

GENÉTICA E PROGRAMAÇÃO COMPUTACIONAL: PROPOSTA METODOLÓGICA PARA UMA ABORDAGEM INTERDISCIPLINAR

Míriam Conceição de Souza Testasicca¹, Carlos Oliveira Pereira¹, Célio Junio Mendonça das Dôres¹, Osvaldo Novais Junior¹ e Thalita Macedo Araújo¹

1. Instituto Federal de Educação Ciência e Tecnologia de Minas Gerais (IFMG), *Campus Ouro Preto*, Ouro Preto, Minas Gerais, Brasil.

RESUMO

As dificuldades para o ensino-aprendizagem de Genética demandam o uso de metodologias alternativas, como as atividades interdisciplinares. Nesse sentido, a Programação Computacional, em função do uso de raciocínio lógico e dedutivo para formulação de algoritmos corretos, pode contribuir para a implementação dessas práticas. Assim, o presente trabalho buscou investigar a aplicabilidade do pensamento computacional e da Programação Computacional no ensino de Genética. Os alunos envolvidos no projeto foram orientados a produzir algoritmos capazes de resolver atividades sobre monoibridismo e diibridismo. Foram implementados quatro protótipos, com nível de dificuldade crescente, tanto em função da temática da atividade quanto da linguagem de programação utilizada. Os dois primeiros protótipos foram desenvolvidos a partir de atividades sobre monoibridismo com uso da linguagem C++. O terceiro protótipo também foi delineado a partir de uma atividade de monoibridismo, mas utilizando-se a ferramenta App Inventor. Por fim, no quarto protótipo, os alunos utilizaram o APP Inventor para resolução de uma atividade de diibridismo. Ao longo do processo, pode ser observado o ganho cognitivo em relação à compreensão dos mecanismos de divisão meiótica, combinação alélica e dominância, associação entre genótipo e fenótipo, cálculo matemático de probabilidade, abstração, ordenação e organização do pensamento computacional. Com isso, conclui-se que o uso da lógica computacional para resolução de problemas palpáveis pode representar um ganho para aprendizagem de ambas disciplinas envolvidas, reforçando conceitos e terminologias específicas da genética à medida que exercita a criatividade e a persistência durante o desenvolvimento das rotinas de programação.

Palavras-chave: Interdisciplinaridade, Genética e Programação Computacional.

ABSTRACT

The difficulties for teaching and learning genetics require the application of alternative methodologies in the classroom, such as interdisciplinary activities. Computational Programming, by reason of the use of logical and deductive reasoning for the formulation of correct algorithms, constitutes a propitious subject for such a practice. The present work has

tried to investigate the applicability of computational thinking and computational programming in the genetics teaching. The students involved in the project were instructed to produce algorithms capable of solving monohybrid-cross and dihybrid-cross activities. One has implemented four prototypes with an increasing level of difficulty as a consequence of both the activity theme and the used programming language. The first two prototypes were developed based on monohybrid activities by using the C++ language. The third prototype was also devised based on a monohybrid-cross activity, but by using the APP Inventor tool. Finally, in the fourth prototype, the students used the APP Inventor tool to solve a dihybrid-cross activity. Throughout the process, one could observe the cognitive gain in understanding the mechanisms of meiotic division, allelic combination and dominance, association between genotype and phenotype, mathematical calculation of probability, abstraction, ordering and organization of the computational thinking. Based on that, one has concluded that the use of computational logic to solve concrete problems may represent a gain for learning both involved subjects, emphasizing on the one hand genetic specific concepts and terminologies, while bringing about on the other hand the practice of creativity and persistence during the development of programming routines.

Keywords: Interdisciplinarity, Genetics and Computer Programming.

1. INTRODUÇÃO

A Biologia é, dentre as ciências naturais, aquela dedicada ao estudo de todos os aspectos relacionados aos seres vivos e às suas relações com o ambiente. Sua aprendizagem possibilita o desenvolvimento de habilidades necessárias à compreensão de acontecimentos e temas da realidade (BRASIL, 1999). Dentre as diversas ramificações da Biologia, encontra-se a Genética, dedicada ao estudo dos genes e dos mecanismos de transmissão da hereditariedade (KNIPPELS, 2002). Por sua importância como base conceitual para a compreensão da evolução, bem como pelo fato de ser uma área de rápida expansão com importantes implicações econômicas, éticas e sociais, a Genética é um dos temas mais abordados em estudos de didática em Biologia (CASAGRANDE, 2006).

De acordo com os Parâmetros Curriculares Nacionais para o Ensino Médio (PCNEM), o estudo da Genética requer a aplicação de conhecimentos de outros conteúdos, pois envolve noções de citologia, bioquímica, probabilidade e análise combinatória (BRASIL, 2000). A partir dos conceitos genéticos “é possível ao aluno relacioná-los às tecnologias de clonagem, engenharia genética e outras ligadas à manipulação do DNA, proceder a análise desses fazeres humanos identificando aspectos éticos, morais, políticos e econômicos envolvidos na produção científica e tecnológica, bem como na sua utilização; o aluno se transporta de um cenário meramente científico para um contexto em que estão envolvidos vários aspectos da vida humana” (BRASIL, 2000).

Embora haja consenso quanto ao papel dos conhecimentos em Genética para a compreensão de fenômenos cotidianos, diversos estudos apontam a existência de dificuldade de aprendizado desse tema em diferentes níveis de ensino (BAHAR; JOHNSTONE; HANSELL, 1999; KNIPPELS, 2002; FABRÍCIO et al., 2006; BRÃO; BENEVIDES PEREIRA, 2015; LEAL; BARBOSA; RÔÇAS, 2016; LEAL; MEIRELLES; RÔÇAS, 2019). Esse fato resulta de problemas tanto de ensino, quanto de aprendizagem, uma vez que professores e alunos podem apresentar dificuldades em relação à abordagem de Genética em sala de aula.

Segundo Knippels (2002), as principais dificuldades no ensino de Genética podem ser classificadas em cinco categorias, as quais, nem sempre, ocorrem separadamente: I. uso de um vocabulário muito específico, com terminologia excessivamente técnica; II. necessidade de conteúdo matemático para compreensão das probabilidades associadas às Leis de Mendel; III. necessidade de conhecimentos prévios de bioquímica e citologia; IV. distanciamento cronológico, entre os conteúdos de bioquímica, citologia e genética, em função da sequência do currículo e conseqüente aumento no nível de abstração; V. natureza complexa do tema, em função dos diferentes níveis de pensamento necessários à compreensão do papel bioquímico da molécula de DNA na transmissão das características fenotípicas dos seres vivos.

Desse modo, o ensino de Genética torna-se um processo desafiador. Dentre as alternativas metodológicas que podem favorecer a aprendizagem de um tema tão complexo quanto a Genética, muitos professores têm adotado o uso de metodologias ativas de aprendizagem (BRÃO; BENEVIDES PEREIRA, 2015; SIQUEIRA; ALTINO FILHO; DUTRA, 2021), atividades lúdicas (BRÃO; BENEVIDES PEREIRA, 2015; MASCARENHAS et al., 2016) e o enfoque interdisciplinar ou multidisciplinar (THIESEN, 2008; ELIAS; RONQUIM, 2020). Nesse último caso, os PCNEM (BRASIL, 2000) reforçam a necessidade da articulação interdisciplinar entre os saberes sob uma perspectiva integradora. A interdisciplinaridade é

“um movimento que acredita na criatividade das pessoas, na complementaridade dos processos, na inteireza das relações, no diálogo, na problematização, na atitude crítica e reflexiva, enfim, numa visão articuladora que rompe com o pensamento disciplinar, parcelado, hierárquico, fragmentado, dicotomizado e dogmatizado que marcou por muito tempo a concepção cartesiana de mundo. Portanto, a interdisciplinaridade é um movimento importante de articulação entre o ensinar e o aprender. Compreendida como formulação teórica e assumida enquanto atitude, tem a potencialidade de auxiliar os educadores e as escolas na resignificação do trabalho pedagógico em termos de currículo, de métodos, de conteúdos, de avaliação e nas formas de organização dos ambientes para a aprendizagem” (THIESEN, 2008).

A adoção de projetos educacionais interdisciplinares permite ao aluno vivenciar correlações entre teoria e prática, com um olhar mais amplo sobre a realidade, favorecendo uma formação mais responsável, crítica e criativa, mais alinhada às demandas sociais e às possibilidades de inclusão a partir de todas as possíveis formas de se aprender e ensinar (THIESEN, 2008). No entanto, o desenvolvimento de ensaios genuinamente interdisciplinares ainda representa um desafio, tanto para professores, quanto para alunos, pois requer que esses saiam da sua zona de conforto, quebrem paradigmas, hábitos e acomodações, se sujeitando às incertezas do novo, desconhecido (ELIAS; RONQUIM, 2020).

Mas a escola é, justamente, o local de aprendizagem, discente e docente, onde o conhecimento se constrói e se reconstrói. Portanto, é o ambiente propício para que se corram os riscos de trabalhar, conjuntamente, disciplinas e conteúdos que aparentemente não se inter-relacionam, como, por exemplo, Biologia e Programação Computacional.

Os sistemas computacionais, tais como *softwares* e jogos, vem sendo amplamente utilizados em projetos interdisciplinares, uma vez que a Ciência da Computação é, por vocação, uma área interdisciplinar. Desde seu surgimento, a Ciência da Computação sempre esteve relacionada à solução de problemas das mais diversas áreas de conhecimento (ALMEIDA, 2000).

A programação de computadores exige do programador raciocínio lógico e dedutivo para formulação de algoritmos corretos. Tal habilidade, definida como raciocínio computacional, pode ser definida “como a capacidade de resolução de problemas de forma sistemática, usando dedução e abstração, habilidades muito bem trabalhadas na Ciência da Computação” (FERREIRA et al., 2015).

Nos últimos anos, diversos projetos utilizando raciocínio computacional foram realizados na educação básica, o que demonstra um aumento do interesse de pesquisadores sobre o tema (BORDINI et al., 2016; DE ARAUJO; ANDRADE; GUERRERO, 2016). No entanto, a maior parte desses trabalhos não apresenta como foco a interdisciplinaridade (NASCIMENTO; SANTOS; TANZI, 2018).

A utilização do pensamento computacional durante a resolução de problemas leva em consideração as seguintes habilidades: I. estruturar a solução do problema baseada no uso de um computador ou de outra ferramenta análoga; II. organizar e analisar os dados de forma lógica; III. utilizar abstrações, tais como modelos ou simulações, para representar os dados; utilizar pensamento algorítmico, ou seja, com uma sequência de etapas; IV. Identificar, analisar e formular propostas capazes de solucionar o problema de forma eficiente; V.

generalizar e transferir as etapas do processo para resolução de outros problemas (WING, 2006). Além disso, algumas atitudes como confiança e persistência para lidar com situações complexas, comunicando-se e trabalhando com pares em prol de um objetivo comum, podem ser desenvolvidas ou reforçadas por essas habilidades (DE ARAUJO; ANDRADE; GUERRERO, 2016).

O enfoque interdisciplinar se faz ainda mais interessante no contexto do ensino médio integrado ao ensino técnico, garantido pela Lei de Diretrizes e Bases da Educação Nacional (LDBEN), no qual as formações básica e técnica ocorrem concomitantemente (BRASIL, 1996). Considerando que, nesse caso, o aprendizado do estudante deve ter, como foco principal, a prática profissional, a oferta de conteúdos dinâmicos, contextualizados e aplicados se faz, mais do que nunca, necessária.

Diante do exposto, o presente trabalho buscou investigar a aplicabilidade do pensamento computacional e da Programação Computacional no ensino de Genética a alunos do curso técnico de Automação Industrial integrado ao ensino médio em uma escola pública de Ouro Preto, Minas Gerais, na tentativa de colaborar com a identificação de métodos de ensino-aprendizagem interdisciplinares voltados à educação profissional.

2. MATERIAIS E MÉTODO

O projeto foi desenvolvido com dois alunos bolsistas, estudantes do terceiro ano do Ensino Médio integrado ao curso técnico de Automação Industrial, de uma escola pública localizada no município de Ouro Preto, Minas Gerais, no ano de 2018.

No desenvolvimento do presente trabalho, foram utilizados os preceitos da pesquisa-prática, uma vez que o projeto almejava contribuir com o desenvolvimento dos alunos e da prática docente a partir da proposição de mudanças, visando a melhoria da aprendizagem e da autonomia discente. As etapas do projeto foram delineadas a partir das quatro fases do ciclo básico da investigação-ação (TRIPP, 2005), em três ciclos subsequentes, conforme representado na figura 1.



Figura 1. Representação das fases dos ciclos da investigação-ação utilizados durante a execução do projeto.

Na fase de “planejamento”, foi realizada a avaliação dos níveis de fluência dos alunos participantes em relação aos conteúdos Genética e Programação computacional a partir da avaliação do histórico escolar e por meio de uma entrevista informal para averiguação dos seus conhecimentos prévios.

Na sequência, dentro da etapa de “ação” determinada por TRIPP (2005), foram selecionadas atividades, extraídas do livro didático (LINHARES; GEWANDSZNAJDER, 2013) para que os alunos iniciassem o uso do pensamento computacional como ferramenta para organização das etapas de resolução. Os exercícios escolhidos abordavam a herança de um único par de alelos (Primeira Lei de Mendel). Para obtenção dos protótipos a linguagem de programação utilizada foi a C++, uma das linguagens mais utilizadas para programação orientada (JUNIOR; VIRTUOSO; MARTINS, 2012) em função da sua facilidade de uso, gratuidade e baixa exigência de recursos de *hardware*.

Os resultados obtidos a partir da “observação” do funcionamento dos protótipos e das linhas de comando desenvolvidas foram analisados seguindo o critério de categorização de Bardin (1977). Na fase de pré-análise, foi definido o *corpus*, constituído das atividades extraídas do livro didático, dos *scripts* desenvolvidos pelos estudantes e das impressões em tela da execução dos protótipos. O registro dos *scripts* e da execução dos protótipos foi realizado pela captura da tela do *prompt* de comando, com posterior arquivamento das imagens obtidas. Os documentos foram, em seguida, categorizados, sendo as unidades de registro definidas como os próprios documentos. Por fim, o tratamento dos resultados foi feito pela análise dos dados contidos nos documentos por meio da inferência da mensagem, propriamente dita, e da interpretação das informações nela contidas (BARDIN, 1977).

O primeiro ciclo de investigação-ação foi finalizado com a “reflexão” acerca do ganho cognitivo dos alunos, mensurado a partir da capacidade de execução das atividades propostas.

De acordo com a teoria de desenvolvimento intelectual no estilo espiral, a aprendizagem ocorre pela passagem sucessiva em um mesmo ponto, mas em um nível superior, de maior complexidade (VYGOTISKY, 2003). Justamente nesse sentido, foram propostos o segundo e o terceiro ciclos de investigação-ação. Em ambos, a fase de “planejamento” foi baseada na análise da fluência dos conhecimentos adquiridos nas etapas do ciclo imediatamente anterior. A principal diferença existente entre o segundo e o terceiro ciclos de investigação-ação diz respeito ao incremento no nível de dificuldade dos exercícios propostos, em função da temática (monoibridismo e diibridismo) e da complexidade da linguagem de programação adotada.

Desse modo, no segundo ciclo de investigação-ação, também foram extraídas atividades de um livro didático (LINHARES; GEWANDSZNAJDER, 2013) com temática relacionada ao monoibridismo. No entanto, foi proposto o uso da ferramenta App Inventor, em função da maior disponibilidade de recursos, apesar do incremento no grau de complexidade exigido na execução das rotinas de programação. Já no terceiro ciclo de investigação-ação, os exercícios se baseavam na herança de dois pares de alelos, ou diibridismo (Segunda Lei de Mendel), novamente com uso do App Inventor.

As etapas de “observação” e “reflexão” do segundo e terceiro ciclos de investigação-ação seguiram-se de modo semelhante ao realizado no primeiro ciclo. Para Tripp (2005), é exatamente esse processo, repetitivo, contínuo e cíclico que definem a pesquisa-ação, de modo que melhorias sejam propostas, em sequência, ao longo do processo, a partir da análise dos resultados e de novo planejamento.

3. RESULTADOS E DISCUSSÃO

A seguir serão apresentados os resultados obtidos em cada um dos três ciclos de investigação-ação aplicados.

3.1. PRIMEIRO CICLO DE INVESTIGAÇÃO-AÇÃO: DESENVOLVIMENTO DE ATIVIDADES SOBRE MONOIBRIDISMO UTILIZANDO A LINGUAGEM DE PROGRAMAÇÃO C++

Os experimentos de monoibridismo ou herança de um único par de alelos, realizados por Mendel, subsidiaram todo o conhecimento sobre hereditariedade que possuímos hoje. Nesses experimentos, foram observados os resultados fenotípicos da prole resultante dos cruzamentos entre ervilhas da espécie *Pisum sativum* (BRÃO; BENEVIDES PEREIRA, 2015). Nessa temática, o primeiro exercício apresentado aos alunos relacionava-se ao cálculo da coloração das sementes de ervilhas resultantes do cruzamento no qual poderia ser realizada a escolha do genótipo parental (LINHARES; GEWANDSZNAJDER, 2013).

A resolução de problemas genéticos exige a apreensão de terminologia específica e a ordenação de processos de combinação alélica (genética) que resultará na manifestação das características fenotípicas. Utilizando a linguagem C++, os alunos implementaram essa ordenação de processos para que o cálculo fenotípico fosse realizado pela máquina, a partir de comandos lógicos, aqui descritos como *Script 1*, conforme explicitado no quadro 1.

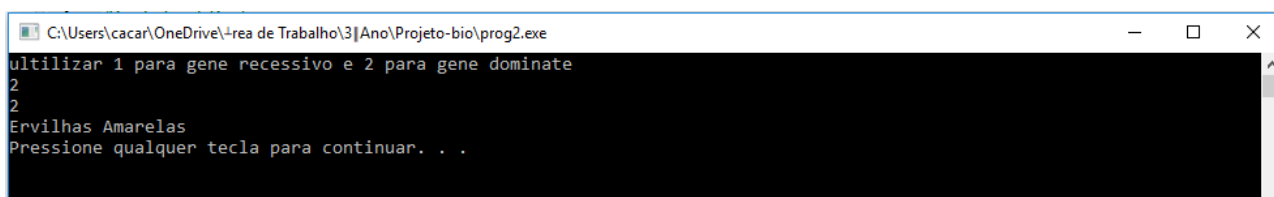
Quadro 1. Script 1: comandos, redigidos em C++, para execução de uma rotina capaz de determinar a coloração das sementes de ervilhas, após cruzamento.

```
include<stdio.h>
#include<stdlib.h>
int main()
{
int car,car2;
printf("utilizar 1 para gene recessivo e 2 para gene dominante\n");
scanf("%i",&car);
scanf("%i",&car2);
    if (car==1 && car2==1)
        printf("Ervilhas Verdes \n");
    else if((car==1 && car2==2)|| (car==2 && car2==1))
        printf("Ervilhas Verdes \n");
    else
        printf("Ervilhas Amarelas \n");

    system("pause");
return(0);
}
```

O protótipo 1 (Figura 2), resultante da execução do *Script 1*, permitia a interação com o usuário na medida em que solicitava, como entrada, os dados referentes ao alelo doado por cada um dos parentais, codificados como “P” e “M”. A sequência lógica do *Script 1* descrevia as possibilidades de combinação dos alelos parentais, segundo escolha do

usuário, e gerava, na forma de texto impresso na tela do computador, o fenótipo apresentado pela prole.



```
C:\Users\caca\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\prog2.exe
utilizar 1 para gene recessivo e 2 para gene dominante
2
2
Ervilhas Amarelas
Pressione qualquer tecla para continuar. . .
```

Figura 2. Impressão em tela da execução do Script 1, exemplificando a escolha do genótipo dominante para ambos parentais (linhas 2 e 3) e, como resposta, o fenótipo calculado (linha 4).

Cabe, aqui, uma análise mais aprofundada da sexta linha de comando do *Script 1* (QUADRO 1), a qual determina a impressão em tela da instrução ao usuário para escolha do genótipo parental. Pode-se perceber o uso indevido do termo “gene” em detrimento do termo “alelo”. A falta de clareza em relação a terminologia em genética, bastante relatada em outros estudos (KNIPPELS, 2002; BRÃO; BENEVIDES PEREIRA, 2015; LEAL; MEIRELLES; RÔÇAS, 2019), é uma das maiores dificuldades relatadas pelos estudantes em relação a essa temática. Para além disso, pode-se observar que, ao realizar a codificação das informações, associando numerais aos fenótipos, como linguagem alternativa mais facilmente interpretada pela máquina e, também, visando a facilidade de interação com o usuário, os alunos demonstraram alta capacidade de abstração e domínio do processo.

Um segundo exemplo de aplicação dos preceitos da hereditariedade de um fenótipo relacionado a um único par de alelos foi proposto aos alunos (adaptado de LINHARES; GEWANDSZNAJDER, 2013). Nesse caso, o comprimento dos pelos, longos ou curtos, de cobaias (porquinhos-da-índia), seriam determinados pelos alelos dos parentais, segundo escolha do usuário. Como resposta, o algoritmo deveria calcular a probabilidade de ocorrência, na prole, de cada um dos fenótipos possíveis.

A sintaxe criada, aqui denominada *Script 2*, encontra-se descrita no quadro 2. A solicitação de entrada pelo usuário, dessa vez, foi o genótipo de cada um dos parentais: homocigoto dominante, heterocigoto ou homocigoto recessivo. Nota-se que, nesse caso, os alunos optaram pela associação da descrição do fenótipo a título de informação adicional ao usuário (linhas 6, 7, 9 e 10).

Quadro 2. Script 2: comandos, redigidos em C++, para execução de uma rotina capaz de determinar a probabilidade fenotípica do comprimento dos pelos de cobaias, após cruzamento.

```
#include<stdio.h>
#include<stdlib.h>
int main()
{
int P,M;
printf("digite o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto
recessivo - Pelos longos)\n");
scanf("%i",&P);
printf("digite o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3
homozigoto recessivo - Pelos longos)\n");
scanf("%i",&M);
if(P==1 && M==1){
printf("100 por cento de filhotes com pelos curtos\n");
}
else if((P==1 && M==2)||(P==2 && M==1)){
printf("100 por cento de filhotes com pelos curtos\n");
}
else if((P==1 && M==3)||(P==3 && M==1)){
printf("50 por cento de filhotes com pelos curtos e 50 por cento de filhotes com pelos longos\n");
}
else if(P== 2 && M==2){
printf("75 por cento de filhotes com pelos curtos e 25 por cento de filhotes com pelos longos\n");
}
else if(P==3 && M==3){
printf("100 por cento de filhotes com pelos longos\n");
}
system("pause");
return(0);
}
```

Durante a execução do *Script 2*, tão logo o usuário fizesse a escolha do genótipo dos parentais, seriam impressas, na tela, as probabilidades genotípicas da prole, conforme observado na figura 3.

Uma análise dos resultados demonstra que os alunos apresentaram capacidade de compreensão i) dos mecanismos de divisão meiótica dos alelos parentais; ii) das possibilidades de combinação de alelos provenientes de eventos independentes (produção de gametas dos parentais); iii) dos cálculos matemáticos de probabilidade, iv) da associação implícita entre genótipo e fenótipo e i) da existência de dominância entre os dois alelos. Além disso, ao serem comparados os *Scripts 1* e *2*, nota-se que, em função da maior complexidade da segunda atividade, foi requerido um número maior de comandos, pensados e ordenados de modo a contemplar todas as respostas possíveis.

```

I:\sers\cacar\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\teste.exe
o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
r cento de filhotes com pelos curtos
one qualquer tecla para continuar. . .

I:\sers\cacar\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\teste.exe
o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
r cento de filhotes com pelos curtos
one qualquer tecla para continuar. . .

I:\sers\cacar\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\teste.exe
o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
r cento de filhotes com pelos curtos e 50 por cento de filhotes com pelos longos
one qualquer tecla para continuar. . .

I:\sers\cacar\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\teste.exe
o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
r cento de filhotes com pelos curtos e 25 por cento de filhotes com pelos longos
one qualquer tecla para continuar. . .

I:\sers\cacar\OneDrive\+rea de Trabalho\3\Ano\Projeto-bio\teste.exe
o genotipo do pai (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
o genotipo da mãe (1:homozigoto dominante - Pelos curtos,2 heterozigoto - Pelos curtos, 3 homozigoto recessivo - Longos)
r cento de filhotes com pelos longos
one qualquer tecla para continuar. . .

```

Figura 3. Impressão em tela da execução do *Script 2*, exemplificando o cálculo probabilístico do fenótipo da prole a partir da escolha do genótipo dos parentais; ambos homozigotos dominantes (A), pai homozigoto dominante e mãe heterozigoto (B), pai homozigoto dominante e mãe homozigoto recessivo (C), ambos heterozigotos (D) e ambos homozigotos recessivos (E).

Um trabalho desenvolvido com alunos do Ensino Médio Integrado de uma escola pública estadual, visando o desenvolvimento da lógica de programação como ferramenta interdisciplinar, demonstrou que esse tipo de abordagem pode apresentar uma série de contribuições para a aprendizagem do aluno: maior facilidade de memorização dos conteúdos, melhora na criatividade e na persistência, promoção da socialização e o desenvolvimento do raciocínio crítico (RIOS; CURY, 2016). Portanto, evidenciam-se os potenciais benefícios ao ensino-aprendizagem de ambas as disciplinas envolvidas: Biologia e Programação Computacional.

A reflexão sobre os resultados obtidos no primeiro ciclo de investigação-ação permitiu aos envolvidos na pesquisa a proposição de novas atividades, dando início ao segundo ciclo de investigação-ação.

3.2. SEGUNDO CICLO DE INVESTIGAÇÃO-AÇÃO: DESENVOLVIMENTO DE ATIVIDADES SOBRE MONOIBRIDISMO UTILIZANDO A FERRAMENTA APP INVENTOR

O App Inventor é uma ferramenta desenvolvida pelo Google e mantida pelo Massachusetts Institute of Technology (MIT) que permite o desenvolvimento de aplicativos para celulares Android em um ambiente de programação de funcionalidades e *design* de interfaces (WOLBER, 2011). Além disso, o “App Inventor propicia um ambiente de aprendizagem baseado no construcionismo, uma vez que permite aos estudantes criarem aplicações à medida que descobrem e exercitam sua criatividade, tornando o aprendizado mais lúdico” (GOMES; DE MELO, 2013). Tudo isso faz com que essa ferramenta se torne uma alternativa bastante viável para uso no ensino de programação em ambiente escolar.

Desse modo, foi proposto aos alunos a repetição do segundo exercício realizado no ciclo de investigação-ação anterior, mas, dessa vez, utilizando o App Inventor. As FIGURAS 4 e 5 representam, respectivamente, o *Script* 3 e o Protótipo 3, obtidos a partir da atividade proposta, ou seja, do cálculo das probabilidades fenotípicas da prole de cobaias, em relação ao comprimento dos pelos, a partir da escolha do genótipo parental.

A análise dos comandos utilizados na construção do *Script* 3, comparativamente ao *Script* 2, permite inferir que, ainda que haja diferenças estruturais, em função da especificidade da linguagem adotada, a lógica de programação, em relação às etapas e raciocínio computacional, se mantêm. Uma das principais diferenças entre essas duas linguagens diz respeito ao perfil de interação com o usuário e às possibilidades estéticas da impressão em tela, como pode ser avaliado na figura 5.

A ordenação lógica das ideias é considerada etapa fundamental para o raciocínio computacional exigido no desenvolvimento de programas, dentre as quais podem ser destacadas: organização, criatividade, perseverança, padronização e otimização (JUNIOR, 2009). De modo semelhante, além de outras habilidades cognitivas, o raciocínio lógico, necessário ao entendimento de Genética, é um passo fundamental para a solução de problemas.

```

when Button1 Click
do
set VerticalArrangement3 Visible to true
set VerticalArrangement1 Visible to false
set global P to "1"

when Button2 Click
do
set VerticalArrangement3 Visible to true
set VerticalArrangement1 Visible to false
set global P to "2"

when Button3 Click
do
set VerticalArrangement3 Visible to true
set VerticalArrangement1 Visible to false
set global P to "3"

when Button4 Click
do
set VerticalArrangement3 Visible to false
set VerticalArrangement1 Visible to false
set global M to "1"
set btninicia Visible to true
set btnreset Visible to true

when Button5 Click
do
set VerticalArrangement3 Visible to false
set VerticalArrangement1 Visible to false
set global M to "2"
set btninicia Visible to true
set btnreset Visible to true

when Button6 Click
do
set VerticalArrangement3 Visible to false
set VerticalArrangement1 Visible to false
set global M to "3"
set btninicia Visible to true
set btnreset Visible to true

when Screen1 Initialize
do
set VerticalArrangement1 Visible to true
set VerticalArrangement3 Visible to false
set lb1 Visible to false
set lb2 Visible to false
set lb3 Visible to false
set lb4 Visible to false
set btnreset Visible to false
set btninicia Visible to false

when btninicia Click
do
if (get global P == "1" and get global M == "1")
then set lb1 Visible in true
if (get global P == "2" and get global M == "1" or get global P == "1" and get global M == "2")
then set lb2 Visible in true
if (get global P == "3" and get global M == "1" or get global P == "2" and get global M == "2")
then set lb3 Visible in true
if (get global P == "1" and get global M == "2")
then set lb4 Visible in true
set btnreset Visible in true

when btnreset Click
do
set VerticalArrangement1 Visible to true
set VerticalArrangement3 Visible to false
set lb1 Visible to false
set lb2 Visible to false
set lb3 Visible to false
set lb4 Visible to false
set btnreset Visible to false
set btninicia Visible to false

```

Figura 4. Script 3: programação em blocos, utilizando a ferramenta APP Inventor, para execução de uma rotina capaz de determinar a probabilidade fenotípica do comprimento dos pelos de cobaias, após cruzamento.



Figura 5. Impressão em tela da execução do Script 3, exemplificando a tela apresentada ao usuário para escolha do genótipo do pai (A) e o resultado do cálculo probabilístico do fenótipo da prole a partir da escolha de ambos os parentais homozigotos dominantes (B).

Novamente, os resultados obtidos no segundo ciclo de investigação-ação subsidiaram a decisão dos membros da equipe envolvida na pesquisa de propor atividades com maior nível de complexidade.

3.3. TERCEIRO CICLO DE INVESTIGAÇÃO-AÇÃO: DESENVOLVIMENTO DE ATIVIDADES SOBRE DIIBRIDISMO UTILIZANDO A FERRAMENTA APP INVENTOR

Não satisfeito com os resultados alcançados a partir dos experimentos de monoibridismo, Mendel passou a conduzir novos ensaios, dessa vez, observando o comportamento de dois fenótipos, simultaneamente, após cruzamento entre indivíduos da espécie *P. sativum* (diibridismo). Nesse caso, considerando-se genes localizados em cromossomos distintos, os fenótipos da prole resultam de eventos de segregação alélica independentes (BRÃO; BENEVIDES PEREIRA, 2015).

A compreensão e aplicação dos preceitos da Segunda Lei de Mendel, como é denominada a lei de recombinação ou da segregação independente, requer o domínio de uma série de conhecimentos, tais como já mencionado na seção 3.1. Para além deles, também é necessária a associação de conhecimentos matemáticos envolvendo cálculos com eventos probabilísticos simultâneos. Por tudo isso, não é incomum que alunos apresentem dificuldades de aprendizagem dessa temática (MASCARENHAS et al., 2016).

Durante a execução do terceiro ciclo de investigação-ação, considerando-se a consolidação dos conhecimentos adquiridos em relação ao uso da ferramenta App Inventor,

foi proposta, aos alunos, a resolução de uma atividade de cálculo fenotípico da prole de plantas a partir de dois fenótipos relacionados às pétalas das flores: cor (vermelha ou branca) e aspecto (estreita ou larga). Para tanto, os estudantes escreveram o *Script 4* (Figura 6), cujas telas de execução estão exemplificadas na figura 7.

```

initialize global p to 0
initialize global s to 0

when Screen1.Initialize
do
  set va2.Visible to false
  set btnInicia.Visible to false
  set btnReset.Visible to false

when BTN1VL.Click
do
  set global p to 0
  set va2.Visible to true
  set va1.Visible to false

when btn2VL.Click
do
  set global s to 0
  set btnInicia.Visible to true
  set va2.Visible to false

when BTN1VE.Click
do
  set global p to 1
  set va2.Visible to true
  set va1.Visible to false

when btn2VE.Click
do
  set global s to 1
  set btnInicia.Visible to true
  set va2.Visible to false

when BTN1BL.Click
do
  set global p to 2
  set va2.Visible to true
  set va1.Visible to false

when btn2BL.Click
do
  set global s to 2
  set btnInicia.Visible to true
  set va2.Visible to false

when BTN1BE.Click
do
  set global p to 3
  set va2.Visible to true
  set va1.Visible to false

when btn2BE.Click
do
  set global s to 3
  set btnInicia.Visible to true
  set va2.Visible to false

when btnReset.Click
do
  set va1.Visible to true
  set va2.Visible to false
  set btnInicia.Visible to false
  set btnReset.Visible to false
  set lblResult.Text to 0

when btnInicia.Click
do
  if (get global p = 0 and get global s = 0)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Vermelho e La."
  if (get global p = 0 and get global s = 1) or (get global p = 1 and get global s = 0)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Vermelho e In."
  if (get global p = 0 and get global s = 2) or (get global p = 2 and get global s = 0)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Róseo e Largo."
  if (get global p = 0 and get global s = 3) or (get global p = 3 and get global s = 0)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Róseo e Inter."
  if (get global p = 1 and get global s = 1)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Vermelho e Es."
  if (get global p = 1 and get global s = 2) or (get global p = 2 and get global s = 1)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Róseo e Inter."
  if (get global p = 1 and get global s = 3) or (get global p = 3 and get global s = 1)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Róseo e Estr."
  if (get global p = 2 and get global s = 2)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Branco e Largo."
  if (get global p = 2 and get global s = 3) or (get global p = 3 and get global s = 2)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Branco e Inte."
  if (get global p = 3 and get global s = 3)
  then set lblResult.Text to "100% dos cruzamentos resultarão em Branco e Estr."
  set btnReset.Visible to true
  set btnInicia.Visible to false
  
```

Figura 6. *Script 4*: programação em blocos, utilizando a ferramenta APP Inventor, para execução de uma rotina capaz de determinar a probabilidade fenotípica das pétalas de flores em relação à cor e formato, após cruzamento.

Como previsto, o nível de complexidade exigido para construção dos blocos e montagem do *Script 4*, comparativamente ao *Script 3*, é consideravelmente maior em função da necessidade de avaliação de uma quantidade maior de possibilidades e, portanto, do volume de cálculos. No entanto, uma vez que a ferramenta APP Inventor permite uma configuração estética amigável ao usuário, as telas de interação apresentadas (Figura 7A e B) permanecem limpas e claras.

É importante salientar que os resultados apresentados constituem a versão final de sucessivas tentativas e erros anteriores. Além da sua gratuidade, uma das maiores vantagens do APP Inventor é a possibilidade de avaliação do funcionamento do aplicativo, de modo instantâneo, durante o seu desenvolvimento. Com isso, os alunos podem realizar infinitos ensaios até que o algoritmo apresente o funcionamento desejado, com total liberdade para erros e acertos, aprendendo continuamente.

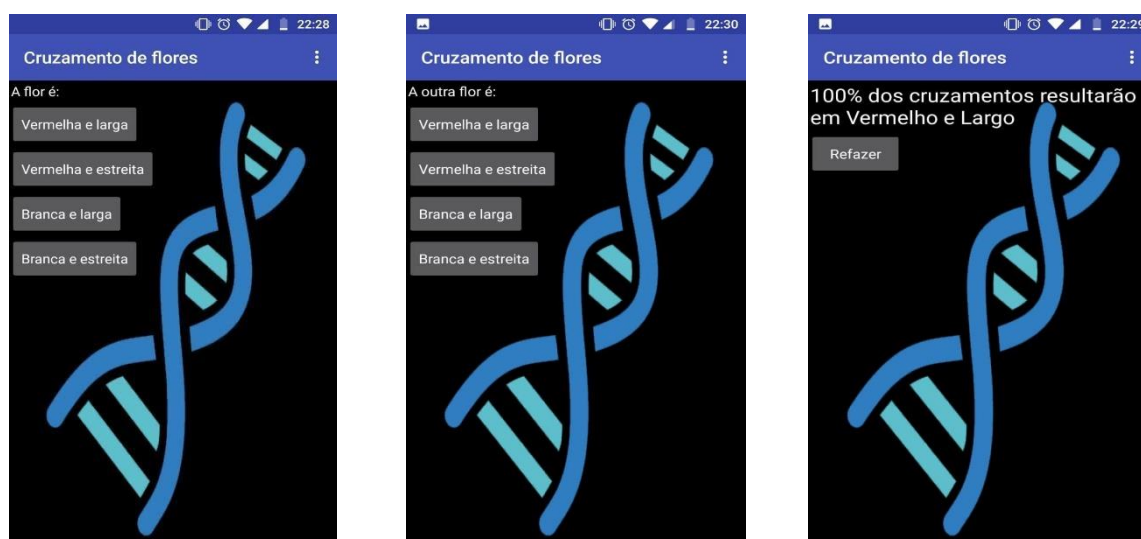


Figura 7. Impressão em tela da execução do *Script 4*, exemplificando a tela apresentada ao usuário para escolha dos fenótipos parentais (A) e (B) e o resultado do cálculo probabilístico do fenótipo da prole a partir da escolha de ambos os parentais homocigotos dominantes (C).

O APP Inventor tem sido utilizado como ferramenta didática para o aprendizado de programação em diversas instituições de ensino básico, médio e superior, não só para formação de alunos, mas, também, de professores (WOLBER, 2011). Apenas mais recentemente essa ferramenta tem sido relatada na literatura como possível instrumento de promoção de interdisciplinaridade. Como exemplo, pode ser citado um trabalho realizado com alunos de graduação do curso de Tecnologia em Bancos de Dados de uma faculdade

pública do estado de São Paulo para desenvolvimento de aplicativos voltados à solução de problemas do seu cotidiano (DA SILVA JACINTO et al., 2018). Uma outra pesquisa, realizada com alunos do curso de graduação de Análise e Desenvolvimento de Sistemas de uma faculdade pública do estado de São Paulo resultou no desenvolvimento de jogos de aprendizagem de língua inglesa. Em comum, esses estudos enfatizam as vantagens que o desenvolvimento das práticas interdisciplinares representa para a autonomia intelectual dos estudantes a partir da aprendizagem significativa propiciada, especialmente, pela possibilidade de resolução de problemas cotidianos.

Nos últimos anos, tem sido observado o aumento no desenvolvimento e implementação da lógica de programação para alunos de diferentes níveis de ensino. Ainda maior é a eficiência da aprendizagem de programação e algoritmos quando é utilizada uma abordagem divertida e criativa, buscando a contextualização em problemas cotidianos dos alunos (SILVA; SILVA; SANTOS, 2009).

4. CONCLUSÃO

Os resultados obtidos no presente projeto permitem concluir que o uso da lógica computacional, a partir da implementação de rotinas de programação, podem contribuir para que estudantes exercitem a capacidade de resolução de atividades de genética e reforcem conceitos e terminologias específicas desse ramo de estudo da Biologia. Por outro lado, a oportunidade de utilizar, durante o delineamento dos *scripts*, exemplos que possam representar situações palpáveis, aplicáveis a outras áreas de conhecimento, pode tornar mais significativo o ensino-aprendizagem de Programação Computacional.

5. REFERÊNCIAS

- ALMEIDA, M. E. B. Informática e formação de professores. Coleção Informática para a mudança na Educação. **Brasília, DF: Ministério da Educação**, 2000.
- BAHAR, M.; JOHNSTONE, A. H.; HANSELL, M. H. Revisiting learning difficulties in biology. **Journal of Biological Education**, v. 33, n. 2, p. 84–86, 1999.
- BARDIN, L. **Análise de conteúdo**. Lisboa: Edições 70, 1977.
- BORDINI, A.; et al. Computação na Educação Básica no Brasil : o Estado da Arte Computer

Science in the Brazilian Basic Education : State of the Art Abstract : **Revista de Informática Teórica e Aplicada**, v. 23, n. 2, p. 210–238, 2016.

BRÃO, A.; BENEVIDES PEREIRA, A. Biotecnética: possibilidades do jogo no ensino de genética. **REEC: Revista electrónica de enseñanza de las ciencias**, v. 14, n. 1, p. 55–76, 2015.

BRASIL. **Lei n. 9.394, de 20 de dezembro de 1996. estabelece as Diretrizes e Bases da Educação Nacional**. Brasília, DF. Diário Oficial [da] República Federativa do Brasil, 1996.

BRASIL. **Parâmetros Curriculares Nacionais do Ensino Médio: Ciências da Natureza Brasília: Ministério da Educação**. Brasília, DF: MEC/Semtec, 2000.

BRASIL. **Parâmetros Curriculares Nacionais**. Brasília, DF: MEC/Semtec, 1999.

CASAGRANDE, G. L. **A genética humana no livro didático de biologia**. (Dissertação) Mestrado em Educação científica e Tecnológica - Universidade Federal de Santa Catarina, Florianópolis, Santa Catarina, 2006.

DE ARAUJO, A. L. S. O.; ANDRADE, W. L.; GUERRERO, D. D. S. A systematic mapping study on assessing computational thinking abilities. **IEEE Frontiers In Education Conference (Fie)**, 2016.

ELIAS, M. A.; RONQUIM, J. Ensino De Genética Por Meio Da Interdisciplinaridade Entre Biologia E Planejamento Urbano. **Arquivos do Mudi**, v. 24, n. 2, p. 22–29, 2020.

FABRÍCIO, M. F. L.; JÓFILI, Z. M. S.; SEMEN, L. S. M.; LEÃO, A. M. A. C. A compreensão das leis de Mendel por alunos de biologia na educação básica e na licenciatura. **Ensaio Pesquisa em Educação em Ciências**, v. 8, n. 1, p. 83–103, 2006.

FERREIRA, A. C.; MELHOR, A.; BARRETO, J.; PAIVA, L. F.; MATOS, E. Experiência Prática Interdisciplinar do Raciocínio Computacional em Atividades de Computação Desplugada na Educação Básica. **XXI Workshop de Informática na Escola (WIE 2015)**, 2015.

GOMES, T. C. S.; DE MELO, J. C. B. App Inventor for Android: Uma Nova Possibilidade para o Ensino de Lógica de Programação. **II Congresso Brasileiro de Informática na Educação**, Campinas, SP, 2013.

SANTOS JUNIOR, C. I. **Introdução a Lógica de Programação**. Joinville: Clube de Autores, 2009.

JUNIOR, M. A. C.; VIRTUOSO, G. H. F.; MARTINS, P. J. Propriedades desejáveis a uma linguagem de programação: Uma análise comparativa entre as linguagens C, C++ e Java. **VI SULCOMP**, Criciúma, SC, 2012.

KNIPPELS, M. C. P. J. **Coping with the abstract and complex nature of genetics in biology education**. Utrecht:CD-Press, 2002.

LEAL, C. A.; BARBOSA, J. V.; RÔÇAS, G. A genética e seus conteúdos estruturantes na investigação de livros do PNLD 2015. **Revista Eletrônica Debates em Educação Científica e Tecnológica**, v. 6, n. 03, p. 66–91, 2016.

LEAL, C.; MEIRELLES, R.; RÔÇAS, G. O QUE ESTUDANTES DO ENSINO MÉDIO PENSAM SOBRE GENÉTICA? Concepções discentes baseada na Análise de conteúdo. **Revista Eletrônica Científica Ensino Interdisciplinar**, v. 5, n. 13, p. 71–86, 2019.

- LINHARES, S.; GEWANDSZNAJDER, F. **Biologia hoje**. São Paulo: Ática, v. 3, 2013.
- MASCARENHAS, M. J. O.; DA SILVA, V. S. C.; MARTINS, P. R. P.; FRAGA, E. C.; BARROS, M. C. Estratégias metodológicas para o ensino de genética em escola pública. **Pesquisa em foco**, v. 21, n. 2, p. 5-24, 2016.
- NASCIMENTO, C.; SANTOS, D. A.; TANZI, A. Pensamento computacional e interdisciplinaridade na educação básica: um mapeamento sistemático. **VII Congresso Brasileiro de Informática na Educação**, Fortaleza, CE, 2018.
- RIOS, P. T. G.; CURY, D. Utilizando o SCRATCH no desenvolvimento de Lógica de Programação como contribuição interdisciplinar. **XII TISE-Nuevas Ideas en Informática Educativa**, Santiago, Chile, 2016.
- SILVA, I. F. A.; SILVA, I. M. M.; SANTOS, M. S. Análise de problemas e soluções aplicadas ao ensino de disciplinas introdutórias de programação. **IX JEPEX**, Recife-PE, 2009.
- SIQUEIRA, M. L. G.; ALTINO FILHO, H. V.; DUTRA, É. D. R. ENSINO DA GENÉTICA: UMA PROPOSTA DE ABORDAGEM AO ENSINO MÉDIO. **VI Seminário Científico do UNIFACIG**, Manhuaçu – MG, 2021.
- THIESEN, J. S. A interdisciplinaridade como um movimento articulador no processo ensino-aprendizagem. **Revista brasileira de educação**, v. 13, n. 39, p. 545–554, 2008.
- TRIPP, D. Pesquisa-ação: uma introdução metodológica. **Educação e Pesquisa**, v. 31, n. 3, p. 443–466, 2005.
- VYGOTISKY, L. S. **A formação social da mente: o desenvolvimento dos processos psicológicos superiores**. Trad. Por José Cipolla Neto; Luis Silveira Menna Barreto, Solange Castro Affeche. São Paulo: Editora Martins Fontes, 2003.
- WING, J. M. Computational thinking. **Communications of the ACM**, v. 49, n. 3, p. 33–35, 2006.
- WOLBER, D. App inventor and real-world motivation. **SIGCSE'11 - Proceedings of the 42nd ACM Technical Symposium on Computer Science Education**, 2011.