MatemáTICas deportivas

Joaquín Aguilar Barriuso
Diego Alonso Santamaría
Elena Torres María

SUMA+ núm. 96
pp. 43-56
Artículo recibido en *Suma* en julio de 2019 y aceptado en enero de 2020

¿Para qué sirven las matemáticas? Esta es una de las preguntas que nos hacen habitualmente los alumnos. ¿No podemos hacer que esa pregunta deje de tener sentido enseñando las matemáticas de una manera práctica? Con este artículo pretendemos hacer ver las matemáticas relacionadas con elementos de nuestro mundo, en este caso con la lotería del *draft* de la NBA, lo que permite una introducción a la combinatoria y al cálculo de probabilidades y además acercarlas a los alumnos ayudados con las nuevas tecnologías, realizando aplicaciones para el móvil. Para ello hemos utilizado la aplicación AppInventor.

Palabras clave: Draft NBA, AppInventor, Combinatoria, Cálculo de probabilidades, Programación de *apps*.

Sport Maths and ICT // What use is maths? That is a question we get asked a lot by our students. Can we make that question disappear by teaching maths in a more practical way? This article aims to show how mathematics is used in the real world, in this case in the NBA Draft Lottery, with an introduction to combinatorics and the calculation of probabilities. It also strives to make mathematics more accessible through the use of new technologies by creating mobile applications using the APPInventor app.

Keywords: NBA Draft, AppInventor, Combinatorics, Calculation of probabilities, App programming.

El objetivo principal de este artículo es dar la importancia que se merece a una parte de la matemática como es la probabilidad que en muchas ocasiones es como la hermana pobre de todo el bagaje matemático que se ha ido acumulando a través de los siglos.

Es una disciplina relativamente reciente que se desarrolla de forma exponencial en el siglo XX aunque la historia que se cuenta de ella tiene que ver con los juego de azar (cartas, naipes…) a los que tenían mucha afición, allá por el siglo XVII, en Francia, Italia y otros países de la Europa de esa época.

En la enseñanza de las matemáticas se suele relegar la impartición de las lecciones de combinatoria y probabilidad al final del curso escolar aunque la mayoría de las veces, y siempre con la excusa de la falta de tiempo por tantas actividades complementarias y extraescolares a las que sometemos a los alumnos, no se explica y damos mucha más importancia a otras partes de las matemáticas como el álgebra, la geometría, el análisis, el cálculo y un largo etc.

Proponemos mostrar a nuestros alumnos una aplicación del mundo de la probabilidad a la vida real y, en concreto, desarrollarlo a través de uno de los grandes intereses de la juventud actual como es el deporte, presentando los contenidos de la mano del famoso *draft* de la NBA. Pero vamos más allá. Alejándonos de la educación tradicional y valiéndonos del uso de las nuevas tecnologías, pretendemos que los alumnos apliquen lo aprendido construyendo un artefacto digital, en este caso una aplicación para móvil (figura 1) realizada con Appinventor.

Matemáticas en la NBA

Vamos a estudiar las matemáticas que están inmersas en la NBA, que es la liga más importante de baloncesto que hay en el planeta. En concreto, vamos a hacer de la mano de la lotería del *draft*, una introducción a la combinatoria, al cálculo de probabilidades y a la toma de decisiones basadas en las matemáticas, es decir, cómo utiliza las matemáticas una empresa como la NBA para crear una liga más igualada y competitiva.

Pero empecemos desde el principio, vamos a dar unas pinceladas de cómo es el funcionamiento interno de la NBA y luego seguimos con el sorteo del *draft*.

EE.UU. es un país muy extenso y por eso los 30 equipos que componen la liga se dividen en dos grandes conferencias: la Conferencia Oeste y la Conferencia Este, cada una con 15 equipos. Aunque existe esta división, los equipos juegan todos contra todos en un total de 82 partidos en la liga regular. Sin embargo, a la hora de la clasificación, existe una por conferencia y al finalizar los partidos de la liga regular, se clasifican los 8 mejores equipos de cada conferencia para jugar unos *playoffs* por el título de la NBA. Para los 7 equipos restantes de cada conferencia, la temporada habrá llegado a su fin y tendrán una pequeña ventaja en el siguiente reparto de los jugadores jóvenes más talentosos, lo que llamamos el *draft*.

¿QUÉ ES EL *DRAFT*?

La NBA realiza cada año un sorteo o lotería en la que los peores equipos tienen un mayor número de posibilidades de obtener a los mejores jugadores jóvenes que ese año se convierten en profesionales.

El *draft* está dividido en dos grandes rondas:

Figura 1. Pantalla principal y otras pantallas de la App (Vídeos, Test y Combinatoria)

— La segunda ronda es muy sencilla pues los treinta equipos de la NBA realizan su elección en orden inverso a su clasificación, es decir, en orden inverso al número de victorias que consiguieron esa temporada.
— En la primera ronda, las 14 primeras elecciones del *draft* (los 14 mejores jugadores jóvenes) se sortean entre los 14 peores equipos. Para ello se reparte a cada uno de ellos un número concreto de papeletas o boletos en función de su posición en la liga. Esa cantidad concreta de papeletas ya está preestablecida por la NBA y en el sorteo de este año ha variado muy sustancialmente con respecto al reparto que se llevaba a cabo en los últimos años, y las razones se explicarán en el próximo apartado. Después de la elección de los 14 peores equipos, las elecciones del 15 al 30 se realizan por los equipos de *playoffs* en orden inverso a su clasificación. En consecuencia, en 2019, en los dos últimos puestos estuvieron los finalistas de la NBA, el equipo de los Golden State Warriors y el campeón de ese curso, el equipo de los Toronto Raptors (en el que militaban los españoles Marc Gasol y Sergi Ibaka) y que sería el último equipo que elegiría en la primera ronda.

La lotería y sus boletos

Cada boleto de la lotería contiene cuatro números, cada uno del uno al catorce, ordenados y sin repetirse. Un posible boleto puede estar formado por los números 4, 7, 10 y 14.

¿Cuántas papeletas existen? Imaginemos una urna en la que tenemos catorce bolas numeradas del 1 al 14. Elegimos una primera bola y miramos el número. Para esta primera bola tenemos un total de 14 posibilidades, pero en la segunda extracción ya solo hay 13 bolas, luego solo hay 13 posibilidades para este segundo número, 12 para el tercero y 11 para el cuarto, es decir, un total de $14\times 13\times 12\times 11=24\,024$ posibilidades diferentes. Pero debemos quedarnos solo con los boletos cuyos números están ordenados de menor a mayor. Fijémonos en la cuaterna de números iniciales 4, 7, 10 y 14. Esos cuatro números se pueden poner de un total de $4\times 3\times 2\times 1=24=4!$ formas o permutaciones diferentes. Luego si queremos quedarnos solo con los boletos ordenados de 4 números habrá un total de $24\,024/24=1\,001$. Parece magia, pero exactamente sale esa cifra tan redonda salvo que nos sobra uno para que fuese el millar. Bueno, hay una solución simple: se elimina el boleto cuyos números son 11, 12, 13 y 14 y asunto arreglado.

Como a los matemáticos nos gusta siempre definir las cosas para que no existan dudas, al número de formas diferentes de elegir 4 números diferentes de un total de 14 se les da el nombre de variaciones si se tiene en cuenta el orden y combinaciones en el caso de que el orden no se tenga en cuenta, y se denotan respectivamente por:

— *Variaciones sin repetición* de catorce elementos tomados de cuatro en cuatro:
$$V_{14,4}=24\,024$$
— *Combinaciones sin repetición* de catorce elementos tomados de cuatro en cuatro:
$$C_{14,4}=1\,001$$
— El otro número que ha salido se denominan las *permutaciones* de 4 elementos:
$$P_4=24=4!$$

En términos de factoriales se pueden calcular de la siguiente forma:

$$V_{14,4}=\frac{14!}{(14-4)!}=\frac{14!}{10!}=\cdots=24\,024$$
$$C_{14,4}=\frac{14!}{(14-4)\cdot!4!}=\frac{14!}{10!\cdot 4!}=\cdots=1\,001.$$

Indicaremos como primer equipo del *draft* E_1 al último clasificado, el segundo equipo E_2 al penúltimo clasificado, y así sucesivamente, indicando el equipo E_k al equipo clasificado en la posición $(30-k+1)$, siendo $k=1,2,\ldots,14$.

Estos catorce equipos van a recibir una cantidad de boletos de los indicados anteriormente. Depende de ese reparto la probabilidad de que en el sorteo cada uno de los equipos resulte premiado. En los *draft* de los años anteriores a 2019 el equipo que quedaba peor clasificado E_1 recibía 250 boletos, el penúltimo clasificado E_2 recibía 199 boletos, el antepenúltimo E_3 recibía 156 boletos y así iba decreciendo el número de boletos según se iba avanzando en la clasificación.

En términos de probabilidades el equipo peor clasificado E_1 tenía una probabilidad del 25% de que le tocase en el primer sorteo, 19,9% el equipo E_2 y el 15,6% el equipo E_3, y esos porcentajes iban disminuyendo hasta el equipo E_{14} con un 0,5% de probabilidad. Además, no solo se sorteaba la primera elección, sino que después se sorteaban la segunda y la tercera elección.

¿Qué ocurría?, y aquí viene la toma de decisiones basada en matemáticas. Pues que la NBA se dio cuenta que había equipos que perdían a posta para aumentar sus posibilidades en este *draft*. Es decir, equipos que buscaban la peor clasificación posible para recibir el mayor número de boletos para el sorteo y así tener más posibilidades de elegir a los mejores jugadores jóvenes de ese año.

Equipo (E_k)	P_k-18	P_k-19
E_1	0,250	0,140
E_2	0,199	0,140
E_3	0,156	0,140
E_4	0,119	0,125
E_5	0,088	0,105
E_6	0,063	0,090
E_7	0,043	0,075
E_8	0,028	0,060
E_9	0,017	0,045
E_{10}	0,011	0,030
E_{11}	0,008	0,020
E_{12}	0,007	0,015
E_{13}	0,006	0,010
E_{14}	0,005	0,005

Tabla 1. Probabilidades en el primer sorteo

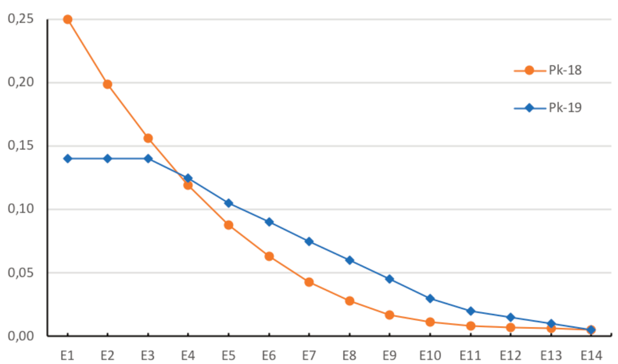


Figura 2. Representación gráfica de la tabla 1

Esta estrategia tan ruin recibe el nombre de *tanking* y, aunque está denunciado y perseguido por el comisionado de la NBA, los equipos lo utilizaban de forma clamorosa y hay numerosos ejemplos que confirman esta mala práctica.

Esto ha motivado que a partir de 2019 se haya modificado de forma importante la asignación de papeletas y así los tres peores equipos clasificado, E_1, E_2 y E_3, han recibido 140 boletos cada uno de ellos y el resto del reparto ha sido más uniforme, por lo que las posibilidades de cada equipo se han modificado de forma importante. Además, ya no solo se sortean las tres primeras elecciones, sino que hay cuatro sorteos.

En la tabla 1 aparecen las probabilidades de cada uno de los equipos en el primer sorteo, tanto en el *draft* de 2018 como en el de 2019. La figura 2 es una representación de estos valores.

DRAFT Y PROBABILIDADES: SEGUNDO SORTEO

El número de boletos que va a recibir cada equipo E_k lo vamos a indicar por B_k y por P_k a la probabilidad de que en el primer sorteo de la primera ronda del *draft* le toque la lotería al equipo E_k. Usaremos la notación $E_k^{(m)}$ para designar el suceso «al equipo E_k le ha tocado el m-ésimo sorteo de la primera ronda».

Según esto, el valor de P_k es muy fácil de calcular ya que es el cociente entre el número de boletos B_k y el número total de boletos repartidos que es 1 000.

Se realiza el primer sorteo de la primera ronda y le toca a un equipo, dicho equipo tiene que esperar a la segunda ronda para volver a elegir un segundo jugador. Pero hay un segundo sorteo en esta primera ronda, la pregunta es ¿qué probabilidad tiene ahora cada equipo para elegir en este segundo sorteo?

Como el equipo premiado en el primer sorteo ya no entra en este segundo sorteo, sus papeletas se eliminan y no entran en liza, y el número de papeletas es ahora $(1\,000 - B_i)$.

Designaremos por S_j las probabilidades teóricas a priori de cada uno de los catorce equipos E_j para elegir en este segundo turno. Para calcularla utilizaremos la probabilidad total, que es la suma de las probabilidades de los sucesos $E_i^{(1)} \cap E_j^{(2)}$ para todo $i \neq j$, en los que suponemos que el equipo que ha hecho la primera elección es E_i.

Vamos a calcular la probabilidad de que en el segundo sorteo le toque al equipo E_1 sumando las probabilidades de los sucesos $E_i^{(1)} \cap E_1^{(2)}$ para todo $i \neq 1$, suponiendo que en el primer sorteo el equipo premiado es el E_i, cuya probabilidad es P_i y luego hay que multiplicar por la probabilidad de que en el segundo sorteo el premiado sea el equipo primero, esta probabilidad es $P_i \times B_1 / (1\,000 - B_i)$. Podemos ver el cálculo de las probabilidades de estos sucesos para el *draft* de 2018 y el *draft* de 2019 en las tablas 2 y 3 respectivamente.

Utilizando los resultados de la tabla 3, para el cálculo de S_1, en el *draft* de 2019 obtenemos que:

$$S_1 = p\left(E_1^{(2)}\right) = \sum_{i=2}^{n} p\left(E_i^{(1)} \cap E_1^{(2)}\right) =$$
$$= \sum_{i=2}^{n} p\left(E_i^{(1)}\right) p\left(E_1^{(2)} / E_i^{(1)}\right) = 0{,}134173189.$$

Y de forma similar se haría para el *draft* de 2018 usando los valores de la tabla 2 y obteniendo:

$$S_1 = p\left(E_1^{(2)}\right) = 0{,}2151011632.$$

La tabla 4 contiene las probabilidades de cada equipo en el segundo sorteo y la figura 2 su representación tanto en el caso del *draft* de 2018 como en el de 2019.

*Draft*18
$p\left(E_2^{(1)} \cap E_1^{(2)}\right) = 0{,}199 \times \frac{250}{801} = 0{,}0621098626$
$p\left(E_3^{(1)} \cap E_1^{(2)}\right) = 0{,}156 \times \frac{250}{844} = 0{,}0462085308$
$p\left(E_4^{(1)} \cap E_1^{(2)}\right) = 0{,}119 \times \frac{250}{881} = 0{,}0337684449$
$p\left(E_5^{(1)} \cap E_1^{(2)}\right) = 0{,}088 \times \frac{250}{912} = 0{,}0241228070$
$p\left(E_6^{(1)} \cap E_1^{(2)}\right) = 0{,}063 \times \frac{250}{937} = 0{,}0168089647$
$p\left(E_7^{(1)} \cap E_1^{(2)}\right) = 0{,}043 \times \frac{250}{957} = 0{,}0112330198$
$p\left(E_8^{(1)} \cap E_1^{(2)}\right) = 0{,}028 \times \frac{250}{972} = 0{,}0072016461$
$p\left(E_9^{(1)} \cap E_1^{(2)}\right) = 0{,}017 \times \frac{250}{983} = 0{,}0043234995$
$p\left(E_{10}^{(1)} \cap E_1^{(2)}\right) = 0{,}011 \times \frac{250}{989} = 0{,}0027805864$
$p\left(E_{11}^{(1)} \cap E_1^{(2)}\right) = 0{,}008 \times \frac{250}{992} = 0{,}0020161290$
$p\left(E_{12}^{(1)} \cap E_1^{(2)}\right) = 0{,}007 \times \frac{250}{993} = 0{,}0017623363$
$p\left(E_{13}^{(1)} \cap E_1^{(2)}\right) = 0{,}006 \times \frac{250}{994} = 0{,}0015090543$
$p\left(E_{14}^{(1)} \cap E_1^{(2)}\right) = 0{,}005 \times \frac{250}{995} = 0{,}0012562814$

Tabla 2

*Draft*19
$p\left(E_2^{(1)} \cap E_1^{(2)}\right) = 0{,}140 \times \frac{140}{860} = 0{,}0227906976$
$p\left(E_3^{(1)} \cap E_1^{(2)}\right) = 0{,}140 \times \frac{140}{860} = 0{,}0227906976$
$p\left(E_4^{(1)} \cap E_1^{(2)}\right) = 0{,}125 \times \frac{140}{875} = 0{,}0200000000$
$p\left(E_5^{(1)} \cap E_1^{(2)}\right) = 0{,}105 \times \frac{140}{895} = 0{,}0164245810$
$p\left(E_6^{(1)} \cap E_1^{(2)}\right) = 0{,}090 \times \frac{140}{910} = 0{,}0138461538$
$p\left(E_7^{(1)} \cap E_1^{(2)}\right) = 0{,}075 \times \frac{140}{925} = 0{,}0113513513$
$p\left(E_8^{(1)} \cap E_1^{(2)}\right) = 0{,}060 \times \frac{140}{940} = 0{,}0089361702$
$p\left(E_9^{(1)} \cap E_1^{(2)}\right) = 0{,}045 \times \frac{140}{955} = 0{,}0065968586$
$p\left(E_{10}^{(1)} \cap E_1^{(2)}\right) = 0{,}030 \times \frac{140}{970} = 0{,}0043298969$
$p\left(E_{11}^{(1)} \cap E_1^{(2)}\right) = 0{,}020 \times \frac{140}{980} = 0{,}0028571428$
$p\left(E_{12}^{(1)} \cap E_1^{(2)}\right) = 0{,}015 \times \frac{140}{985} = 0{,}0021319796$
$p\left(E_{13}^{(1)} \cap E_1^{(2)}\right) = 0{,}010 \times \frac{140}{990} = 0{,}0014141414$
$p\left(E_{14}^{(1)} \cap E_1^{(2)}\right) = 0{,}005 \times \frac{140}{995} = 0{,}0007035175$

Tabla 3

TERCER SORTEO

En el tercer sorteo de la primera ronda, para averiguar $T_k = p(E_k^{(3)})$, la probabilidad de que el k-ésimo equipo no resulte afortunado hasta esta fase del sorteo, hay que calcular mucho más porque hay que sumar la probabilidad de cada una de las $13 \times 12 = 156$ posibilidades de que esto ocurra.

Asi, obtenemos, por ejemplo que:

$$T_1 = p\left(E_1^{(3)}\right) = \begin{cases} 0{,}1274865239 & \text{en } 2019 \\ 0{,}177734929 & \text{en } 2018 \end{cases}.$$

Utilizando una hoja de cálculo se pueden calcular todas estas probabilidades. En la tabla 5 aparecen, redondeadas a 4 cifras decimales, todas ellas.

EL CUARTO SORTEO DEL DRAFT 2019

Hasta el año 2018 solo se sorteaban los tres primeros equipos que elegían, y a continuación los equipos que no habían salido premiados lo hacían en orden inverso a su puesto en la clasificación. A partir del *draft* de 2019 se sortean cuatro equipos por lo que la probabilidad de que a un equipo le toque la suerte en este último sorteo es tremendamente complicada de calcular puesto que hay que calcular la escalofriante cantidad de $13 \times 12 \times 11 = 1\,716$ sucesos. Para determinar cada una de dichas probabilidades hay múltiples combinaciones. Este cálculo tan sofisticado requiere de un buen programa con un lenguaje de programación que utilice bucles y que explore todas las posibilidades.

Equipo (E_k)	S_k-18	S_k-19
E_1	0,2151	0,1342
E_2	0,1881	0,1342
E_3	0,1574	0,1342
E_4	0,1260	0,1223
E_5	0,0966	0,1054
E_6	0,0710	0,0920
E_7	0,0494	0,0780
E_8	0,0326	0,0634
E_9	0,0200	0,0483
E_{10}	0,0130	0,0327
E_{11}	0,0095	0,0220
E_{12}	0,0083	0,0166
E_{13}	0,0071	0,0111
E_{14}	0,0059	0,0056

Tabla 4. Probabilidades en el segundo sorteo

Equipo (E_k)	T_k-18	T_k-19
E_1	0,1777	0,1275
E_2	0,1712	0,1275
E_3	0,1559	0,1275
E_4	0,1330	0,1189
E_5	0,1068	0,1056
E_6	0,0812	0,0941
E_7	0,0579	0,0814
E_8	0,0389	0,0674
E_9	0,0241	0,0523
E_{10}	0,0158	0,0360
E_{11}	0,0115	0,0245
E_{12}	0,0101	0,0186
E_{13}	0,0087	0,0125
E_{14}	0,0073	0,0063

Tabla 5. Probabilidades en el tercer sorteo

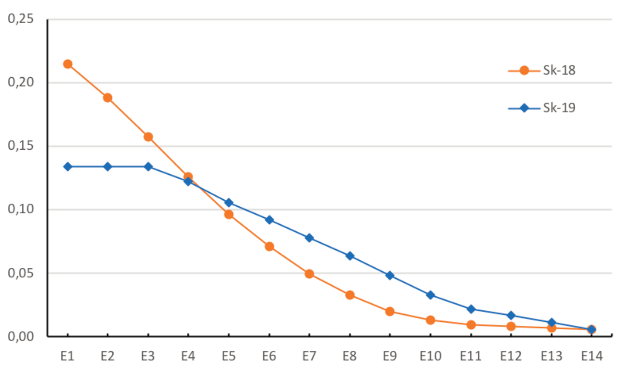


Figura 3. Representación gráfica de la tabla 4

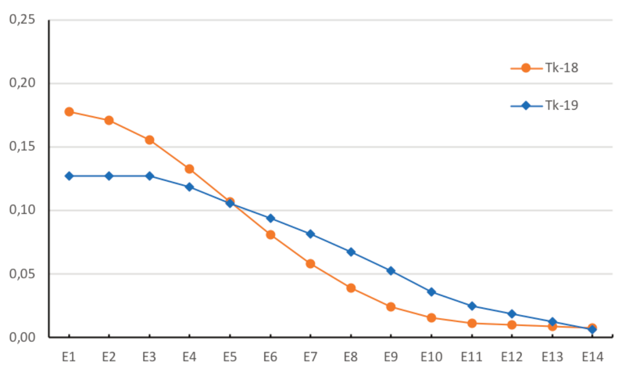


Figura 4. Representación gráfica de la tabla 5

En la figura 5 se muestra un programa en JavaScript que permite realizar el cálculo de las probabilidades de cada uno de los cuatro sorteos.

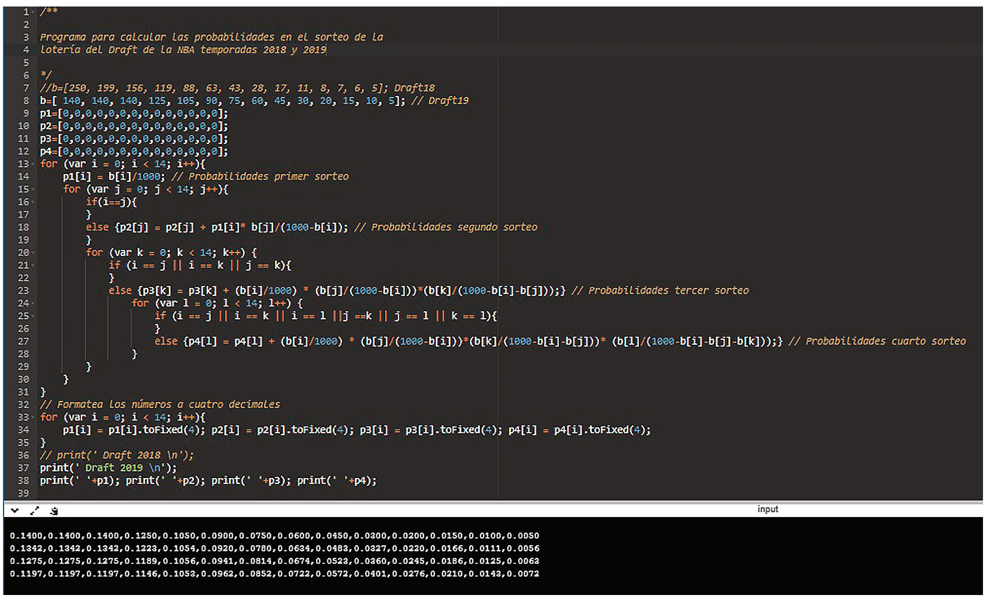
Por ejemplo, si el equipo E_1 tiene la mala fortuna de no haber salido elegido en ninguno de los tres primeros sorteos, este programa permite obtener que la probabilidad de que salga en este último sorteo, C_1, es:

$$C_1 = p\left(E_1^{(4)}\right) = 0{,}119720473.$$

Todos los valores de las probabilidades en el cuarro sorteo están recogidas en la tabla 6 y respresentadas en la figura 6.

COMPARATIVA DE LOS FORMATOS DE *DRAFT*

En la tabla 7, podemos ver los porcentajes correspondientes a los dos modelos de *draft* (en negrita el *draft* que entró en vigor desde 2019, y entre paréntesis el de años anteriores). Se han resaltado en rojo aquellas probabilidades que han disminuido más de un 10%,



```
/**

Programa para calcular las probabilidades en el sorteo de la
lotería del Draft de la NBA temporadas 2018 y 2019

*/
//b=[250, 199, 156, 119, 88, 63, 43, 28, 17, 11, 8, 7, 6, 5]; Draft18
b=[ 140, 140, 140, 125, 105, 90, 75, 60, 45, 30, 20, 15, 10, 5]; // Draft19
p1=[0,0,0,0,0,0,0,0,0,0,0,0,0,0];
p2=[0,0,0,0,0,0,0,0,0,0,0,0,0,0];
p3=[0,0,0,0,0,0,0,0,0,0,0,0,0,0];
p4=[0,0,0,0,0,0,0,0,0,0,0,0,0,0];
for (var i = 0; i < 14; i++){
    p1[i] = b[i]/1000; // Probabilidades primer sorteo
    for (var j = 0; j < 14; j++){
        if(i==j){
        }
        else {p2[j] = p2[j] + p1[i]* b[j]/(1000-b[i]); // Probabilidades segundo sorteo
        }
        for (var k = 0; k < 14; k++) {
            if (i == j || i == k || j == k){
            }
            else {p3[k] = p3[k] + (b[i]/1000) * (b[j]/(1000-b[i]))*(b[k]/(1000-b[i]-b[j]));} // Probabilidades tercer sorteo
                for (var l = 0; l < 14; l++) {
                    if (i == j || i == k || i == l ||j ==k || j == l || k == l){
                    }
                    else {p4[l] = p4[l] + (b[i]/1000) * (b[j]/(1000-b[i]))*(b[k]/(1000-b[i]-b[j]))* (b[l]/(1000-b[i]-b[j]-b[k]));} // Probabilidades cuarto sorteo
                }
        }
    }
}
// Formatea los números a cuatro decimales
for (var i = 0; i < 14; i++){
    p1[i] = p1[i].toFixed(4); p2[i] = p2[i].toFixed(4); p3[i] = p3[i].toFixed(4); p4[i] = p4[i].toFixed(4);
}
// print(' Draft 2018 \n');
print(' Draft 2019 \n');
print(' '+p1); print(' '+p2); print(' '+p3); print(' '+p4);

input

0.1400,0.1400,0.1400,0.1250,0.1050,0.0900,0.0750,0.0600,0.0450,0.0300,0.0200,0.0150,0.0100,0.0050
0.1342,0.1342,0.1342,0.1223,0.1054,0.0920,0.0780,0.0634,0.0483,0.0327,0.0220,0.0166,0.0111,0.0056
0.1275,0.1275,0.1275,0.1189,0.1056,0.0941,0.0814,0.0674,0.0523,0.0360,0.0245,0.0186,0.0125,0.0063
0.1197,0.1197,0.1197,0.1146,0.1053,0.0962,0.0852,0.0722,0.0572,0.0401,0.0276,0.0210,0.0143,0.0072
```

Figura 5. Código en JavaScript para el cálculo de cada uno de los sorteos

Equipo (E_k)	C_k-19
E_1	0,1197
E_2	0,1197
E_3	0,1197
E_4	0,1146
E_5	0,1053
E_6	0,0962
E_7	0,0852
E_8	0,0722
E_9	0,0572
E_{10}	0,0401
E_{11}	0,0276
E_{12}	0,0210
E_{13}	0,0143
E_{14}	0,0072

Tabla 6. Probabilidades en el cuarto sorteo

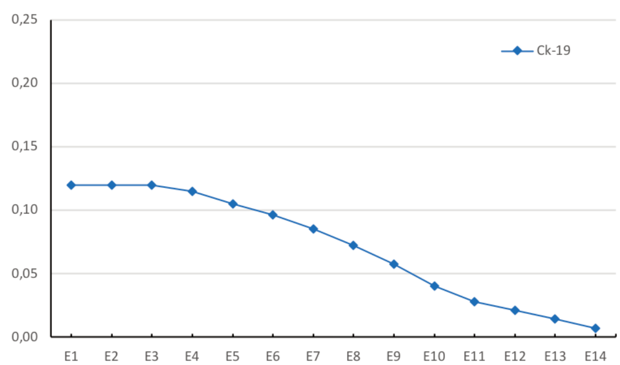


Figura 6. Representación gráfica de la tabla 6

y en naranja si han disminuido entre un 5% y un 10%. Además, se han remarcado en verde oscuro aquellas que han aumentado más de un 10%, y en verde claro si crecieron entre un 5% y un 10%.

Las conclusiones extraídas del estudio de la tabla 7 son claras: con el cambio de probabilidades, los últimos equipos han disminuido sus oportunidades de conseguir a los mejores jugadores. La táctica del *tanking* ya no ofrece una ventaja tal como plantearse apostar por una temporada llena de derrotas.

AppInventor

Una vez hemos explicado la importancia del cálculo de probabilidades en la NBA, podemos proponer a los alumnos el uso de un programa llamado AppInventor para que apliquen lo aprendido, y creen una aplicación para móvil que ayude a realizar estos cálculos. Pero primero, veamos qué es AppInventor.

Appinventor es un entorno de programación visual que permite a cualquier persona, incluyendo a los no familiarizados con la programación, crear aplicaciones para *smartphones* y tabletas con el sistema operativo Android, acercando a niños y mayores a la programación, pasando de ser meros usuario de aplicaciones a los creadores de las mismas.

El proceso de aprendizaje de lenguajes de programación tradicionales es mucho más lento que el de uso de bloques, con los cuales en menos de 30 minutos podemos tener una aplicación totalmente funcional.

Este entorno usa una interfaz gráfica parecida a Scratch, que permite arrastrar y soltar bloques de instrucciones de colores que hay que unir para que la aplicación realice la función deseada. De esta forma nos apartamos de la programación en texto donde debemos aprender y teclear unos comandos para cada una de las tareas a realizar.

A pesar de que no genera ningún código textual como Java, que se pueda descargar para un desarrollo más profundo y no sea tan flexible como este, permite realizar aplicaciones de una gran complejidad, de manera muy sencilla.

¿POR QUÉ APPINVENTOR?

MIT AppInventor es una gran herramienta para trabajar con las nuevas metodologías en el aula. Crear una aplicación móvil supone una gran motivación para los alumnos, ya que obtienen un producto final publicable, percibiendo el sentido de su esfuerzo, aprendizaje y trabajo.

Es gratuita, *online,* sin publicidad, no requiere instalación y es sencilla. Está ampliamente probada y el

Posición en la liga	Boletos en el draft	% de ganar la lotería del draft para elegir en las primeras 14 elecciones, a partir de 2019 (% de ganar la lotería del draft, desde 2005 a 2018)													
		1ª	2ª	3ª	4ª	5ª	6ª	7ª	8ª	9ª	10ª	11ª	12ª	13ª	14ª
30	140 (250)	14,0 (25,0)	13,4 (21,5)	12,7 (17,8)	12,0 (35,7)	47,9									
29	140 (199)	14,0 (19,9)	13,4 (18,8)	12,7 (17,1)	12,0 (31,9)	27,8 (12,3)	20,0								
28	140 (156)	14,0 (15,6)	13,4 (15,7)	12,7 (15,6)	12,0 (22,6)	14,8 (26,5)	26,0 (4,0)	7,0							
27	125 (119)	12,5 (11,9)	12,2 (12,6)	11,9 (13,3)	11,5 (0,9)	7,2 (35,1)	25,7 (16,0)	16,7 (1,2)	2,2						
26	105 (88)	10,5 (8,8)	10,5 (9,7)	10,6 (10,7)	10,5	2,2 (26,1)	19,6 (36,0)	26,7 (8,4)	8,7 (0,4)	0,6					
25	90 (63)	9,0 (6,3)	9.2 (7,1)	9,4 (8,1)	9,6		8,6 (43,9)	29,8 (30,5)	20,5 (4,0)	3,7 (0,1)	0,1				
24	75 (43)	7,5 (4,3)	7.8 (4,9)	8,1 (5,8)	8,5			19,7 (59,9)	34,1 (23,2)	12,9 (1,8)	1,3 (>0)	>0			
23	60 (28)	6,0 (2,8)	6.3 (3,3)	6,7 (3,9)	7,2				34,5 (72,4)	32,1 (16,8)	6,7 (0,8)	0,4 (>0)	>0		
22	45 (17)	4,5 (1,7)	4.8 (2,0)	5,2 (2,4)	5,7					50,7 (81,3)	25,9 (12,2)	3,0 (0,4)	0,1 (>0)	>0	
21	30 (11)	3,0 (1,1)	3.3 (1,3)	3,6 (1,6)	4,0						65,9 (87,0)	19,0 (8,9)	1,2 (0,2)	>0 (>0)	>0
20	20 (8)	2,0 (0,8)	2.2 (0,9)	2,4 (1,2)	2,8							77,6 (90,7)	12,6 (6,3)	0,4 (0,1)	>0 (>0)
19	15 (7)	1,5 (0,7)	1.7 (0,8)	1,9 (1,0)	2,1								86,1 (93,5)	6,7 (3,9)	0,1 (>0)
18	10 (6)	1,0 (0,6)	1.1 (0,7)	1,2 (0,9)	1,4									92,9 (96,0)	2,3 (1,8)
17	5 (5)	0,5 (0,5)	0.6 (0,6)	0,6 (0,7)	0,7										97,6 (98,2)

Con la nueva regla, los peores equipos disminuyen sus oportunidades de ganar

Hay más equipos con opciones de ganar sin tener que perder partidos

Tabla 7. Comparativa de probabilidades entre los dos modelos de *draft* tiempo para realizar una actividad es corto. Una vez que se familiarizan con la herramienta se pueden realizar aplicaciones sencillas en una o dos sesiones. Las aplicaciones que se pueden crear pueden ser muy útiles y personalizadas, y son escalables, podemos incrementar el número de pantallas o la funcionalidad de la misma.

De esta forma, tenemos la oportunidad de trabajar algunas de las competencias en el aula: se fomenta la creatividad, las competencias sociales y cívicas a la hora de realizar el trabajo en grupo, la competencia digital, competencia en ciencias y tecnología, aprender a aprender, sentido de iniciativa y espíritu emprendedor y comunicación lingüística.

Una vez que se familiarizan con la herramienta se pueden realizar aplicaciones sencillas en una o dos sesiones.

Podemos trabajar de dos formas diferentes:

- Creación guiada de *apps*: se muestra a los alumnos un tutorial con los pasos a seguir para realizar una pequeña aplicación de modo que los alumnos aprendan el uso de la herramienta.
- Proyecto: una vez que los alumnos están familiarizados con la aplicación se les propone la realización de una *app* propia. Aquí es donde los alumnos pueden ser creativos y echar mano de su imaginación.

UN POCO DE HISTORIA

Creada inicialmente por Hal Abelson, profesor de Ingeniería y Ciencias de la Computación del MIT. Abelson se tomó un año sabático y se fue a colaborar con Google Labs, su famoso laboratorio de ideas. Se basó en la tesis de fin de máster de Ricarose Roque, que había realizado una versión general de la interfaz del director del MIT, Eric Klopfer, llamada StarLogo, y creó un sistema visual e intuitivo para desarrollar aplicaciones móviles de manera sencilla y rápida.

StarLogo empezó siendo la tesis de Mitchel Resnick, que dirigía el MIT's Media Arts and Science Program, y cuyos tutores fueron Abelson y Seymour Papert, un pionero de la educación computacional. Papert inventó Logo, un lenguaje de programación simple diseñado para introducir a los niños en las bases de la programación. Inicialmente Logo guiaría a un robot que incluye un lápiz en su estructura, creando un dibujo. Logo fue la base de Lego.

Ejemplo de programación con Logo
Dibujar un cuadrado de 70 x 70
REPITE 4[AVANZA 70 GIRADERECHA 90]

En 2010 se lanzó la primera versión, llamada Google APP Inventor, y un año más tarde liberaron el código, lo cedieron al MIT, e hicieron una aportación para la financiación inicial. El MIT continuó el proyecto desde la universidad, pasándose a llamar App Inventor Classic. Lo mejoraron y en 2013 sacaron una nueva versión más potente: MIT App Inventor.

En 2016 se introdujo una nueva versión que evita tener que instalar nada en el ordenador, ya que la prueba de aplicaciones se puede realizar de 3 maneras diferentes: conexión USB con el móvil o tablet, por wifi, o por medio del emulador.

¿CÓMO SE TRABAJA?

Consta de 4 fases:

Diseño de la aplicación

Para comenzar el proceso de creación necesitaremos únicamente un ordenador con conexión a internet, preferiblemente con un navegador como Google Chrome o Mozilla Firefox, y tener una cuenta de Google. Además, debemos tener claro qué queremos que realice nuestra aplicación y el diseño de las pantallas, podemos tener un boceto en papel si es necesario. Una vez que tengamos el croquis de nuestra aplicación, debemos crearla. Para ello tenemos la pantalla **Diseñador** (figura 7), que consta de varios apartados. En la **Paleta** tenemos todos los componentes que puede tener la aplicación. Basta con arrastrar un componente a la sección **Visor**. Veremos el componente en la sección **Componentes**, la cual tiene un botón para cambiar el nombre de estos. Se recomienda poner un nombre que indique la función del propio elemento. Posteriormente, en **Propiedades** podemos cambiar las características de nuestro componente: color, tamaño, etc. En la sección **Medios** podemos subir imágenes, sonidos, etc.

Creación de bloques

Una vez que tengamos el diseño de nuestra *app* creado, debemos darle funcionalidad, en la pantalla **Bloques**. En esta pantalla tenemos dos secciones, como se puede ver en la figura 8.

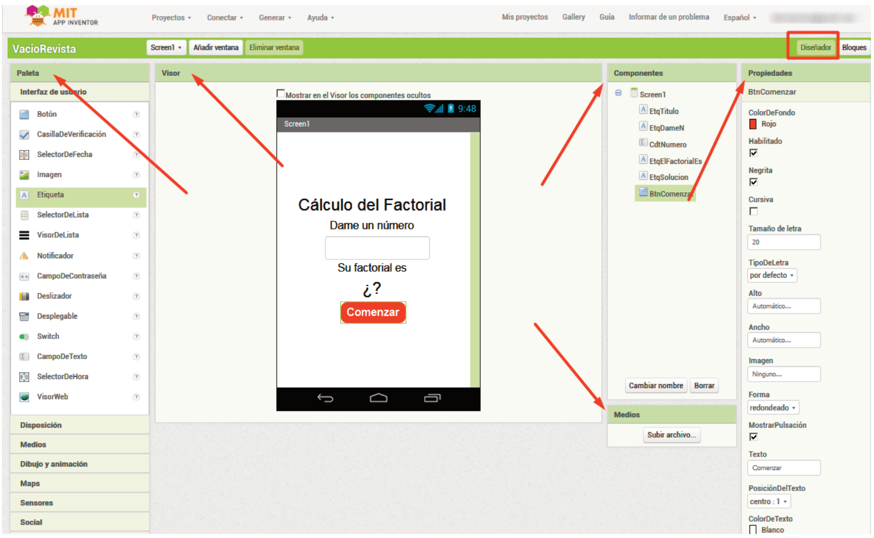


Figura 7. Pantalla Diseñador

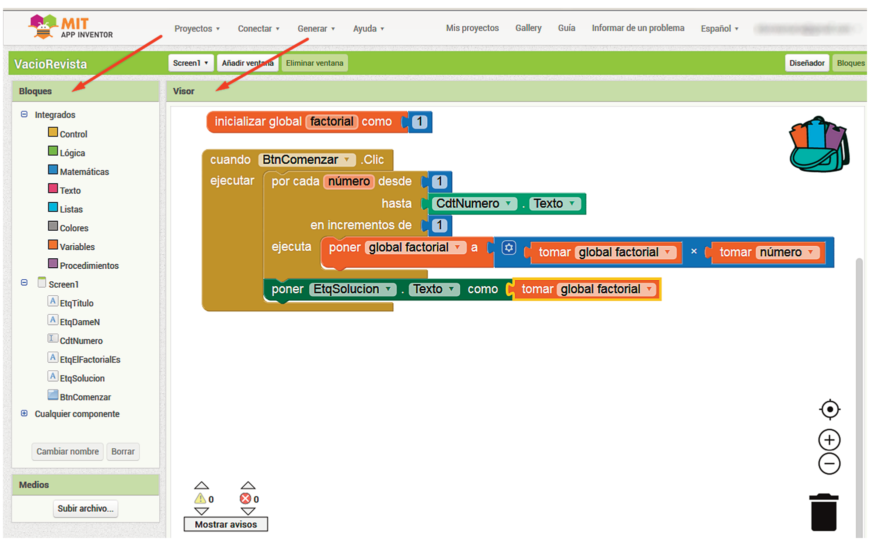


Figura 8. Pantalla Bloques

En la sección **Bloques** disponemos de los **Bloques Integrados**, que son aquellos comunes a cualquier aplicación. Debajo podemos encontrar los correspondientes a cada uno de los componentes (figura 9). Los bloques tienen diferentes formas y colores, que nos indican de qué tipo son y cómo podemos unirlos. Pinchando en cada uno de ellos podemos ver los bloques disponibles de cada componente o de cada tipo.

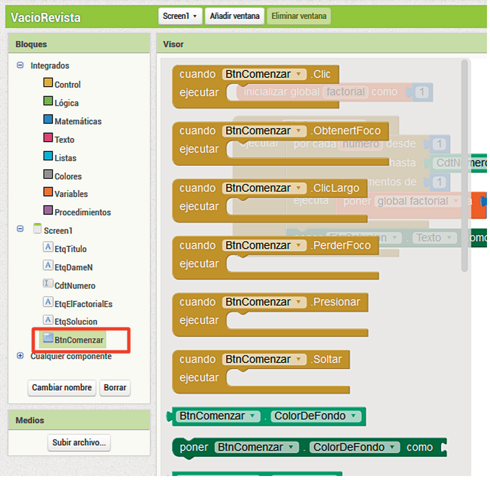


Figura 9. Bloques de cada componente

Realización de pruebas

Para probar nuestra aplicación la forma más sencilla y rápida es tener instalada la aplicación MIT AI2 Companion en el móvil.

Pulsamos en el menú **Conectar > AI Companion** (figura 10) y escaneamos el código QR (figura 11) que genera. También podemos usar el **Emulador** o conectar por **USB**.

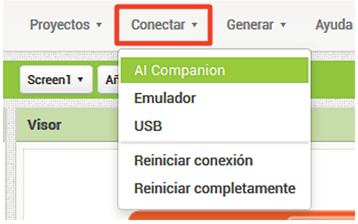


Figura 10. Menú Conectar

Figura 11. Código QR

Generación de la App

Una vez probada podemos generar la aplicación. Podemos generar un código QR o descargarla al ordenador (figura 12).

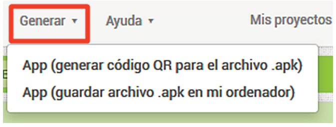


Figura 12. Generar *app*

UN EJEMPLO PRÁCTICO

Veamos ahora un ejemplo práctico de uso de AppInventor en el aula.

Una de las actividades más interesantes es que sean ellos mismos los que realicen una aplicación que calcule las combinaciones con o sin repetición.

Para ello, primero debemos hacer una aplicación que calcule el factorial de un número. Esto lo podemos realizar de forma guiada con ellos, para que posteriormente puedan ampliar la *app* con otras pantallas que calculen variaciones y combinaciones con o sin repetición.

Factorial

El diseño constará de seis elementos. De ellos, cuatro son etiquetas (figura 13):

Figura 13. Diseño de la *app* factorial y sus elementos

— *EtqTitulo*: donde pone «Cálculo del Factorial».
— *EtqDameN:* donde se muestra «Dame un número».
— *EtqElFactorialEs*: cuyo texto dice «Su factorial es».
— *EtqSolucion*: que en principio muestra dos interrogaciones, pero será donde se muestre el resultado del cálculo del factorial.
— *CdtNumero*: campo de texto donde el usuario puede introducir el número del cual se quiere conocer el factorial. Es importante que en las propiedades se indique que solo puedan introducirse números.
— *BtnCalcular*: botón que si se pulsa realiza el cálculo.

Conviene ahora hacer pensar a los alumnos, cómo programar el factorial de un número (figura 14).

Combinaciones

Ahora vamos a realizar un programa que calcule las combinaciones. Vamos a explicar solo los elementos que consideramos imprescindibles, empezando por el diseño de la pantalla (figura 15).

Primero realizaremos la programación solo de las combinaciones sin repetición, y dejaremos para la siguiente tarea el cálculo de las combinaciones con repetición.

Para ello, creamos cinco variables (figura 16).

Reutilizamos el cálculo del factorial de la práctica anterior para crear el procedimiento *factorial* (figura 17)

Usamos el procedimiento *factorial* para calcular las combinaciones de n elementos tomados de m en m (figura 18):

— Calculamos el factorial de n y asignamos el resultado a la variable *factorialN*.
— Calculamos el factorial de m y asignamos el resultado a la variable *factorialM*.
— Calculamos el factorial de $n-m$ y asignamos el resultado a la variable *factorialNmenosM*.
— Asignamos el resultado de la operación *factorialN* / (*factorialM* × *factorialNmenosM*) a la variable *combinaciones*.

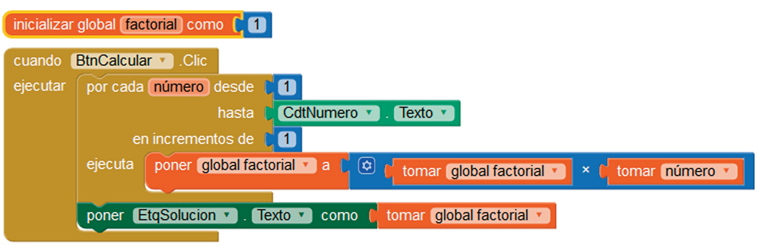


Figura 14. Programación del factorial por bloques

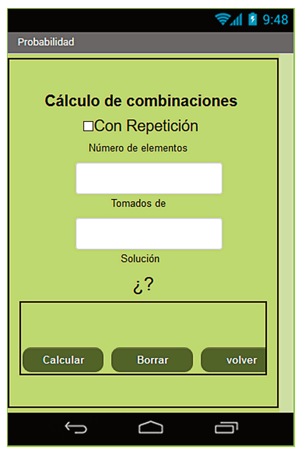


Figura 15. Diseño de la *app* combinaciones

Figura 16. Variables de la *app* combinaciones

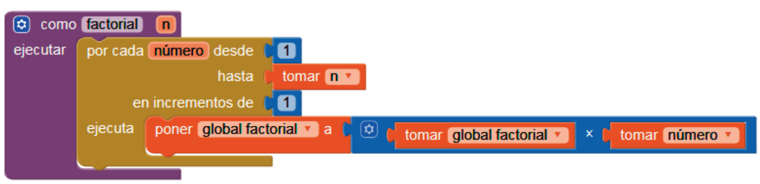


Figura 17. Procedimiento factorial. Programación por bloques

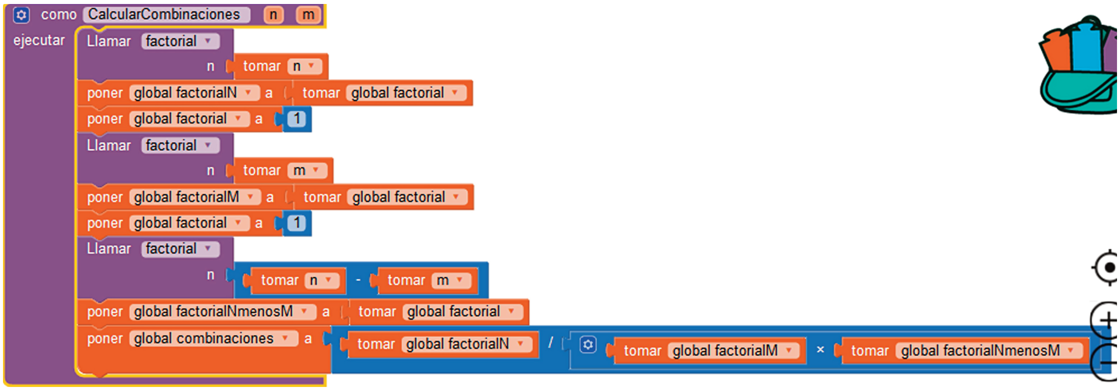


Figura 18. Programación del procedimiento para el cálculo de combinaciones

Ahora vamos a darle funcionalidad al botón *BtnCalcular* (figura 19):

— Primero inicializamos las variables.
— Después llamamos a *CalcularCombinaciones*.
— Por último vamos a modificar este botón para añadir la funcionalidad de cálculo de combinaciones con repetición. Para ello introducimos un condicional si-entonces-sino que está controlado por la variable *CdVConRepetición* (figura 20).

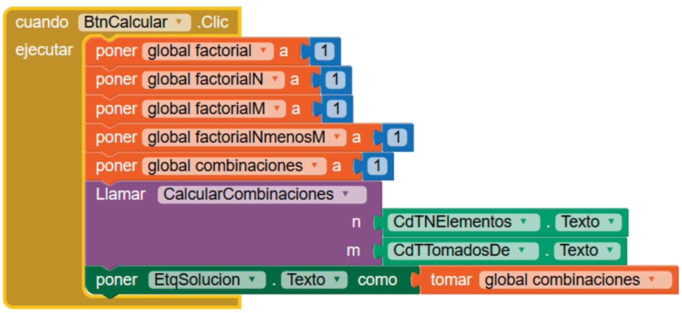


Figura 19. Programación del botón para el cálculo de combinaciones

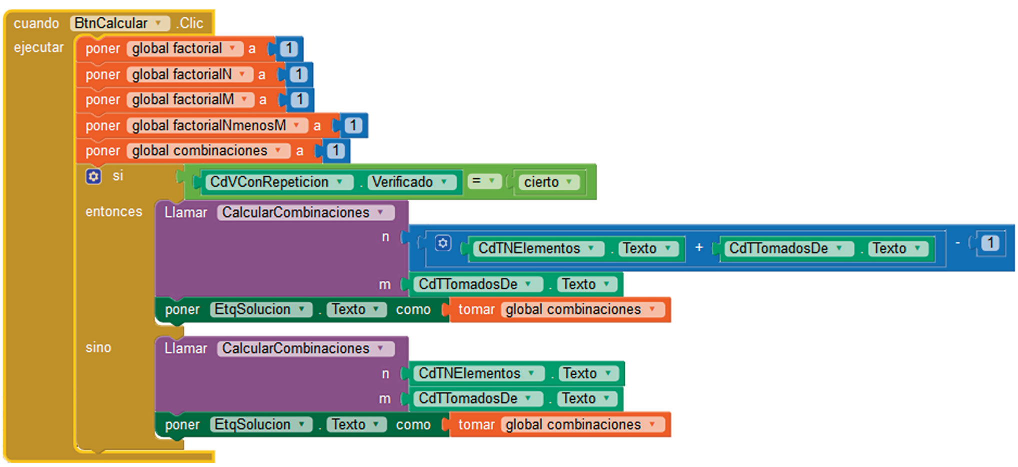


Figura 20. Programación del botón para el cálculo de combinaciones con repetición

Mejoras que pueden realizar los alumnos:

- — Evitar que se puedan introducir números negativos.
- — Evitar que se pueda meter un número *m* mayor que el *n*.
- — Evitar números decimales.
- — Añadir una pantalla para el cálculo de variaciones.
- — Añadir una pantalla con un test sobre el *draft* de la NBA y la combinatoria.

Nosotros hemos realizado una aplicación a modo de ejemplo. Esta aplicación puede pasársele a los alumnos o servir de ejemplo para que vean cómo debe quedar la suya. Tiene tres secciones:

- — La primera es un enlace a varios vídeos de YouTube que explican el *draft* de la NBA y la combinatoria asociada al mismo. Se pueden usar en el aula o pasar a los alumnos la aplicación.
- — La segunda parte consta de un test, que puede servir de modelo para que los alumnos realicen uno propio o para que lo realicen y comprueben su puntuación.
- — La tercera es una aplicación para el cálculo de combinaciones con o sin repetición.

Se puede ver y descargar la aplicación, incluidos los manuales, mediante el código QR que aparece abajo.

Referencias bibliográficas

Alonso, D. (2019a), «Las matemáticas del draft NBA», *Revista ZONA AZUL*, n.º 29, 26-27.

— (2019b), «Draft, tanking y matemáticas», *Revista ZONA AZUL*, n.º 30, 24-25.

Cabello, J. M. (2015), *Crea tus aplicaciones android con app inventor 2*, IC Editorial, Antequera, Málaga.

Hardesty, L.(2010), «The MIT roots of Google's new software», *MIT News*, <http://news.mit.edu/2010/android-abelson-0819>.

Huertas, J.I (2013), «Estrenamos versión 2 de App Inventor», *Programamos,* < https://programamos.es/estrenamos-version-de-app-inventor/>.

Posada, F. (2019), «Creando aplicaciones para móviles Android con MIT App Inventor 2», *INTEF*, <https://intef.es/wp-content/uploads/2019/03/MIT-App-Inventor-2.pdf>.

Stewart, I. (2016), *Números increíbles*, Planeta, Barcelona.

App Inventor for Educators, <http://teach.appinventor.mit.edu/>

MIT App Inventor, <https://appinventor.mit.edu/explore/about-us.html>.

<http://www.blogdebasket.com/tag/nba-draft-2019>.

<https://revistasuperlopez.lopezdemendoza.info/>.

Joaquín Aguilar Barriuso
IES Cardenal López de Mendoza (Burgos)
<joaquinaguilar@lopezdemendoza.es>

Diego Alonso Santamaría
IES Cardenal López de Mendoza (Burgos)
<diegoalonsosan@hotmail.com>

Elena Torres María
IES Cardenal López de Mendoza (Burgos)
<etorresmaria@hotmail.com>