

EDUSCRATCH

PROGRAMANDO COM O



RESPOSTA MÚLTIPLA

Análise ao projecto “Escrever Português”

(Out-2010)

Não é possível estabelecer um método único para desenvolver um projecto Scratch, já que depende muito da inspiração e do gosto do programador; mas podemos sempre criar o nosso próprio método, vendo como os outros utilizam os seus. É esse o objectivo deste documento, em que se elegeu o projecto “Escrever Português” como exemplo de projectos de perguntas com resposta múltipla.

Todos os sprites deste projecto se escondem quando se clica na bandeira verde, porque o controlo inicial é feito no Palco. Como se pode ver ao lado, o Palco exhibe o “cenário1” com imagens de escritores portugueses e fica a aguardar que o utilizador o clique.

Quando é clicado, se acaso ainda está no “cenário1” anuncia “Início” para acordar os sprites; mas ele próprio recebe também este aviso para mudar para o cenário “Quadro”, inicializar os contadores “Erradas”, “Certas” e “Perg” e entrar num “repete 20” que lhe permite controlar a sequência das perguntas.

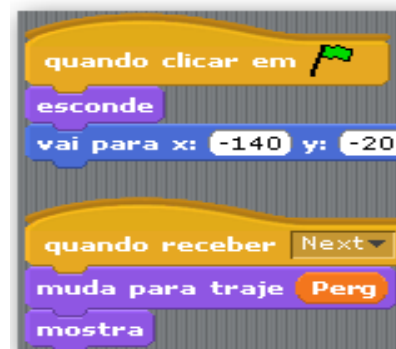
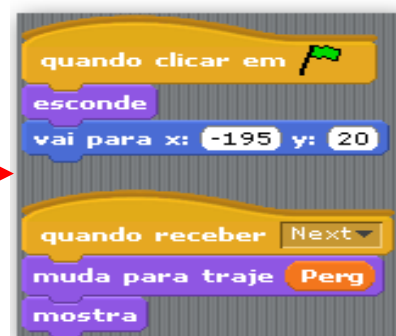
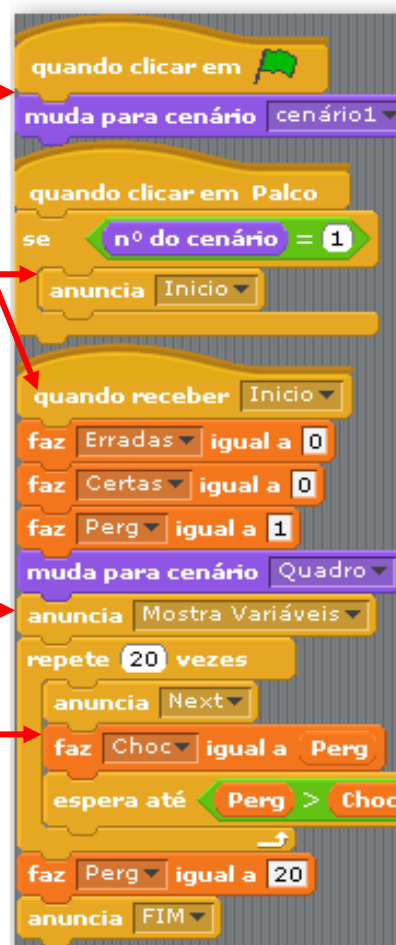
Repare-se que os demais cliques no Palco já não surtirão efeito, uma vez que o cenário deixou de ser o primeiro, inibindo-se assim que o anúncio “início” seja enviado de novo.

Mas há aqui um outro aspecto a necessitar explicação, porque se trata de um subterfúgio para contornar um “bug” da versão 1.4 que, permitindo que se escondam e mostrem as variáveis, não as mantém nos lugares em que foram postas. Então, foi criado o sprite “Máscara” que, inicialmente, tapa as variáveis com a fotografia de José Saramago e que, obedecendo ao aviso “Mostra Variáveis”, esconde-se para as destapar.

Dentro do ciclo “repete 20 vezes”, há o anúncio “Next” para passar à pergunta seguinte, seguido de uma reserva do valor da variável “Perg” na variável “Choc”. Esta simples acção permite que o Palco espere pela altura em que “Perg” deixa de ser igual a “Choc” e, só depois, é que volta ao início do ciclo para anunciar, com “Next”, a passagem para a pergunta seguinte. Vejamos: “Perg” foi inicializada igual a 1; e, “Choc” também ficou igual a 1. Depois, alguma coisa há-de incrementar “Perg” para que, ficando maior que “Choc” torne verdadeira a condição que está a travar a acção; e o ciclo recomeça, com “Perg” alegadamente = 2 e “Choc” a ficar também = 2. Isto vai ser feito 20 vezes e a última volta só é destravada com “Perg=21”, o que ficaria feio para mostrar no ecrã, no final de 20 perguntas, pelo que se força “Perg” a ser igual a 20. O aviso “FIM” vai ser ouvido pelo “Gato” para encerrar o projecto.

Entretanto, o sprite “Pergunta”, (que estava escondido e bem posicionado), acorda com o anúncio “Next” e muda para o seu traje nº 1, que era o actual valor da variável “Perg”. (De cada vez que receber “Next”, faz a mesma gracinha). Assim, desde que tenha 20 trajes, vai mostrá-los, ordenadamente, conforme “Perg” for sendo incrementada de 1 até 20, (ainda não se sabe onde).

E os trajes das respostas – “RespA”, RespB” e “RespC” – procedem de igual forma garantindo que a cada pergunta correspondam sempre três respostas, no ecrã. Mas estes sprites precisam de mais comandos, para saber o que fazer quando e se forem clicados. É o que veremos na página seguinte.



Tomando como exemplo o sprite “RespA” e sabendo que umas vezes será ele a resposta verdadeira e outras vezes a falsa, teria de se construir um bloco com vinte “Ses” aninhados, o que daria um bloco compridíssimo e muito complicado.

Então, para simplificar, pode dividir-se as vinte hipóteses em quatro grupos: 1-5, 6-10, 11-15 e 16-20, com um bloco para cada grupo. Assim, se “Perg” é maior que 5 “anuncia A06-10” (grupo 6-10 da resposta A). Se for menor ou igual a 5 trata-se neste bloco: se for 1 ou 3 está errada e leva “NÃO”. Se for 2, 4 ou 5 está certa e leva “SIM”

(Aquele “A” antes do nome 06-10 do grupo é para distinguir este aviso dos que as respostas “B” e “C” também fazem. Senão, todas respondiam aos anúncios umas das outras).

Passando ao bloco seguinte, apenas se sabe que “Perg” é maior do que 5; e é altura de ver se também é maior do que 10. Se for, passa o controlo para o bloco seguinte. Se não for, tem valores de 6 a 10 e trata-se aqui. Só a 7 está certa. 6, 8, 9 e 10 são “NÃO”

Desta forma, no bloco seguinte serão tratadas as respostas de 11 a 15 ou passa para outro bloco onde se tratam as 16-20; e assim sucessivamente para os quatro blocos,

Como temos vindo a verificar, quando classificamos se uma resposta está certa ou errada, apenas anunciamos “SIM” ou “NÃO” Estes anúncios vão servir para orientar o sprite “Gato”.

O “gato” estava escondido mas apareceu com o “Início” e até disse “Olá” durante 2 segundos, ficando à espera dos SIM e NÃO. Quando recebe “SIM” toca um som alegre, diz “Certo”, soma 1 na conta das respostas certas e soma 1 no valor de “Perg”. **CÁ ESTÁ.** Vimos na página anterior que o Palco estava à espera que “Perg” aumentasse para anunciar “Next”, lembram-se? É aqui que isso acontece.

Claro que no caso da resposta estar errada, toca-se um som triste, aumenta-se a contagem das erradas mas incrementa-se “Perg” na mesma, para passar à pergunta seguinte. Neste caso, porém, foi criada aqui uma variante para dar alguma explicação no caso da resposta estar errada. Começa com o anúncio “Erro”, que é recebido pelo sprite “Recado”.

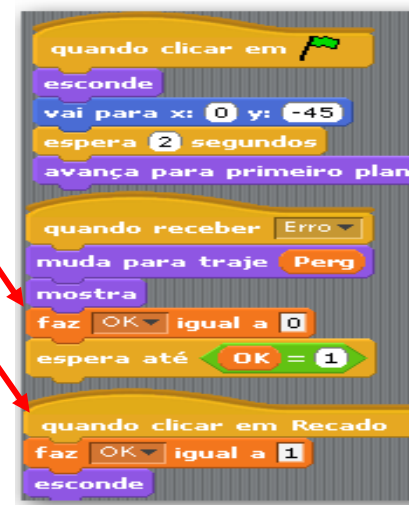
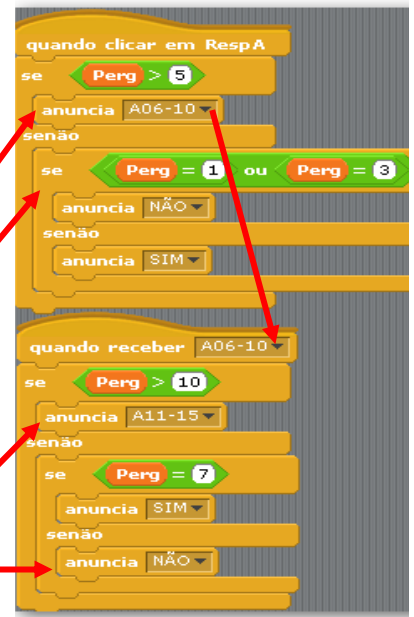
“Recado” tem estado escondido desde o início; e até esperou 2 segundos depois do clique na bandeira, para que todos os sprites se colocassem e ele pudesse ficar em primeiro plano, evitando assim que os sprites resposta sejam clicados mais de uma vez, já que lhes fica sobreposto.

Assim que recebe o anúncio “Erro” muda para o traje com a explicação correspondente à pergunta corrente, faz a variável OK=0 para poder ficar à espera que ela passe a ter o valor 1.

Isso vem a acontecer quando, depois de ler a explicação dada, o utilizador clica sobre ela para a fazer desaparecer (é o “esconde”) e para que, fazendo OK=1 destrave o bloco do “Erro”.

Este, por sua vez, termina a execução, o que dá origem a que o bloco “NÃO” do “Gato” (que tinha ficado à espera por causa do comando “anuncia Erro e espera”) incremente “Perg” e termine.

Ora o Palco estava também à espera que “Perg” aumentasse para anunciar “Next”, se bem se lembram; e isto já parece um dominó a cair.



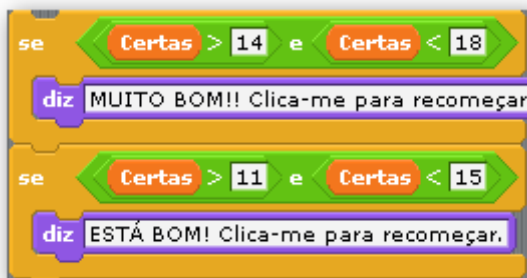
Finalmente, voltamos ao “Gato” para ver o que ele faz quando recebe “FIM”

(é o Palco que emite este anúncio, lembram-se?).

Pois o “Gato” dá uma resposta apropriada e classificativa conforme o número de respostas certas que foram contabilizadas. Repare-se como estes SEs aninhados se tornam complexos e compridos como o peixe-espada; e só havia 7 classes. Imaginem como ficaria o SE das 20 perguntas de que falámos acima ...

Mesmo assim, convém notar que o último teste “apanha” os resultados desde 3 a 5 e já não é preciso testar mais, pois os que ficam são os 2, 1 e 0, a que corresponde a classificação “Péssimo”.

Filosofando, poder-se-ia dizer que é escusado aninhar estes SEs pois também ficava bem colocar, seguidos, uns como estes:



(só dois para exemplificar)

Mas, já se vê, as condições complicaram-se para delimitar as classes sem ambiguidades; e, neste exemplo de Ses seguidos e não aninhados, quando uma delas é verdade, todas as condições seguintes são testadas desnecessariamente. Logo, a técnica de aninhar os SEs é melhor, (excepto em casos muito raros).

Os blocos do sprite “Máscara” são auto-explicativos, mas reproduzem-se aqui ao lado, para enfatizar necessidade de trazer o sprite para primeiro plano, sob pena de não tapar, as variáveis mas ficar tapado por elas.

Frederico

Nota: O projeto “EscreverPortuguês” que apoia este documento está publicado em

<http://kids.sapo.pt/scratch/projects/EduScratch/2587>

