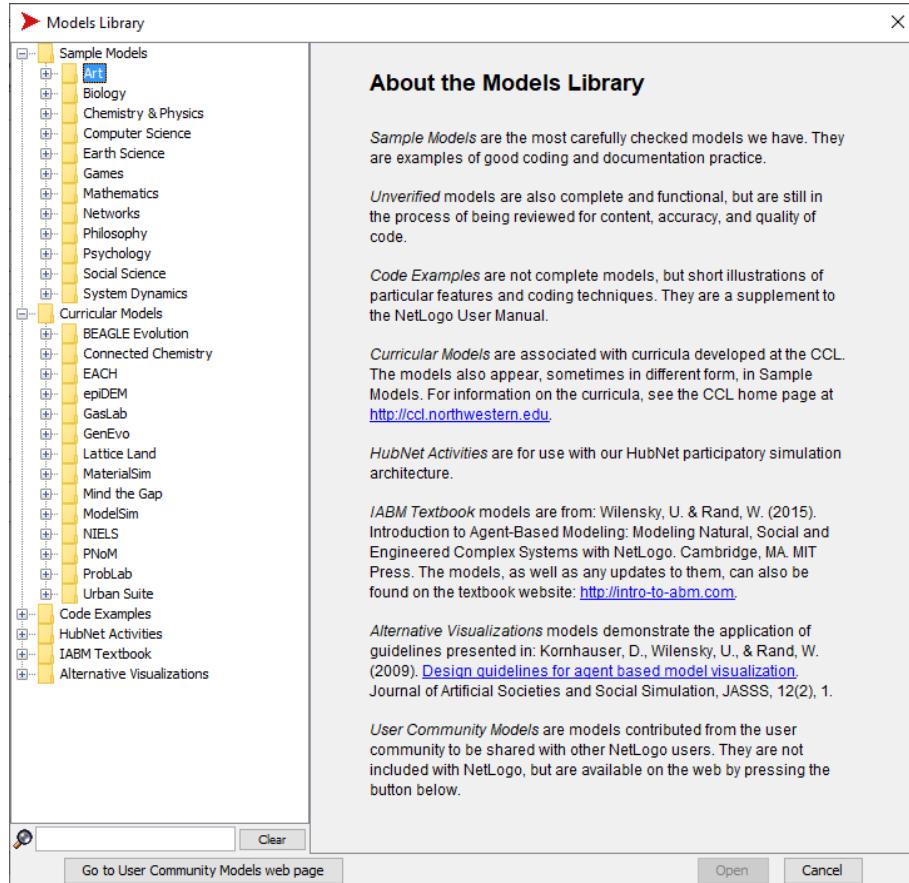


Figura 3: Esempi di modelli disponibili in net-logo

## Target e requisiti della la lezione

Questa learning unit è pensata per ragazzi che vanno tra i 16 e i 19 anni. Infatti tratta un argomento di biologia che in molti indirizzi scolastici è affrontato in uno degli ultimi tre anni, e ha dei prerequisiti che richiedono di usare con disinvoltura diverse competenze, sia informatiche, che interdisciplinari.

### 3.1 Prerequisiti didattici interdisciplinari

I ragazzi devono avere ben chiara la teoria della genetica mendeliana per poterla modellare. Il meccanismo di ereditarietà, di alleli recessivi e dominanti, di genotipo e fenotipo, devono essere chiari. Altre nozioni richieste sono le basi elementari della probabilità, saper leggere criticamente dei grafici. Per quanto riguarda la relazione da scrivere a fine esperienza gli unici requisiti che ci sentiamo di dare sono quelli di saper trattare un testo con un software di tipo editor di testo (Microsoft Word, Libre office, Google document) e saper utilizzare un foglio elettronico (Excel, Calc, Google fogli). Un altro prerequisito molto utile è una buona conoscenza dell'inglese per comprendere la documentazione di NetLogo.

### 3.2 Prerequisiti didattici informatici

Oltre ad una conoscenza preliminare di tutti i concetti logici che stanno alla base dell'informatica e del pensiero computazionale, bisogna avere una certa dimestichezza con i concetti di algoritmo e procedura. Inoltre questa non è una prima lezione su Net-Logo: gli studenti devono aver già studiato la struttura principale di questo strumento, sia dal punto di vista dell'interfaccia che dal punto di vista del codice. Devono saper usare monitor, grafici, bottoni e slider, saper impostare l'aspetto delle turtles e gestire la quantità di patches. Dal punto di vista del codice devono conoscere l'utilizzo delle procedure setup e run, come chiamare una procedura, creare le turtles anche con dei breed, e chiamare operazioni su di esse. Consigliamo poi di analizzare in classe alcuni esempi di modelli disponibili nella libreria dei modelli (Figura 3), in particolare *Wolf Sheep Predation* è un esempio né complicato e né troppo simile al task di GOM in cui all'interno si possono trovare molti dei comandi che serviranno per implementare la simulazione di GOM. E' inoltre utile che gli studenti siano già abituati a leggere documentazione di linguaggi di programmazione e a cercare ciò che gli serve online, una conoscenza che non bisogna dare per scontata, ma utile anche per il semplice fine di leggere altri esempi a proprio piacere.

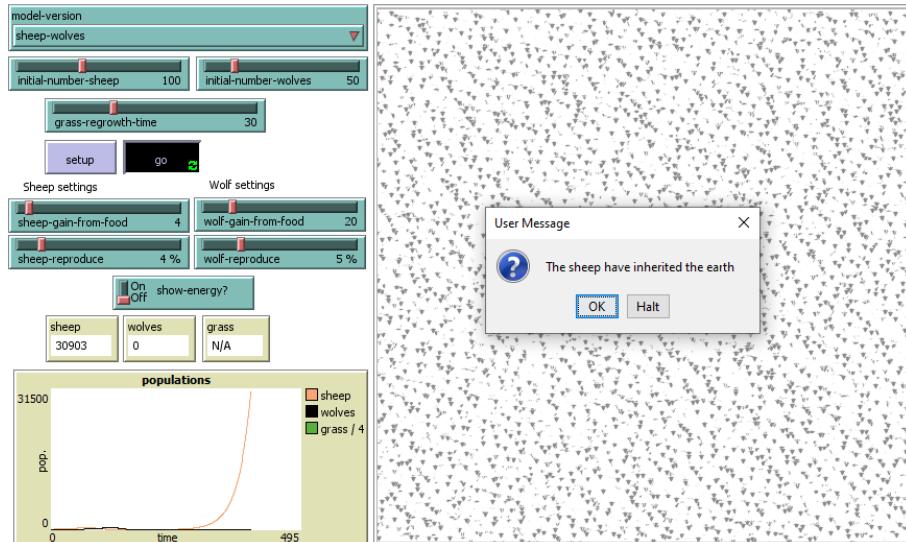
### 3.3 Obiettivi didattici interdisciplinari

L'obiettivo principale dell'esperienza è capire a fondo i concetti delle leggi di Mendel. Gli studenti impareranno ad analizzare e interpretare i fenomeni che

succedono in un ambiente simulato, consideranno la loro capacità descrivere una relazione scientifica. Come obiettivo secondario invece far abituare i ragazzi ad imparare ad usare un tool per eseguire un esperimento di laboratorio è sicuramente un'esperienza formativa in vista di una possibile iscrizione ad una facoltà scientifica universitaria (come per esempio fisica), oltre che essere uno stimolo per la curiosità dei ragazzi che possono così avere idee per effettuare verifiche su ciò che studiano e non solo.

### 3.4 Obiettivi didattici informatici

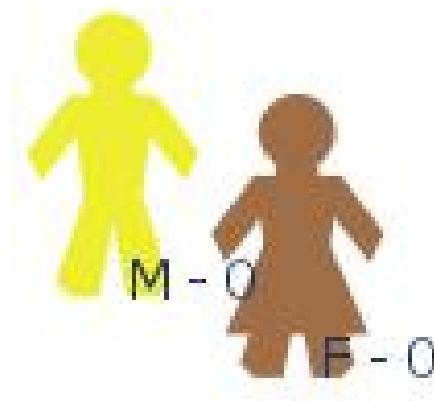
Questa esperienza è pensata per ragazzi che già hanno cominciato ad utilizzare NetLogo e che possiedono anche una certa esperienza e dimestichezza con l'informatica in generale. L'obiettivo è quindi quello di rafforzare e mettere alla prova le loro conoscenze specialmente per quanto riguarda il tool di NetLogo e il pattern di programmazione funzionale. Da un punto di vista più astratto è interessante far vedere agli studenti cosa succede nel mondo simulato quando fanno cambiamenti al codice e come spesso bisogna dover aggiungere vincoli alla simulazione per evitare che si blocchi o che rallenti troppo. Un esempio si può vedere in *Wolf Sheep Predation* dove la simulazione termina non appena le pecore diventano più di trentamila.



Inoltre, potendo facilmente regolare i parametri tramite gli slider, si introduce il concetto di tuning nei programmi e i ragazzi apprenderanno come gli esiti possano variare molto a seconda dei valori scelti per i parametri.

## Come svolgere l'esperienza

*Game of Mendel: A song of blonds and browns* è pensato per essere principalmente un'esperienza di laboratorio in cui il docente di informatica guida gli studenti nell'implementazione della simulazione assistendoli nel caso di difficoltà. L'obiettivo di questa simulazione è di creare un mondo abitato da esseri umani che possono essere castani o biondi e di due sessi opposti. Questi, se si incontrano nel mondo, possono riprodursi e generare un nuovo essere umano che ha uguali probabilità di essere maschio o femmina, mentre il colore dei capelli è deciso tramite le leggi di Mendel. In questo caso l'allele dominante è quello castano mentre il biondo è quello recessivo. Questi individui infine avranno un'età che servirà per decidere quando farli morire.



Una volta che la classe avrà realizzato con successo questa simulazione, il docente darà come nuovo obiettivo quello di aggiungere il gene del colore degli occhi alla simulazione e di poterci apportare miglioramenti e rifiniture a discrezione dello studente. In questa fase gli studenti saranno meno seguiti nella realizzazione del task e, nel caso in cui non riescano a completarlo durante il laboratorio, potranno continuare a lavorarci a casa. Successivamente ci sarà un nuovo incontro dove gli studenti dovranno analizzare ciò che osservano tramite la simulazione, prima nei singoli gruppi, poi facendo un confronto tra i risultati dei diversi gruppi.

### 4.1 Sviluppare GOM in classe

I ragazzi sono, in una prima fase, guidati nella creazione del programma. Ovviamente è richiesto che abbiano un minimo di praticità con NetLogo o non riusciranno a stare dietro all'insegnante, che ogni tanto dovrà fermarsi per verificare che tutto proceda correttamente. A seconda delle classi questa fase potrebbe essere più improntata come una lezione frontale o più interattiva, dove con studenti molto svegli si potrebbero anche chiamare a turno al computer del professore per far programmare loro.

### **Parti di codice programmate insieme**

- Creazione degli individui, con i due breed, uno per gli uomini uno per le donne
- Setup delle tartarughe: nasceranno in posizioni casuali
- Procedure per nascere senza genitori: le tartarughe avranno un genere casuale ed un genotipo dei capelli casuale (successivamente entrambi scelti in accordo ad una probabilità specificata). Inoltre avranno anche una quantità casuale di energia.
- Procedura per muoversi: ad ogni tick le tartarughe crescono di un anno, fanno un passo casuale e chiamano la procedura per morire e quella per riprodursi. La loro energia aumenta di uno
- Procedura per morire: le tartarughe, se superano una certa età, muoiono
- Procedura per la riproduzione: le tartarughe femmine in una fascia di età fertile, nel caso in cui abbiano abbastanza energia e una tartaruga maschio anch'essa fertile nella stessa posizione generano con lui un figlio. Loro perdono l'energia, il figlio viene settato ereditando le caratteristiche genetiche dei genitori

Successivamente verrà richiesto agli studenti di apportare alcune modifiche al codice e all'interfaccia definiti insieme:

### **Parti di codice programmate dagli studenti**

- La modifica della funzione di riproduzione in modo che i figli vengano creati impostando come alleli un allele casuale da ogni genitore, e che il fenotipo dei figli sia concorde con i loro alleli
- Specificare la proporzione iniziale dei diversi genotipi: recessivo omozigote, dominante omozigote o eterozigote
- Creare slider nell'interfaccia che sostituiscano le costanti usate nel codice. Questi slider devono regolare: la quantità di uomini e di donne iniziali, la grandezza delle turtles, la massima quantità di individui, la percentuale iniziale dei diversi genotipi
- Creare i monitor per visualizzare il numero di individui castani e biondi attualmente presenti
- Creare un grafico che mostri nel tempo la variazione del numero di biondi e castani e uno per la variazione del numero di maschi e femmine.

Una volta che gli studenti avranno realizzato tutto ciò, con un paio di ore di lezione, la simulazione che avranno modellerà due delle tre leggi di Mendel.

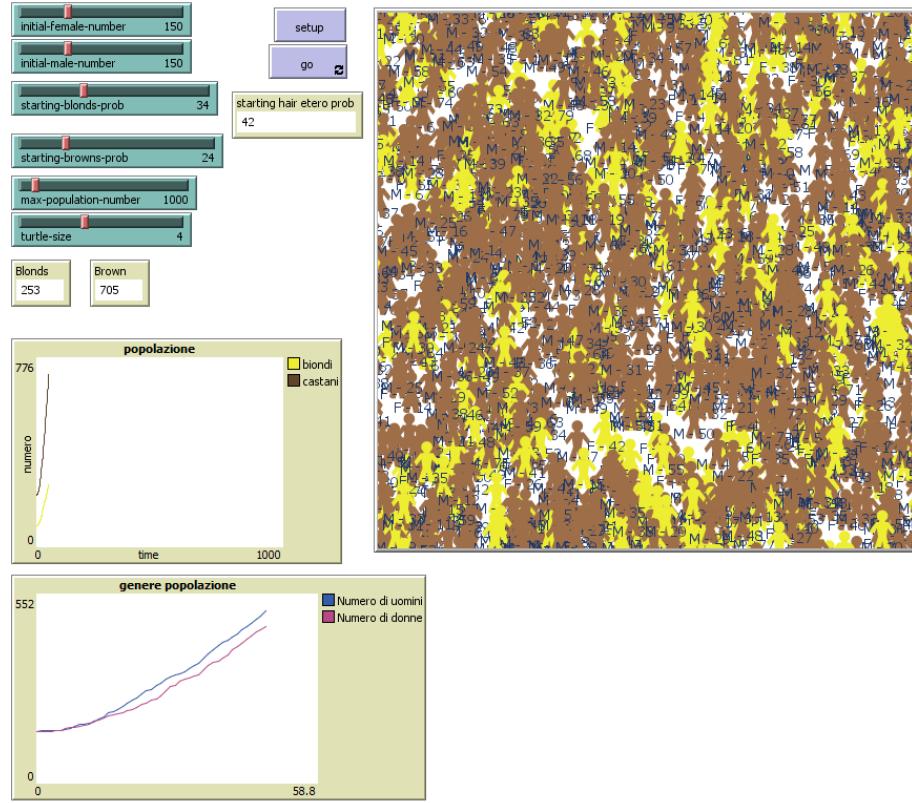


Figura 4: Una possibile interfaccia per GOM

## 4.2 Task avanzati

Il principale task avanzato è quello di aggiungere il gene per gli occhi alle turtles e quindi aggiungere la terza legge di Mendel alla simulazione. Per completare questo task bisogna non soltanto sapersi muovere nel codice, ma anche capire come gestire graficamente questa aggiunta dentro Net-logo. Il docente di informatica può consigliare agli studenti di abbandonare le texture a cui applicare un colore e utilizzando il tool per la creazione di *turtle shape* di Net-logo creare una shape per ogni possibile combinazione di geni e assegnarla in base agli alleli dell'individuo.



Ci sono diversi aspetti migliorabili per rendere la simulazione più accurata, un esempio potrebbe essere quello di trovare un algoritmo per calcolare quando far morire una turtle o per gestire la fertilità in base al sesso della turtle. Su questo task non vogliamo influenzare gli studenti dicendogli cosa fare ma vogliamo che siano loro ad analizzare la propria simulazione e decidere quindi cosa cambiare.

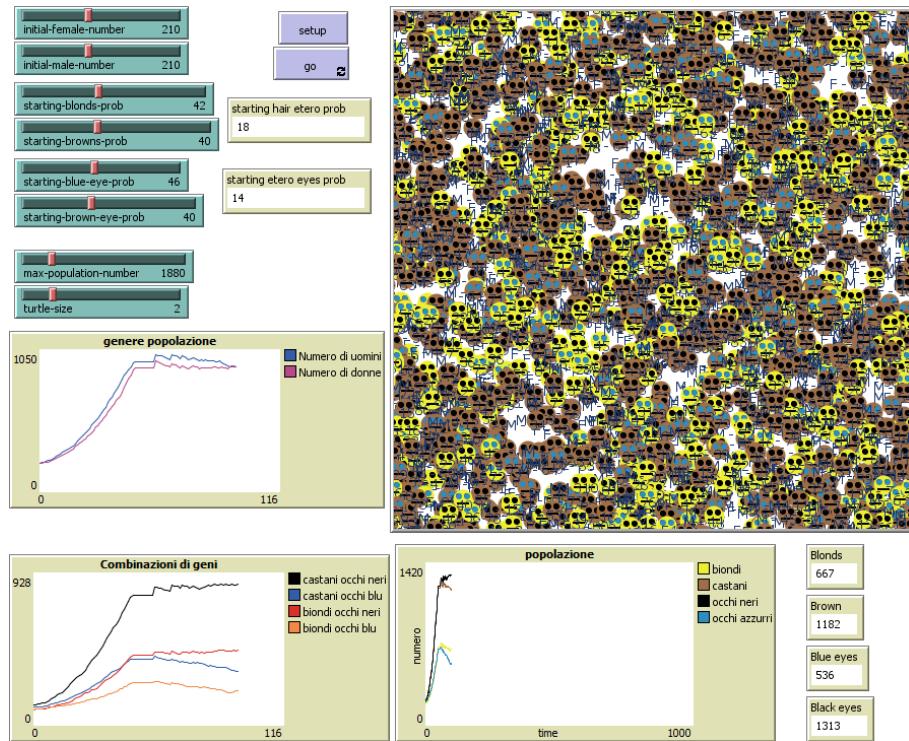


Figura 5: Una possibile interfaccia per una simulazione "avanzata"

### 4.3 Discussione in aula

GOM prevede un secondo incontro in laboratorio in cui gli studenti mostrano ed eseguono la propria simulazione davanti la classe e il docente e in seguito si discute assieme anche agli altri studenti sui risultati e su cosa ci si aspettava sarebbe successo durante la simulazione. Questo incontro non soltanto serve per discutere le simulazioni ma può essere anche visto come un incontro di round up per avere un feedback dal professore sulla propria simulazione e quindi in caso di poterla correggere prima di scrivere la relazione.

### 4.4 La relazione

Adesso che l'ambiente da osservare è pronto, gli studenti dovranno avviare il sistema ed osservarlo con diverse combinazioni di parametri date da noi. Prima di far fare ai ragazzi le loro osservazioni, o in laboratorio o a casa, si discute su cosa si aspettano che succeda con le varie configurazioni. L'insegnante prenderà appunti, per vedere se alla fine l'esperimento confermerà o smentirà le supposizioni degli studenti. Le combinazioni dei parametri che abbiamo scelto per la relazione sono:

Numero di donne e di uomini iniziali	Percentuali iniziali di biondi omozigoti, castani eterozigoti e castani omozigoti	Massima quantità di individui	Dimensione delle tartarughe
200 e 200	25, 50 e 25	1550	4
300 e 300	25, 50 e 25	5000	2
200 e 200	15, 40 e 45	1700	4
200 e 200	45, 40 e 15	1700	4
200 e 200	60, 20 e 20	1700	4
200 e 200	0, 50 e 50	1700	4

Per gli studenti che hanno realizzato la versione avanzata della simulazione, consigliamo sia di provare a inserire le stesse percentuali di omozigoti ed eterozigoti dominanti e recessive e mischiare combinazioni diverse per occhi e capelli delle percentuali da noi proposte in modo da vedere se i due geni hanno lo stesso comportamento. Oltre a queste combinazioni che avvieranno per un numero di tick (almeno 5000) che sembrerà a loro sufficiente, gli studenti dovranno cercarne un altro paio interessanti. Nella relazione dovranno spiegare se avvengono cose particolari, commentando sempre i punti seguenti:

### Caratteristiche da osservare

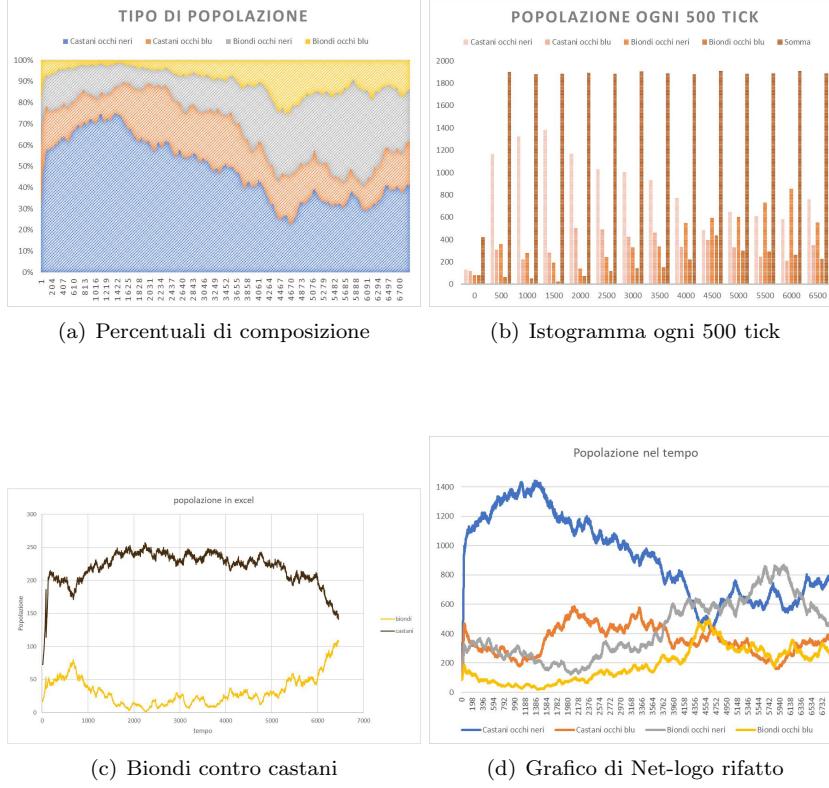
- Dopo almeno 5000 iterazioni uno dei due fenotipi scompare? Se sì scrivere quale e perché
- Nel corso del tempo come varia la proporzione tra i due fenotipi? Sapresti spiegarne il motivo?

Un'interessante feature di Net-Logo è quella di poter esportare i dati di un grafico in un file CSV. È possibile quindi caricare questo file CSV in un foglio di calcolo per poter calcolare statistiche quali la media di popolazione di biondi o castani, il massimo e il minimo raggiunto da esse e altre a discrezione degli studenti e del docente di scienze.

Tempo	Castani occhi neri	Castani occhi blu	Biondi occhi neri	Biondi occhi blu	Valori iniziali			
					Initial-female-number	210		
					starting-brown-eye-prob	40		
0	133	121	84	82	starting-blonds-prob	42		
1	140	124	87	82	turtle-size	2		
2	140	125	89	82	initial-male-number	210		
3	144	128	90	85	max-population-number	1880		
4	149	130	92	85	starting-browns-prob	40		
5	152	132	94	86	starting-blue-eye-prob	46		
6	159	134	95	86				
7	163	135	95	87				
8	168	141	102	89				
9	177	147	103	90				
10	178	147	106	90	Media	917,597	1442	615
11	185	150	108	90	Castani occhi neri	354,959	588	328
12	191	156	111	90	Castani occhi blu	430,558	869	317
13	201	163	115	92	Biondi occhi neri	188,861	499	84
14	212	169	117	93	Biondi occhi blu			
15	230	175	120	94				
16	245	181	121	97				
17	261	187	123	99				
18	272	192	127	103				
19	287	192	128	106				
20	302	199	131	108				
21	317	208	133	112				
22	335	215	133	115				
23	352	222	135	120				
24	370	228	138	120				
25	392	233	147	120				
26	406	241	155	122				
27	426	250	159	123				
28	442	257	166	125				
29	460	264	172	129				
30	479	269	181	129				
31	499	278	188	133				

Figura 6: CSV importato ed elaborato in Microsoft Excel

Dai fogli di calcolo è possibile anche creare nuovi grafici da poter utilizzare nella relazione. Anche qui le possibilità sono molteplici e sta all'inventiva dello studente decidere se e quali grafici utilizzare come corredo alle proprie teorie sui fenomeni della simulazione. Un consiglio che il professore può dare è quello di rifare il grafico della popolazione di Net-Logo nel foglio di calcolo poiché questo non dà grandi possibilità di personalizzazione dal punto di vista grafico.



## 4.5 Confronto

Poiché ciò che osserveranno i ragazzi potrebbe essere definito da una casualità, le loro osservazioni fatte nelle relazioni verranno confrontate. In questo modo si cercherà di capire se le leggi genetiche possono, e in che modo, regolare in modo stabile il fenotipo della popolazione.

## 4.6 Valutazione

Per questa Learning Unit, soprattutto per dal punto di vista informatico, abbiamo pensato ad una valutazione più morbida trattandosi più di una lezione interattiva. Ciononostante verrà comunque assegnato un voto in decimi, basato sui seguenti parametri parametri:

### **Schema di valutazione**

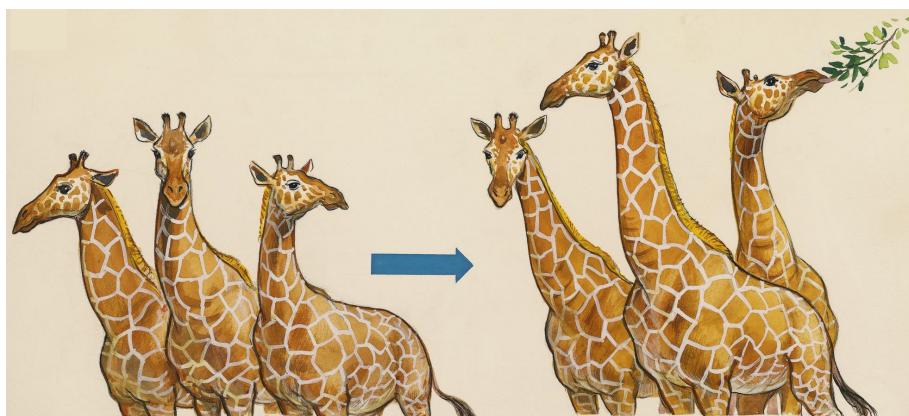
- 3 punti per aver correttamente completato il programma per l'osservazione delle prime due leggi di Mendel
- 2 punti per l'aggiunta del tratto degli occhi
- 2 punti per l'utilizzo corretto dell'interfaccia
- 3 punti per l'analisi presente nella relazione, comprese le tabelle

Nei primi due punti della valutazione si terrà conto, nell'assegnazione di un voto per punto, della qualità del codice oltre che del raggiungimento degli obiettivi. Inoltre un punto extra verrà assegnato a studenti che si mostreranno particolarmente coinvolti o che faranno osservazioni particolarmente interessanti durante l'attività. Un punto extra verrà assegnato anche ai ragazzi che faranno una personalizzazione del sistema particolarmente originale.

Starà alla discrezione dell'insegnante di scienze decidere se valutare anche lui i ragazzi basandosi su questo lavoro e, se sì, come. La relazione sarà comunque parte della valutazione del voto di informatica perché permette di osservare quanta attenzione i ragazzi hanno prestato alla logica dell'attività.

## Altre possibilità per riadattare la lezione

Net-Logo è un tool con grandi potenzialità ma con una curva d'apprendimento inizialmente molto ripida, sarebbe quindi un peccato non sfruttarlo una volta che gli studenti ne hanno preso dimestichezza. Partendo da **GOM** si può passare a simulazioni un po' più complesse da far fare agli studenti anche senza il supporto diretto e costante del docente di informatica, per esempio sempre riguardo l'argomento della genetica una simulazione sulle teorie di **Darwin** sull'evoluzione delle specie prendendo il caso del collo delle giraffe dovrebbe essere un compito alla portata per studenti che sono riusciti a completare GOM.



I concetti genetici sono davvero molti: esiste la dominanza parziale, alleli di caratteristiche genetiche diverse associate (i conigli con il pelo bianco hanno anche gli occhi rossi), la terza legge di Mendel o fattori di mutazione: si potrebbe di conseguenza anche pensare un lavoro a gruppi, successivo a questo, in cui con delle simulazioni fate con NetLogo gli studenti spieghino ai compagni alcuni aspetti della genetica.

Potrebbe essere interessante (soprattutto per istituti tecnici informatici) far reimplementare GOM agli studenti utilizzando una diversa tecnologia che abbia un approccio diverso da quello funzionale e orientato agli agenti di Net-Logo. Un esempio potrebbe essere **Processing**, una libreria grafica per Java molto ben documentata e che è disponibile con un proprio IDE che semplifica alcuni aspetti di Java. Reimplementare GOM con Processing significherebbe rifarlo utilizzando un approccio **object oriented**, lavorare alla grafica con Processing è sicuramente più semplice rispetto a Net-logo e si potrebbe quindi chiedere ai ragazzi di fare la simulazione con una grafica migliore; infine assieme al docente si potrebbe analizzare e discutere quali sono state le differenze fra i due tool e in cosa uno è meglio dell'altro, soprattutto evidenziando le differenze dei paradigmi di programmazione diversi.