randintlist

Creating random integer number lists,
with multiple numbers or not,
sorted or not.

Version 0.1.4 – 05/05/2025

Cédric Pierquet
c pierquet – at – outlook . fr
https://forge.apps.education.fr/pierquetcedric/packages-latex

10 numbers, between 1 and 100, without repetition:

3,85,62,97,91,48,34,8,60,100

The 5th value is:

91

10 numbers, between 1 and 100, without multiples of 5:

99,11,57,4,84,71,67,83,74,3

The 9th value is:

74

15 numbers, between 1 and 20, with repetition:

3,9,15,11,19,10,11,13,8,7,18,13,4,6,11

The last value is:

11

6 sorted numbers, between 1 and 51, without repetition:

ascending : 18,23,38,39,41,44

descending : 47>25>24>12>10>6

[1] *The luarandom package do the same things, but with the obligation to compile with LuaLaTeX.*

[2] *The tuple package is so much better… but I keep* `randintlist`*, without new features…*

Contents

1 Loading, useful packages

In order to load randintlist, simply use:

```
\usepackage{randintlist}
```

Loaded packages are ifthen, simplekv, listofitems, randomlist, xintexpr and xstring.

2 The Macros

2.1 Global usage

Package randintlist supports the creation of random integer number lists where a number will appear only once or multiple times. Generated lists can te used with listofitems.

All engines TEX are compatible with this package.

2.2 Generate the list

```
%generate list
\randintlist[keys]{\macro}
```

Available keys are:

- min: minimum value (default 1);
- max: maximum value (default 50);
- nb: number of values (default 6);
- sep: separator for the list (default ,);
- sort: sorting options, within no/asc/dec (default no);
- repeat: boolean to authorize repeating values (default false);
- exclude: list of excluded values (default empty);
- seed: random seed value according to used packages (default -).

```
%default values
\randintlist{\mylistA}\mylistA
```

18,11,21,17,24,26

```
%10 between 1 and 50, with ascending
\randintlist[sort=asc,min=1,max=50,nb=10]{\mylistB}\mylistB
```

8,20,22,28,37,41,42,46,49,50

```
%15 between 1 and 50, with ascending and repetitions allowed
\randintlist[sort=asc,min=1,max=50,nb=15,repeat]{\mylistC}\mylistC
```

1,1,3,8,18,20,23,26,29,37,40,42,47,48,50

```
%15 between 1 and 50, without multiples of 5
\randintlist[%
    sort=asc,min=1,max=50,nb=15,repeat,%
    exclude={5,10,15,20,25,30,35,40,45,50}]%
    {\mylistC}\mylistC
```

2,3,4,6,8,19,26,28,29,32,33,42,43,43,44

```
%list used with listofitems
\randintlist{\mylistD}\mylistD\par
\readlist*\mylistused{\mylistD}\showitems{\mylistused}\par
\mylistused[1]; \mylistused[-1]
```

50,17,20,33,49,13

$\boxed{50}\boxed{17}\boxed{20}\boxed{33}\boxed{49}\boxed{13}$

50; 13

2.3 Accessing elements

```
%accessing item
\getitemfromrandintlist[separator]{\macro}{index}[\macrores]
```

```
%with default keys
\randintlist{\mylistE}raw list: \mylistE\par
items list:\par
\xintFor* #1 in {\xintSeq{1}{6}}\do{\getitemfromrandintlist{\mylistE}{#1}\par}
first element: \getitemfromrandintlist{\mylistE}{1}\par
```

raw list: 22,49,39,1,25,41

items list:

22

49

39

1

25

41

first element: 22

```
\getitemfromrandintlist{\mylistE}{3}[\myres]%
third element: \myres
```

third element: 39

2.4 Version française

Voilà les commandes en version française, la syntaxe et les clés ne seront pas explicitées.

```
%obtenir la liste
\ListeRandint[Min=..,Max=..,Nb=..,Repet=..,Graine=..,Tri=..,Sep=..,Exclure=..]{\macro}

%extraire un élément
\ExtraireEltListeRandint[sep]{\macro}{position}[\macrores]
```

```
%liste
\ListeRandint[Min=5,Max=15,Nb=7,Repet,Tri=croiss,Sep={/}]{\maliste}\maliste\\
%élément
\ExtraireEltListeRandint[/]{\maliste}{4}
```

5/6/7/8/9/11/14
8

```
%liste
\ListeRandint[Min=50,Max=100,Nb=10,Repet,Tri=croiss]{\malisteB}\malisteB\\
%troisième élément
\ExtraireEltListeRandint{\malisteB}{3}[\montroisieme]%
troisième élément : \montroisieme
```

51,54,57,62,62,67,70,74,75,79
troisième élément : 57

3 Example

The following example uses Ti*k*Z, and comes from `luarandom`'s documentation.

```
\begin{tikzpicture}[scale=0.75]
    \randintlist[min=1,max=100,nb=100]{\mylistsquare}
    \draw[thin,gray] (0,0) grid (10,10) ;
    \foreach \i in {1,...,100}{%
        \xdef\tmpnumber{\getitemfromrandintlist{\mylistsquare}{\i}}%
        \xdef\tmpnumberrow{\xinteval{\xintiiRem{\i-1}{10}}}%
        \xdef\tmpnumbercol{\xinteval{\xintiiQuo{\i-1}{10}}}%
        \draw ({0.5+\tmpnumbercol},{0.5+\tmpnumberrow}) node {\tmpnumber} ;
    }%
\end{tikzpicture}
```

5	62	26	53	2	54	75	80	13	59
14	41	43	88	98	30	52	23	42	67
92	76	32	94	7	27	60	86	90	8
79	69	72	50	9	33	11	70	4	49
65	87	36	77	38	99	95	96	51	15
3	48	10	74	66	63	78	83	22	35
97	55	82	17	37	16	24	46	31	6
40	61	57	56	81	85	39	12	73	91
84	100	29	89	20	18	21	64	58	68
28	25	44	47	71	34	93	45	19	1

4 History

0.1.4: Bugfix

0.1.3: Bugfix

0.1.2: Changing name of internal macro

0.1.1: Possibility to exclude values

0.1.0: Initial version

5 The code

```
% Author     : C. Pierquet
% licence    : Released under the LaTeX Project Public License v1.3c or later, see
     http://www.latex-project.org/lppl.txt

\NeedsTeXFormat{LaTeX2e}
\ProvidesPackage{randintlist}[2025/05/05 0.1.4 Create a list of random numbers with or without multiple values]

%------History
% 0.1.4 Bugfix
% 0.1.3 new usage of extracting element (with storing \macro)
% 0.1.2 Changing name of macro
% 0.1.1 Possibility to exclude values
% 0.1.0 Initial version

%------Packages
\RequirePackage{simplekv}
\RequirePackage{listofitems}
\RequirePackage{randomlist}
\RequirePackage{xintexpr}
\RequirePackage{xstring}
\RequirePackage{ifthen}

%-----Macros (latex3) for sorting and seed
\ExplSyntaxOn
\cs_new_eq:NN \randintseed \sys_gset_rand_seed:n
\NewDocumentCommand\intascsortlist{m}
{
    \clist_sort:Nn #1
    {
        \fp_compare:nNnTF {##1} > {##2}
        { \sort_return_swapped: }
        { \sort_return_same: }
    }
}
\NewDocumentCommand\intdessortlist{m}
{
    \clist_sort:Nn #1
    {
        \fp_compare:nNnTF {##1} < {##2}
        { \sort_return_swapped: }
        { \sort_return_same: }
    }
}
\ExplSyntaxOff

%----Internal macro (latex2) for testing if element is in list
\newcommand\ifintvalueinlist[2]{\IfSubStr{,#2,}{,#1,}}

\newcommand\boolvalueinlist[2]{\IfSubStr{,#2,}{,#1,}{\def\resisinlist{1}}{\def\resisinlist{0}}}

\newcommand\testifintvalueinlist[4]{%
    \IfSubStr{,#2,}{,#1,}{\xdef\RESTMPVALUE{1}}{\xdef\RESTMPVALUE{0}}%
    \xintifboolexpr{ \RESTMPVALUE == 1}{#3}{#4}%
}

%----Macro for generating
\defKV[randomlistintegers]{%
    min=\def\TAEEmin{#1},%
    max=\def\TAEEmax{#1},%
    nb=\def\TAEEnb{#1},%
    sep=\def\TAEEsep{#1},%
    sort=\def\TAEEtri{#1},%
    seed=\def\TAEEseed{#1},%
    exclude=\def\TAEEexcluded{#1}
}

\setKVdefault[randomlistintegers]{%
    min=1,%
    max=50,%
    nb=6,%
    sep={,},%
    sort=no,%
    repeat=false,%
    seed={-},%
    exclude={}
}

\NewList{tmprandintlist}

\NewDocumentCommand\randintlist{ O{} m }{%1=keys,2=listname
    \useKVdefault[randomlistintegers]%
    \setKV[randomlistintegers]{#1}%
    \ifboolKV[randomlistintegers]{repeat}%repeat or not
        {%repeat allowed
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \randintseed{\TAEEseed}%
                }%
            %list creation of first element
            \def\resisinlist{1}%
            \whiledo{\resisinlist=1}{%
                \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
            }%
            \xdef#2{\tmpresrandint}%
            %list creation of other elements
            \xintFor* ##1 in {\xintSeq{2}{\TAEEnb}}%
                \do{%
                    \def\resisinlist{1}%
                    \whiledo{\resisinlist=1}{%
                        \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                        \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
                    }%
                \xdef#2{#2,\tmpresrandint}%
                }%
        }%
        {%no repeating
            %randomize numbers
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \RLsetrandomseed{\TAEEseed}%
                }%
            \ClearList{tmprandintlist}%clearing the list
            \xintFor* ##1 in {\xintSeq{\TAEEmin}{\TAEEmax}}%
                \do{%
                    \ifintvalueinlist{##1}{\TAEEexcluded}%
                        {}%
                        {%
                            \InsertRandomItem{tmprandintlist}{##1}%
                        }%
                }%
            %list creation (first then other)
            \xdef#2{\tmprandintlist[0]}%
            \xintFor* ##1 in {\xintSeq{1}{\TAEEnb-1}}%
                \do{%
                    \xdef#2{#2,\tmprandintlist[##1]}%
                }%
        }%
    %sorting
    \IfStrEq{\TAEEtri}{asc}%if ascending
        {\intascsortlist{#2}}%
        {}%
    \IfStrEq{\TAEEtri}{des}%if descending
        {\intdessortlist{#2}}%
        {}%
    \StrSubstitute{#2}{,}{\TAEEsep}[#2]%swipping separator if necessary
}

%-----Macro for extracting
\NewDocumentCommand\getitemfromrandintlist{ O{,} m m o }{%
    \IfStrEq{#1}{/}%
        {%
            \setsepchar[.]{#1}%
        }%
        {%
            \setsepchar{#1}%
        }%
    \readlist*\TMPLISTRANDINT{#2}%
    \IfNoValueTF{#4}{\TMPLISTRANDINT[#3]}{\itemtomacro\TMPLISTRANDINT[#3]#4}%
}

%-----french version
\defKV[randomlisteentiers]{%
    Min=\def\TAEEmin{#1},%
    Max=\def\TAEEmax{#1},%
    Nb=\def\TAEEnb{#1},%
    Sep=\def\TAEEsep{#1},%
    Tri=\def\TAEEtri{#1},%
    Graine=\def\TAEEseed{#1},%
    Exclure=\def\TAEEexcluded{#1}
}

\setKVdefault[randomlisteentiers]{%
    Min=1,%
    Max=50,%
    Nb=6,%
    Sep={,},%
    Tri=non,%
    Repet=false,%
    Graine={-},%
    Exclure={}
}

\NewDocumentCommand\ListeRandint{ O{} m }{%1=keys,2=listname
    \useKVdefault[randomlisteentiers]%
    \setKV[randomlisteentiers]{#1}%
    \ifboolKV[randomlisteentiers]{Repet}%repeat or not
        {%repeat allowed
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \randintseed{\TAEEseed}%
                }%
            %list creation of first element
            \def\resisinlist{1}%
            \whiledo{\resisinlist=1}{%
                \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
            }%
            \xdef#2{\tmpresrandint}%
            %list creation of other elements
            \xintFor* ##1 in {\xintSeq{2}{\TAEEnb}}%
                \do{%
                    \def\resisinlist{1}%
                    \whiledo{\resisinlist=1}{%
                        \xdef\tmpresrandint{\fpeval{randint(\TAEEmin,\TAEEmax)}}%
                        \boolvalueinlist{\tmpresrandint}{\TAEEexcluded}%
                    }%
                \xdef#2{#2,\tmpresrandint}%
                }%
        }%
        {%no repeating
            %randomize numbers
            \IfStrEq{\TAEEseed}{-}%
                {}%
                {%
                    \RLsetrandomseed{\TAEEseed}%
                }%
            \ClearList{tmprandintlist}%clearing the list
            \xintFor* ##1 in {\xintSeq{\TAEEmin}{\TAEEmax}}%
                \do{%
                    \ifintvalueinlist{##1}{\TAEEexcluded}%
                        {}%
                        {%
                            \InsertRandomItem{tmprandintlist}{##1}%
                        }%
                }%
            %list creation (first then other)
            \xdef#2{\tmprandintlist[0]}%
            \xintFor* ##1 in {\xintSeq{1}{\TAEEnb-1}}%
                \do{%
                    \xdef#2{#2,\tmprandintlist[##1]}%
                }%
        }%
    %sorting
    \IfStrEq{\TAEEtri}{croiss}%if ascending
        {\intascsortlist{#2}}%
        {}%
    \IfStrEq{\TAEEtri}{decroiss}%if descending
        {\intdessortlist{#2}}%
        {}%
    \StrSubstitute{#2}{,}{\TAEEsep}[#2]%swipping separator if necessary
}

%-----Macro for extracting
\NewDocumentCommand\ExtraireEltListeRandint{ O{,} m m o }{%
    \IfStrEq{#1}{/}%
        {%
            \setsepchar[.]{#1}%
        }%
        {%
            \setsepchar{#1}%
        }%
    \readlist*\TMPLISTRANDINT{#2}%
    \IfNoValueTF{#4}{\TMPLISTRANDINT[#3]}{\itemtomacro\TMPLISTRANDINT[#3]#4}%
```