

A continuación se muestran los pasos a seguir para construir el ejercicio:

1. Abra Descartes y cree una función *tiraDados()* sin argumentos.
2. Haga que la función guarde en una variable *D1* un número aleatorio entero entre 1 y 6 y en *D2* otro aleatorio entero entre 1 y 6.
3. Incluya una variable *suma* que guarde la suma de los valores de ambos dados.
4. Cree un vector *Tiro* con tamaño 11 (hay 11 posibles valores entre el tiro mínimo de 2 y el máximo de 12).
5. Cree un gráfico tipo texto en el que se muestre el valor de cada una de las entradas del vector.
6. Cree un botón *Lanzar* que active la función.
7. Haga que la función añada una unidad a la entrada (cajón) del vector que corresponde a la suma de los dados.
8. Haga que la función realice no sólo un tiro de dados, sino 100, cada vez que es llamada por el botón.

A continuación se presentan los resultados esperados de los pasos de realización de este ejercicio así como algunas observaciones.

1. La función se añade en el selector *Definiciones* del editor de configuraciones oprimiendo el botón + y seleccionando en el menú de la ventana emergente la opción *función*. En la misma ventana se introduce el nombre de la función. Note que la función lleva paréntesis vacíos al final, que indica que no tiene argumentos. Una vez añadida conviene dejar vacío el campo de texto después del signo = pues no se pretende que dicha función devuelva valor alguno, sólo que realice una serie de cálculos. No se visualizan cambios en el interactivo tras aplicar los cambios. Consulte el apartado sobre funciones en la documentación de Descartes si desea más información.
2. Dado que por el momento sólo se está considerando un tiro de los dos dados, conviene marcar el checkbox *algoritmo* de la función e introducir las instrucciones en el panel *hacer* de la misma. Las instrucciones son

```
D1=1+ent(6*rnd)
```

```
D2=1+ent(6*rnd)
```

Recuerde que *rnd* genera un valor aleatorio entre 0 (incluyendo el 0) y 1 (aproximándose mucho a él pero sin incluirlo). Como el valor mínimo de *rnd* es 0, *D1* y *D2* como mínimo pueden valer 1 (el tiro mínimo de un dado). Como el valor máximo de *rnd* es muy cercano a 1 pero sin incluirlo, *D1* y *D2* como máximo pueden valer $ent(1+5.999\dots)$, que como máximo da 6. Note que cualquiera de los posibles resultados tiene igual chance de salir ya que *rnd* tiene una distribución de probabilidad uniforme entre 0 y 1. Si desea más información sobre la variable *rnd*, consulte el apartado sobre *variables generales de Descartes* en la documentación.

3. Basta agregar dicha asignación al final de las instrucciones de la función, con lo que el panel *hacer* debe contener ahora

```
D1=1+ent(6*rnd)
```

```
D2=1+ent(6*rnd)
```

```
suma=D1+D2
```

No se visualiza algo nuevo tras aplicar los cambios.

4. El vector se añade pulsando el botón + en el selector *Definiciones*. En la ventana emergente es preciso seleccionar *vector* en el menú. En la misma ventana se le puede dar su nombre. Una vez añadido, se cambia su parámetro *tamaño* por 11 (que indica que tiene 11 entradas o cajones) y se limpia el contenido del panel de asignaciones del vector. Es importante que el vector quede al inicio de la lista en el panel izquierdo de *Definiciones* para que ya exista cuando la función lo llama. Esto se logra seleccionando el vector y oprimiendo el botón con la flecha hacia arriba que se encuentra sobre el panel, con lo que se moverá un lugar hacia arriba y quedará sobre la función.

Aún no se visualiza nada tras aplicar los cambios.

- El gráfico se añade en el selector *Gráficos*. Como se trata de una serie de datos a imprimir, podemos aprovechar la funcionalidad de *familia*. Así, primero es necesario marcar el checkbox *familia*, indicar que el intervalo va de 0 a 10 (un total de 11 valores) en *intervalo*, e indicar en *pasos* el valor de 10 (el número de pasos que hará para pasar de 0 hasta 11). Note que si va de 0 a 10 en 10 pasos, cada paso corresponde a un número entero. Esto es importante pues el valor del parámetro *s* usado debe llevar valores enteros, que son el índice del vector cuyas entradas se desea imprimir. El texto sugerido puede ser:

Han salido [Tiro[s]] tiros en los que la suma de dados es [s+2].
 Nótese que se agrega un +2 al valor de *s* para la suma de dados. Si *s* vale 0 (la cero-ésima entrada del vector), en esa entrada consideramos un tiro de 2 puntos. Es por ello que se coloca +2 como una forma de indicar el desfaseamiento del valor del vector respecto al valor de un tiro de dados. Como estamos manejando enteros, conviene colocar 0 en el campo *decimales* del gráfico *texto*. Para evitar que se amontonen los textos, hay que indicar en *expresión* que para cada valor de *s*, el nuevo renglón se recorra 16 pixeles hacia abajo. Es decir, en expresión conviene introducir
 (20,20+16*s)

Tras aplicar los cambios, el texto aparece. No obstante, no hay eventos registrados para los distintos resultados del tiro de dados. Ello se debe a que, aunque la función existe, aún no ha sido llamada.

- El botón se agrega pulsando el botón + en el selector *Controles*. En la ventana emergente se selecciona *botón* en el menú. En esa misma ventana se puede asignar su nombre. Una vez añadido el botón, el menú *acción* del mismo debe colocarse en la opción *calcular* y el parámetro de cálculo debe ser

tiraDados()

Tras aplicar los cambios aparece el botón en la parte sur del interactivo. No se observan cambios en el texto al oprimir el botón. Recuerde que el texto muestra los valores del vector, y hasta ahora no hay una asignación que cambie sus valores.

- Recordamos que un tiro de 2 corresponde a la entrada cero-ésima del vector, y un tiro de 12 corresponde a la décima entrada del vector. De tal forma que conviene añadir al final de las instrucciones de la función $Tiro[suma-2]=Tiro[suma-2]+1$, con lo que el contenido en *hacer* de la función es

```
D1=1+ent(6*rnd)
D2=1+ent(6*rnd)
suma=D1+D2
Tiro[suma-2]=Tiro[suma-2]+1
```

Digamos que el resultado de *suma* fue 2. *suma-2* corresponde a la entrada cero-ésima del vector. $Tiro[suma-2]$ corresponde al valor de dicha entrada. La asignación indica que el nuevo valor de $Tiro[suma-2]$ será su valor previo mas una unidad. De tal forma que esa entrada ahora valdrá una unidad más de lo que valía. Así pues, estamos realizando un conteo de cuántas veces sale cada tiro con dos dados.

Tras aplicar los cambios, el texto en el interactivo muestra originalmente cero eventos para todos los tiros. Pero si se oprime el botón, alguno de los valores cambia a 1. Y si se sigue oprimiendo, los textos van cambiando dependiendo del resultado del tiro de dos dados hecho.

- Para esto podemos aprovechar la funcionalidad de ciclo de la función. En su campo de texto *inicio* colocamos una variable *i* que nos servirá como contador. La inicializamos en 0. Es decir, el campo *inicio* de la función debe llevar

i=0

Como este contador debe incrementarse una unidad cada vez que se lanza el par de dados, al final de las instrucciones de la función hemos de añadir $i=i+1$, con lo que el panel *hacer* de la función queda

```
D1=1+ent(6*rnd)
D2=1+ent(6*rnd)
suma=D1+D2
i=i+1
```

```
Tiro[suma-2]=Tiro[suma-2]+1  
i=i+1
```

Finalmente, es necesario indicarle a la función cuándo detenerse. Esto se logra en el campo de texto *mientras*. Como queremos que lance 100 pares de dados, dicho campo debe contener

```
i<100
```

Tras aplicar los cambios, los contadores de cada resultado de los tiros en el texto del interactivo aparecen en cero. Pero al oprimir el botón, éstos aumentan su valor, indicando cuántas veces ha caído cada tiro resultante. Si se oprime el botón más veces, los contadores siguen creciendo. De hecho, cada vez que se oprime el botón se hacen 100 distintos tiros del par de dados.