

Number formatting

```
In[*]:= number2Printed[number_] := Module[{returnedString = "e", foo, bar, idx, oom},
    [模組]
    If[number == 1, Return["1.0e+00"], If[number == 0, Return["0.0e+00"],
    [如果] [返回] [如果] [返回]
    If[number < 1,
    [如果]
    For[idx = 1, StringLength[returnedString] == 1, idx = idx + 1,
    [For迴圈] [字串長度]
    foo = Floor[number / 10^(-idx)];
    [弱取整]
    If[foo == 0, ,
    [如果]
    bar = Round[(number - foo * 10^(-idx)) / 10^(-idx - 1)];
    [捨入]
    If[StringLength[ToString[idx]] == 1,
    [字串長度] [轉換成字串]
    returnedString = StringJoin[ToString[foo],
    [字串結合] [轉換成字串]
    ".", ToString[bar], returnedString, "-0", ToString[idx]],
    [轉換成字串] [轉換成字串]
    returnedString = StringJoin[ToString[foo], ".", ToString[bar],
    [字串結合] [轉換成字串] [轉換成字串]
    returnedString, "- ", ToString[idx]]
    [轉換成字串]
    ]
    ]
];
Return[returnedString]
[返回]
,
oom = (StringLength[ToString[DecimalForm[Floor[number] * 1.]]] - 2);
[字串長度] [轉換成字串] [十進位形式] [弱取整]
foo = Floor[number / 10^oom];
[弱取整]
bar = Round[(number - foo * 10^oom) / 10^(oom - 1)];
[捨入]
If[StringLength[ToString[oom]] == 1,
[字串長度] [轉換成字串]
returnedString = StringJoin[ToString[foo],
[字串結合] [轉換成字串]
".", ToString[bar], returnedString, "+0", ToString[oom]],
[轉換成字串] [轉換成字串]
```

```

    returnedString = StringJoin[ToString[foo], ".", ToString[bar],
                                字串結合 轉換成字串 轉換成字串
                                returnedString, "+", ToString[oom]]
                                轉換成字串
  ];
  Return[returnedString]
  返回
];
]
]
]

```

Import data

```

In[ ]:= recombinationRates = {1*^-7, 1*^-9};
populationSize = 1*^4;

In[ ]:= data = Table[0, {idx, recombinationRates}];
  表格
  Do[data[[idx]] = ToExpression[StringReplace[Import[StringJoin[NotebookDirectory[],
  Do迴圈 轉換成表示式 字串替代 導入 字串結合 筆記本目錄
    number2Printed[recombinationRates[[idx]]], "_times_lengths.txt"],
    {"[" → "{", "]" → "}", "e" → "*^"}], {idx, Length[recombinationRates]}]
    長度

In[ ]:= {ListPlot[Transpose[data[[1]]], PlotRange → {{0, 10 000}, {0, 5000}}, ImageSize → Medium],
  點集圖 轉置 繪製範圍 影像尺寸 中等
  ListPlot[Transpose[data[[1]]], PlotRange → All, ImageSize → Medium]}
  點集圖 轉置 繪製範圍 全部 影像尺寸 中等

{, }

In[ ]:= {ListPlot[Transpose[data[[2]]], PlotRange → {{0, 40 000}, {0, 5000}}, ImageSize → Medium],
  點集圖 轉置 繪製範圍 影像尺寸 中等
  ListPlot[Transpose[data[[2]]], PlotRange → All, ImageSize → Medium]}
  點集圖 轉置 繪製範圍 全部 影像尺寸 中等

{, }

```

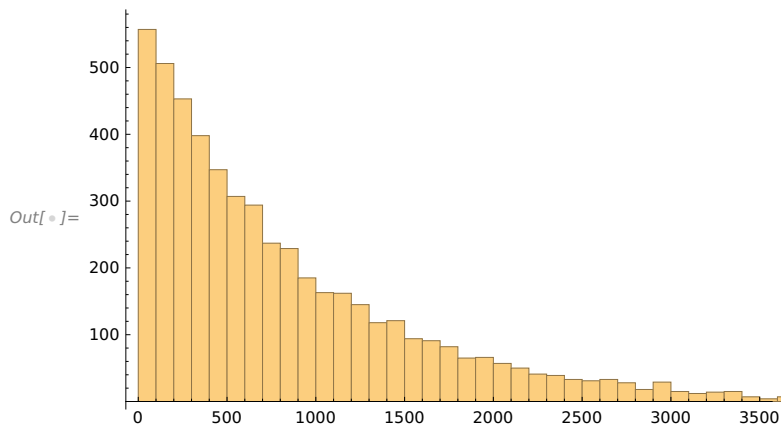
Time slice

```
In[*]:= timeSlice[list_, ini_, fin_] := Module[{slice = {}},
  Do[If[idx[[1]] < fin && idx[[1]] > ini, AppendTo[slice, idx[[2]]], ], {idx, list}];
  Return[slice]]
```

模組
... 如果
附加
返回

```
In[*]:= Histogram[timeSlice[Transpose[data[[1]]], 6000, 6500], 50]
```

長條圖
轉置



```
In[*]:= Histogram[timeSlice[Transpose[data[[1]]], 6000, 7000], 100]
```

長條圖
轉置

