## Realización del conteo dinámico

El conteo dinámico se realiza usando algunos de los métodos de dos clases llamadas ContD, MostrandoSTATUSDELEC y utilitaria.

Un fragmento de la clase ContD se muestra en el siguiente diagrama:

|  |
| --- |
| ContD |
| F : Frame  MB : MenuBar  TA : JTextArea  // . . .  CASILLA : ArrayList<Casilla>  // . . . |
| ContD() :  actionPerformed(e:ActionEvent):void  //. . . |

Diagrama de la clase ContD

Un fragmento de la clase MostrandoSTATUSDELEC se muestra en el siguiente diagrama:

|  |
| --- |
| MostrandoSTATUSDELEC |
| D : Dialog  // . . .  RutaAbsDFdC : String //RutaAbsolutaDeFiledeCasillas  RutaAbsDFdP : String // RutaAbsolutaDeFiledePREP |
| MostrandoSTATUSDELEC(file:File;f:Frame) :  MostrandoSTATUSDELEC(f:Frame) :  windowActivated(arg0:WindowEvent):void  windowClosed(arg0:WindowEvent):void  windowClosing(arg0:WindowEvent):void  windowDeactivated(arg0:WindowEvent):void  windowDeiconified(arg0:WindowEvent):void  windowIconified(arg0:WindowEvent):void  windowOpened(arg0:WindowEvent):void  actionPerformed(e:ActionEvent):void  getCasNExceso(Prom:double;VectorInt:Vector<Integer>):int  leerCasillas(path:String):int[]  leerCasillas(path:String;rowstoskip:int;columnstotake:int):int[]  leePREP(FdP:File):int[]  leePREP(path:String;rowstoskip:int):int[]  leePREP(MyBin:MBin[]):int[]  hacer\_prueba\_dbondad(C:String[][];intCount:int):boolYProm |

Diagrama de la clase MostrandoSTATUSDELEC

Un fragmento de la clase utilitaria se muestra en el siguiente diagrama:

|  |
| --- |
| utilitaria |
|  |
| conca(a:int[];b:int[]):String[][]  contar(k:int;arr:int[]):int  max\_incol1(arr:int[][]):int  print\_array(arr:int[]):void  print\_array(arr:Float):void  print\_array(arr:ArrayList<Float>):void  print\_array\_bid(arr:int[][]):void  print\_array\_dstr\_bid(arr:String[][]):void  reglaSturges(n:int):int  getLimSupInterv(Adotlength:int;N:int):ArrayList<Float>  conca(a:Float[];b:int[]):String[][]  conca(a:Float[];b:int[];Bin:MyBin[]):String[][]  totalDCasillasNBins(MBin:MyBin[]):int  showIds(MBin:MyBin[]):void  mostrar\_casilla\_yresult(ALdC:ArrayList<Casilla>;intNDCas:int;intInicio:int;intNDItems:int):void  show\_line(linea:String;intNumDL:int;intItemDInicio:int;intNDItems:int):void  show\_line(linea:String;intNumDL:int;intItemDInicio:int;  intNDItems:String[]):void  get\_intArray(line:String):int[]  contar\_votos(intNumDCoalicion:int;intIndexPartido:int;lineaH:char[];  linea:char[];partPt:Coalicion):int  get\_votos(intIndexPartido:int;lineaH:char[];linea:char[]):int  purify(charVotos:char[]):char[]  is\_digit(c:char):boolean  showCuentaDVotos(MBin:MyBin;intIndexPartido:int):void  CuentaDVotos(MBin:MyBin[];intIndexPartido:int):int |

Diagrama de la clase utilitaria

Específicamente, en el método actionPerformed(ActionEvent e) de la clase mostrandoSTATUSDELEC, en el siguiente fragmento de código se inicia con el proceso para la realización de un conteo dinámico:

public void actionPerformed(ActionEvent e){ // 4/5 ActionListener

//. . .

if(e.getSource().equals(JBconteodinamico)){ // 5/5 ActionListener

// Aquí se incluye el código para realizar un conteo dinámico

}

}// end actionPerformed(ActionEvent e)

//} else {

// System.out.printf(“NO SE CONTINUA CON EL CONTEO DINAMICO!\n”);

// System.out.printf(“DEBIDO A QUE QUEDAN UNICAMENTE %d CASILLAS!\n”,

// utilitaria.totalDCasillasNBins(MBin));

// }//end if(e.getSource().equals(JBconteodinamico)){// 5/5 ActionListener

El inicio del fragmento de código anterior se describe a continuación usando la notación de las tripletas de Hoare mencionadas en el apéndice HOARE.

### Tripleta 1

%FontSize=14
%TeXFontSize=14
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle\mbox{\tt e.getSource()=JBconteodinamico}\rangle
\]
\end{document}

%FontSize=14
%TeXFontSize=14
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\,\mbox{\tt if(e.getSource().equals(JBconteodinamico)) }\{
\]
\end{document}

%FontSize=14
%TeXFontSize=14
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle\mbox{\tt Se realiza un conteo din\'amico}\rangle
\]
\end{document}

Es decir, si el objeto del cual proviene un evento ActionEvent es el objeto JBconteodinamico de clase JButton (esto es, si se cumple la pre-condición de esta tripleta); entonces se procede a realizar un conteo dinámico. Si la pre-condición no se cumple (i.e., si el evento no proviene del JButton JBconteodinamico), entonces el evento debe provenir de algún otro objeto y no hay razón para realizar el conteo dinámico.

Para continuar, supondremos que la precondición %FontSize=10
%TeXFontSize=10
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle\mbox{\tt e.getSource()=JBconteodinamico}\rangle
\]
\end{document} es verdadera. En la realización del conteo dinámico se utilizan las variables

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
%int votosParaConteoDinDelCortePrep[];
\begin{verbatim}
int N;            //entero que se utiliza para almacenar la cantidad de
                  //clases a utilizar en las pruebas de bondad de ajuste 
int intCasNExceso;//cantidad de casillas en exceso presentes en una clase
ArrayList<Float> LSinter;//lista de n\'umeros reales que contendr\'a los
                  //l\'imites superiores de los intervalos de clases        
Object _lsinter[];//arreglo de objetos del mismo tama\~no que el arreglo A
Float lsinter[];  //arreglo que contiene l\'imites superiores de los
                  //intervalos de clases
String C[][];     //arreglo bidimensional que se utiliza para realizar las
                  //pruebas de bondad de ajuste
int A[];          //arreglo que contiene identificadores de todas las  
                  //casillas registradas en la elecci\'on
int B[];          //arreglo que contiene los identificadores de las
                  //casillas presentes en un corte PREP y para obtener 
                  //frecuencias observadas
\end{verbatim}
\end{document}

Lo primero que se hace, es llamar al método leerCasillas(), se llama a la versión que recibe un String y dos enteros:

### Tripleta 2

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt La variable de tipo String RutaAbsDFdC contiene la ruta}\\
\mbox{\tt absoluta del archivo de casillas}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
A = leerCasillas(RutaAbsDFdC, 7, 10);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El arreglo de enteros A contiene los identificadores (n\'umeros enteros)}\\
\mbox{\tt de todas las casillas utilizadas para la elecci\'on de que se trate}
\end{array}
\rangle
\]
\end{document}

Si se cumple la pre-condición, después de la ejecución del método leerCasillas() en el arreglo A se tendrán los identificadores de las casillas utilizadas en la elección que se está trabajando.

Después, se llama al método leePREP(), éste recibe un String y un entero:

### Tripleta 3

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt La variable de tipo String RutaAbsDFdP contiene la ruta}\\
\mbox{\tt absoluta de un corte PREP}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
B = leePREP(RutaAbsDFdP, 7);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El arreglo de enteros B contiene los identificadores (n\'umeros enteros)}\\
\mbox{\tt de todas las casillas presentes en el corte PREP de que se trate}
\end{array}
\rangle
\]
\end{document}

Si se cumple la pre-condición, después de la ejecución del método leePREP() en el arreglo B se tendrán los identificadores de las casillas presentes en el corte PREP que se está trabajando. A continuación, se llama al método reglaSturges(), pasándole como argumento la cantidad de elementos del arreglo B.

### Tripleta 4

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El tama\~no del arreglo de enteros B es la cantidad}\\
\mbox{\tt de casillas de un corte PREP}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
N = utilitaria.reglaSturges(B.length);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\mbox{\tt N = cantidad de clases a utilizar de acuerdo con la regla de Sturges}
\rangle
\]
\end{document}

Ahora se tiene en N, la cantidad de clases a utilizar para continuar con el procedimiento de conteo dinámico.

### Tripleta 5

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\mbox{\tt A.length}=\mbox{\tt Total de casillas en la elecci\'on}\,\wedge\,\mbox{\tt N}=\mbox{\tt Cantidad de clases a utilizar}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
%LSinter = 
%utilitaria.getLimSupInterv(ContD.CASILLA.size(),N);
\begin{verbatim}
LSinter = utilitaria.getLimSupInterv(A.length,N);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\mbox{\tt LSinter contiene los l\'imites superiores  de los intervalos de clase a utilizar}
\rangle
\]
\end{document}

Por construcción, la precondición de esta tripleta se cumplirá y en la lista LSinter se tendrán los límites superiores de los intervalos de clase a utilizar. Lo siguiente es pasar (por comodidad) los elementos de la lista LSinter a un arreglo de números reales de tamaño adecuado.

### Tripleta 6

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt LSinter contiene los l\'imites superiores  de los intervalos de clase}\\
\mbox{\tt a utilizar}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
_lsinter = LSinter.toArray();
lsinter = new Float[_lsinter.length];
for(int i=0;i<_linter.length;i++){
  lsinter[i]=(Float)_lsinter[i];
}
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El arreglo de n\'umeros reales lsinter contiene los l\'imites superiores de}\\
\mbox{\tt los intervalos de clase a utilizar en una prueba de bondad de ajuste}
\end{array}
\rangle
\]
\end{document}

Para la siguiente tripleta se utilizará una clase llamada MyBin, de la cual se presenta su diagrama a continuación:

|  |
| --- |
| MyBin |
| VectorDInt : Vector<Integer> |
| MyBin() : |

Diagrama de la clase MyBin

La siguiente tripleta indica la obtención de las frecuencias relativas de las casillas del corte PREP (cuyos identificadores están en el arreglo de enteros B) en los intervalos calculados anteriormente para la realización de pruebas de bondad de ajuste:

### Tripleta 7

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt lsinter contiene l\'imites superiores de intervalos}\,\wedge\,\\
\mbox{\tt B contiene identificadores de casillas del corte PREP}\,\wedge\,\\
\mbox{\tt MBin es un arreglo de objetos de clase  MyBin}\\
\mbox{\tt del mismo tama\~no que el arreglo 
lsinter}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
C = utilitaria.conca(lsinter,B,MBin);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt Para i=0,1,...,N-1; C[i][0] contiene los l\'imites inferior y superior de un}\\
\mbox{\tt intervalo de clase}\,\wedge\\
\mbox{\tt Para i=0,1,...,N-1; C[i][1] contiene las  frecuencias relativas de casillas del}\\
\mbox{\tt corte PREP en cada uno de los N intervalos para prueba de bondad de ajuste}\,\wedge\\
\mbox{\tt Cada elemento del arreglo MBin contiene una lista de los identificadores de }\\
\mbox{\tt casillas presentes en cada intervalo}
\end{array}
\rangle
\]
\end{document}

En la siguiente tripleta se utiliza un objeto de la clase boolYProm. La clase booYProm, cuyo diagrama se muestra a continuación, consta de solo dos atributos, un dato booleano y un número real:

|  |
| --- |
| boolYProm |
| ajuste : boolean  doubleProm : double |
|  |

Diagrama de la clase boolYProm

La clase boolYProm se utiliza para instanciar un objeto que nos da información sobre dos cosas; el dato booleano ajuste nos dice si las frecuencias relativas observadas en el corte PREP (en la lista de enteros de cada elemento del arreglo MBin) pasan la prueba de bondad de ajuste a una distribución uniforme con media igual al valor de la variable de tipo double doubleProm del mismo objeto.

### Tripleta 8

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[\langle
\begin{array}{l}
\mbox{\tt El arreglo bidimensional C contiene informaci\'on de los intervalos de}\\
\mbox{\tt clase y las frecuencias relativas observadas en los intervalos de clase}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
%boolYPromD=hacer_prueba_dbondad(C,CantidadDIntentos.intCount);
\begin{verbatim}
boolYProm boolYPromD=hacer_prueba_dbondad(C);
\end{verbatim}
%resultPBA=boolYPromD.ajuste;
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt boolYPromD tiene informaci\'on sobre si los frecuencias relativas}\\
\mbox{\tt pasaron la prueba de bondad de ajuste y la media aritm\'etica
de las}\\
\mbox{\tt frecuencias relativas}
\end{array}
\rangle
\]
\end{document}

En la siguiente tripleta se muestra el inicio de un ciclo while con el que se determina si es necesario y posible retirar algunas casillas del corte PREP para realizar nuevamente una prueba de bondad de ajuste:

### Tripleta 9

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt boolYPromD fue inicializado como se indica en la tripleta anterior}\,\wedge\\
\mbox{\tt los elementos del arreglo MBin contienen las listas de casillas del corte}\\
\mbox{\tt PREP en cada intervalo de clase}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
while((!boolYPromD.ajuste) && (utilitaria.totalDCasillasNBins(MBin) >= 30)) {
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El ciclo se ejecuta si los datos en MBin no pasaron la prueba de}\\
\mbox{\tt bondad de ajuste}\,\wedge\mbox{\tt en el corte PREP aun tenemos m\'as de 30 casillas}
\end{array}
\rangle
\]
\end{document}

Las siguientes tripletas corresponden a los casos en los que se ejecuta el ciclo while de la tripleta anterior.

La siguiente tripleta especifica el método getCasNExceso():

### Tripleta 10

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\verb VectorInt.size() > 
\mbox{\tt Prom}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
private int getCasNExceso(double  Prom,Vector<Integer> VectorInt){
  return (int)Math.floor(VectorInt.size()-Prom);
}
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt Devuelve la cantidad de casillas en exceso en el vector de}\\
\mbox{\tt enteros VectorInt por arriba del n\'umero real Prom}
\end{array}
\rangle
\]
\end{document}

En la siguiente tripleta se eliminan aleatoriamente casillas en exceso (por arriba del promedio de las frecuencias relativas observadas) de los elementos del arreglo MBin.

### Tripleta 11

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt Los elementos del arreglo MBin contienen las listas de las casillas}\\
\mbox{\tt del corte PREP en cada intervalo de clase}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
Random random=new Random();
for(int i=0;i<MBin.length;i++){
  if(MBin[i].VectorDInt.size() > boolYPromD.doubleProm) {
    intCasNExceso = getCasNExceso(boolYPromD.doubleProm,MBin[i].VectorDInt);
    while((intCasNExceso--) > 0) {
      MBin[i].VectorDInt.remove(random.nextInt(MBin[i].VectorDInt.size()-1));
    }//end while() interno
  }//end if()
}//end for() 
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\mbox{\tt Para i=1,2,...,N-1; MBin[i].VectorDInt.size() <= boolYPromD.doubleProm+1}
\rangle
\]
\end{document}

Ahora que ya se han quitado las casillas en exceso, se debe redefinir el número de clases, y para ello se debe encontrar y recalcular la cantidad de casillas que quedan del corte PREP:

### Tripleta 12

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt Para i=1,2,...,N-1;}\\
\mbox{\tt MBin[i].VectorDInt.size() <= boolYPromD.doubleProm+1}
\end{array}
\rangle
\]
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\begin{verbatim}
B = leePREP(MBin);
\end{verbatim}
\end{document}

%FontSize=12
%TeXFontSize=12
\documentclass{article}
\pagestyle{empty}
\begin{document}
\[
\langle
\begin{array}{l}
\mbox{\tt El arreglo de enteros B contiene los identificadores (n\'umeros enteros)}\\
\mbox{\tt de las casillas presentes en el corte PREP despu\'es de quitar las}\\
\mbox{\tt casillas en exceso}
\end{array}
\rangle
\]
\end{document}

Las siguientes tripletas del proceso de obtención de un conteo dinámico corresponden a tripletas mencionadas con anterioridad por lo que solo se anota una referencia en la columna de la derecha. Las tripletas de la siguiente tabla forman parte del ciclo while de la tripleta 9

|  |  |
| --- | --- |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} N = utilitaria.reglaSturges(B.length); \end{verbatim} \end{document} | Tripleta 4 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} %LSinter =  %utilitaria.getLimSupInterv(ContD.CASILLA.size(),N); \begin{verbatim} LSinter = utilitaria.getLimSupInterv(A.length,N); \end{verbatim} \end{document} | Tripleta 5 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} _lsinter = LSinter.toArray(); lsinter = new Float[_lsinter.length]; for(int i=0;i<_linter.length;i++){   lsinter[i]=(Float)_lsinter[i]; } \end{verbatim} \end{document} | Tripleta 6 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} C = utilitaria.conca(lsinter,B,MBin); \end{verbatim} \end{document} | Tripleta 7 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} %boolYPromD=hacer_prueba_dbondad(C,CantidadDIntentos.intCount); \begin{verbatim} boolYProm boolYPromD=hacer_prueba_dbondad(C); \end{verbatim} %resultPBA=boolYPromD.ajuste; \end{document} | Tripleta 8 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} while((!boolYPromD.ajuste) && (utilitaria.totalDCasillasNBins(MBin) >= 30)) { \end{verbatim} \end{document} | Tripleta 9 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} Random random=new Random(); for(int i=0;i<MBin.length;i++){   if(MBin[i].VectorDInt.size() >      boolYPromD.doubleProm) {     intCasNExceso=     getCasNExceso(boolYPromD.doubleProm,     MBin[i].VectorDInt);     while((intCasNExceso--) > 0) {       MBin[i].VectorDInt.remove(       random.nextInt(       MBin[i].VectorDInt.size()-1));     }//end while()   }//end if() }//end for()  \end{verbatim} \end{document} | Tripleta 11 |
| %FontSize=12 %TeXFontSize=12 \documentclass{article} \pagestyle{empty} \begin{document} \begin{verbatim} B = leePREP(MBin); \end{verbatim} \end{document} | Tripleta 12 |
|  |  |