`randexam`: Randomized Exam Generation and Grading

Developer Manual

Version 1.14.2

2019-02-20

1 Concepts

The exam *library* contains N_Q *questions*, each with up to N_V *variants*. Each question variant contains up to N_A *answers*, exactly one of which must be the *correct answer*. The correct answer for variant V of question Q is denoted $C(Q, V)$. The questions within the library are grouped into *zones*.

We randomly generate N_e *exams*. Each exam contains exactly one randomly chosen variant of each library question, with the question order randomly permuted within each zone. The order of the zones is not permuted. Questions are numbered globally with the library and exams, not per-zone. Question q on exam e thus corresponds to variant $V(e, q)$ of question $Q(e, q)$ in the library. The answer order for each exam question is also a random permutation of the original answer order, so $A(e, q, a)$ is the library answer corresponding to exam answer a.

We use capital-letter variables for quantities within the library and small letters for quantities within the randomly generated exams and students. Exams, questions, variants, and students are all numbered with Arabic numerals starting from 1. Answers are numbered with roman letters starting from A.

Each randomly generated exam is identified by a *key* $K(e)$ of length N_K, which is a sequence of answers $K_1, \ldots, K_{N_K}$. The keys are encoded on the Scantron forms as the answers to the final set of questions. For example, on a 96-question Scantron form, the 4-digit key "ECEA" is encoded as the answers (93) E, (94) C, (95) E, and (96) A.

During the exam each student completes a single Scantron form, resulting in N_s Scantron forms for an equal number of students. These forms are scanned and the raw data can be processed to derive the points for each student.

In summary, the size variables used are:

N_Q	number of questions (in the library and on each exam)
N_e	number of exams
N_A	maximum number of answers per question
N_V	maximum number of variants per question
N_s	number of students (equal to the number of Scantron forms)
N_g	number of ranked groups of students for statistics generation
N_D	number of answer-digits required to encode the exam number
N_K	number of answer-digits in the exam key

2 Processing pipeline

The full processing pipeline for making, using, and grading a randomized exam is shown in Figure 1. The purple output files can be inspected and hand-edited before processing runs to control the pro-

cessing pipeline.

3 File formats

All library and exam question are formatted in single TeX files. All data is stored in CSV (comma-separated values) files for easy editing by hand in a spreadsheet or text editor.

3.1 `library.tex`

The `library_template.tex` file should be copied to `library.tex` and edited to fill in the questions. Non-comment text may only go into the locations indicated in the file. This file should be a valid LaTeX file that can be typeset and edited to prepare the questions. To generate the randomized exams with the `randexam` script, the format of `library.tex` must conform to the grammar shown in Figure 2. The basic structure of `library.tex` is thus a tree, as shown in Figure 3. An outline of the LaTeX in the `library.tex` is given in Figure 4. The `randexam` script processes `library.tex` using a state machine. The state transition graph is shown in Figure 5.

3.2 `exams.tex`

This is the generated file containing all of the randomized exams. Each coverpage ends with the exam key that needs to be filled in by the student.

3.3 `specs.csv`

This file encodes the arrays:

array	dimensions	description
$K(e)$	N_e	key for exam e
$Q(e,q)$	$N_e \times N_Q$	library question for question q on exam e
$V(e,q)$	$N_e \times N_Q$	library question variant for question q on exam e
$A(e,q,a)$	$N_e \times N_Q \times N_A$	library answer corresponding to answer a for question q on exam e

The first column is the exam number, the second column is the exam key, and then triples of columns identify the library question Q, variant V, and library answer order. As an example:

```
e,K(e),"Q(e,q=1)","V(e,q=1)","A(e,q=1,:)","Q(e,q=2)","V(e,q=2)","A(e,q=2,:)"
1,ADC,2,2,DEBCA,3,2,DCEBA
2,BED,1,2,BDCAE,2,1,DECBA
```

Here the key of the first exam is $K(1) = \text{ADC}$ and the second question on this exam is variant $V(1,2) = 2$ of library question $Q(1,2) = 3$. The first answer for this question on the exam paper is answer $A(1,2,1) = \text{D}$ from the library question, while the second answer on the exam paper is answer $A(1,2,2) = \text{C}$ from the library question.

3.4 `points.csv`

The points file encodes the array:

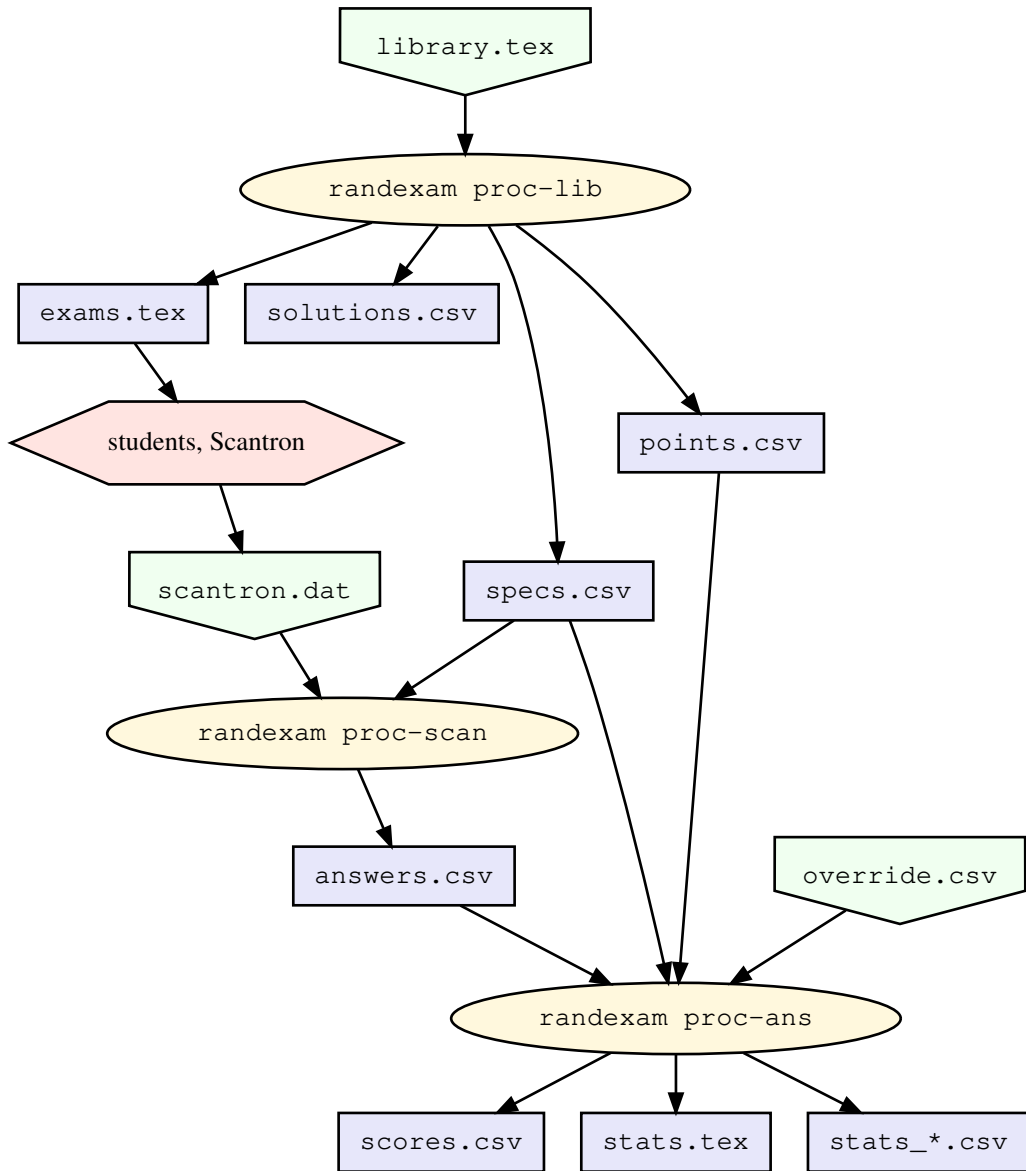| array	dimensions	description
$P(Q,V,A)$	$N_Q \times N_V \times N_A$	points awarded for giving answer A to variant V of question Q

This is stored as a four-column file with entries like:

Figure 1: The pipeline for making and grading the randomized exams. The green pentagons are input files that require human effort to generate, the purple boxes are output files, the yellow ellipses are program runs, and the pink hexagon is the students sitting the exam and the scanning of the Scantron forms.

$\langle library \rangle$ ::= $\langle preamble\ text \rangle$ '\begin{document}' $\langle body \rangle$ '\end{document}'

$\langle body \rangle$ ::= $\langle coverpage\ text \rangle$ $\langle zone\text{-}list \rangle$

$\langle zone\text{-}list \rangle$ ::= $\langle zone \rangle$ $\langle zone\text{-}list \rangle$ | $\langle zone \rangle$

$\langle zone \rangle$ ::=   '\zone' $\langle zone\ text \rangle$
         | '\zone' $\langle zone\ text \rangle$ $\langle question\text{-}list \rangle$

$\langle question\text{-}list \rangle$ ::= $\langle question \rangle$ $\langle question\text{-}list \rangle$ | $\langle question \rangle$

$\langle question \rangle$ ::=   '\question{' $\langle points \rangle$ '}' $\langle variant\text{-}list \rangle$

$\langle variant\text{-}list \rangle$ ::= $\langle variant \rangle$ $\langle variant\text{-}list \rangle$ | $\langle variant \rangle$

$\langle variant \rangle$ ::=   '\variant' $\langle variant\ text \rangle$ '\begin{answers}' $\langle answer\text{-}list \rangle$ '\end{answers}'

$\langle answer\text{-}list \rangle$ ::= $\langle answer \rangle$ $\langle answer\text{-}list \rangle$ | $\langle answer \rangle$

$\langle answer \rangle$ ::=   '\answer' $\langle answer\ text \rangle$
         | '\correctanswer' $\langle answer\ text \rangle$

Figure 2: Grammer for the `library.tex` input file expressed in partial Backus-Naur Form. The *text* entries refer to any valid LaTeX text, and *points* is a numeric value.

```
library
 ├─ preamble text
 ├─ coverpage text
 ├─ zone 1
 │   ├─ zone text
 │   ├─ question 1
 │   │   ├─ variant 1
 │   │   │   ├─ variant text
 │   │   │   └─ answers
 │   │   │       ├─ answer 1
 │   │   │       ├─ answer 2
 │   │   │       └─ ...
 │   │   ├─ variant 2
 │   │   └─ ...
 │   ├─ question 2
 │   └─ ...
 ├─ zone 2
 └─ ...
```

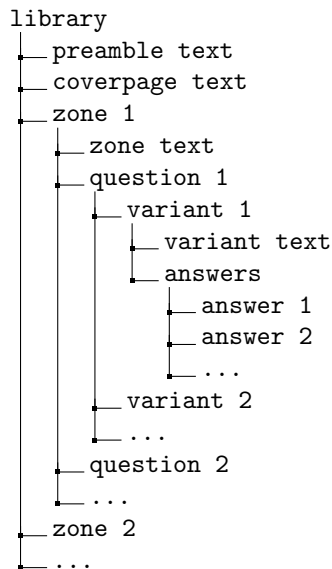Figure 3: Tree structure of the `library.tex` input file.

```
<preamble text>
\begin{document}
    <coverpage text>
    \zone
        <zone text>
        \question{<points>}
            \variant
                <variant text>
                \begin{answers}
                    \answer <answer text>
                    \correctanswer <answer text>
                    <more answers...>
                \end{answers}
            <more variants...>
        <more questions...>
    <more zones...>
\end{document}
```

Figure 4: Outline of the `library.tex` input file.

```
Q,V,A,"P(Q,V,A)"
...
2,1,A,0.0
2,1,B,0.0
2,1,C,0.0
2,1,D,1.0
2,1,E,0.0
```

Here some lines have been removed, as indicated by the ellipsis. This shows that for variant 1 of question 2, answer D is worth 1 point and all other answers are not worth any points, so $P(2, 1, 4) = 1.0$ and $P(2, 1, A) = 0$ for $A \neq 4$.

3.5 `override.csv`

The override file encodes the array:

array	dimensions	description
P_{sQ}^{over}	$N_s \times N_Q$	override points awarded to student s for library question Q (or -1 to not override)

This is stored as a CSV file with the first column being the student NetID and subsequent columns specifying questions. The column header indicates the question number. It is not necessary to include all students or all questions, and any students or questions not specified will have entries of -1 in P_{sQ}^{over}, indicating that they should not be overridden. For example:

```
NetID,8,13
RDEVILLE,0.4,7
MWEST,0,-1
```

Here two students have override scores specified for two questions. Student `RDEVILLE` will have a score of 0.4 for library question $Q = 8$ and 7 for $Q = 13$. Student `MWEST` will have a score of zero for question $Q = 8$, but for question $Q = 13$ his score will not be overridden, meaning it will be computed from the answer and points data as usual.
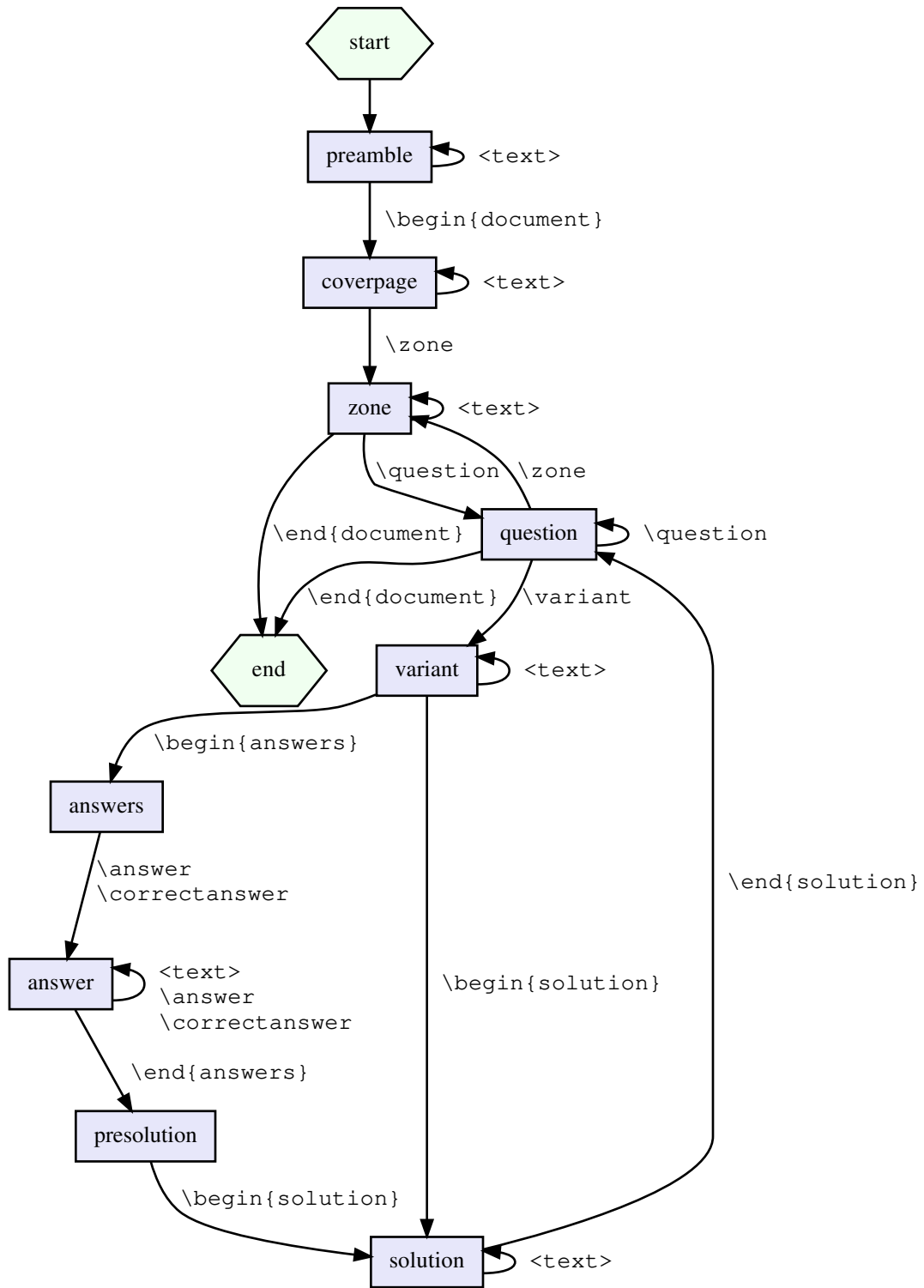
Figure 5: The state machine used by the `randexam` script to read the `library.tex` input file.

3.6 `solutions.csv`

This file gives the correct solutions for each exam in case grading by hand is necessary. This thus encodes the array:

array	dimensions	description
$c(e, q)$	$N_\mathrm{e} \times N_\mathrm{Q}$	the correct answer for question q on exam e

The first two columns give the exam number and key for reference, and then subsequent columns give the correct answers. For example:

```
e,K(e),"C(e,q=1)","C(e,q=2)","C(e,q=3)"
1,ADC,A,A,D
2,BED,D,A,A
3,CAE,A,E,D
4,DBA,A,A,D
5,ECB,B,E,A
```

Here the third exam has key $K(3) = \mathrm{CAE}$ and the correct answer for the first question on this exam is $c(3, 1) = \mathrm{A}$, while the correct answer for the second question is $c(3, 2) = \mathrm{E}$, etc.

The correct answers for each exam are derived from the correct answers in the library file, given for each library question and each variant in the array:

array	dimensions	description
$C(Q, V)$	$N_\mathrm{Q} \times N_\mathrm{V}$	the correct answer for variant V of library question Q

The derivation of $c(e, q)$ is then:

$$A(e, q, c(e, q)) = C\big(Q(e, q), V(e, q)\big). \tag{1}$$

3.7 `scantron.dat`

This is the raw data file from the Scantron machine. There are two possible formats for the Scantron data, depending on whether students are allowed to bubble in more than one answer. Selecting multiple answers is a possible approach for partial credit (where a student may get half or third credit if they select two or three answers, for example). The difference in the data file is how the answers are encoded. If only a single answer is allowed, then it contains lines like:

```
721000001001120412001Y  5380      1    N DEVILLE    R993502067   RDEVILLE 434
721000002001120412001Y  5380      1    N WEST       M795969582   MWEST    252
```

Here each answer is represented by a single digit, with $1 = \mathrm{A}$, $2 = \mathrm{B}$, etc., so student `RDEVILLE` answered D for question 1, C for question 2, and D for question 3.

If, instead, multiple answers are allowed, the question information is encoded in a two-digit decimal value that represents a little-endian "binary" encoding of the bubbled answers. That is, each answer has a numeric value ($2^0 = \mathrm{A}$, $2^1 = \mathrm{B}$, ..., $2^4 = \mathrm{E}$) and the question value is the sum of all selected answer values. The data file then contains lines like:

```
721000001001120412001Y  5380      1    N DEVILLE    R993502067   RDEVILLE 080412
721000002001120412001Y  5380      1    N WEST       M795969582   MWEST    021601
```

Here student `RDEVILLE` answered D for question 1 ($08 = 2^3 = \mathrm{D}$), C for question 2 ($04 = 2^2 = \mathrm{C}$), and both C and D for question 3 ($12 = 2^2 + 2^3 = \mathrm{C} + \mathrm{D}$).

The actual file sent from the scanning office should be renamed to `scantron.dat`. The particular parsing is controlled by the variable `MULTIPLE_ANSWERS_PER_QUESTION`; this variable only controls the parsing of `scantron.dat`.

3.8 `answers.csv`

This file is essentially just a CSV version of the data in `scantron.dat`. It encodes the arrays:

array	dimensions	description
$u(s,i)$	$N_\mathrm{s} \times 4$	student last name, first initial, student ID number, and NetID
$b(s,q,a)$	$N_\mathrm{s} \times N_\mathrm{Q} \times N_\mathrm{A}$	bubble selection of answer a given to question q by student s; 0 = empty, 1 = filled
$k(s)$	N_s	key filled in by student s

The first column in the data file is the student number, then the next four columns store the student information, the sixth column stores the exam key, and then columns 7 and up store the bubbled-in answers to the questions in exam order. As an example:

```
s,Name,Initial,Number,NetID,k(s),"b(s,q=1,:)","b(s,q=2,:)","b(s,q=3,:)"
1,DEVILLE,R,993502067,RDEVILLE,CAE,D,C,CD
2,WEST,M,795969582,MWEST,ECB,B,E,AB
```

Here we see the second student has information giving the last name $u(2,1) =$ `WEST`, the first initial $u(2,2) =$ `M`, the student number $u(2,3) = 795969582$, and the NetID $u(2,4) =$ `MWEST`. The exam key for this student is $k(2) =$ `ECB` and the student bubbled answers $b(2,1,:) = [0,1,0,0,0]$ (selecting just B) for question 1 on this exam paper, bubbled answers $b(2,2,:) = [0,0,0,0,1]$ (selecting just E) for question 2, and bubbled answers $b(2,3,:) = [1,1,0,0,0]$ (selecting both A and B) for question 3.

3.9 `scores.csv`

This file encodes the array $P_\mathrm{se}(s)$ described in Section 4. For reference, the student number and information in array $u(s,i)$ is also included in the first columns. As an example:

```
s,Name,Initial,Number,NetID,P_s(s)
1,DEVILLE,R,993502067,RDEVILLE,2.0
2,WEST,M,795969582,MWEST,2.0
```

Here we see that both students have total points $P_\mathrm{s}(1) = P_\mathrm{s}(2) = 2.0$.

3.10 `stats.tex`

This file presents various summary statistics relating to the student and question performance in the exam. It should be processed with `pdflatex` to produce the output `stats.pdf` file.

3.11 `stats_*.csv`

Multiple files are written to encode the statistics arrays described in Section 6. These are named after the variables they contain, so n_QV^a is be stored in `stats_n_a_QV.csv`, for example. The format is broadly similar to those files described in detail above, with column headings that specify the contents.

4 Grading

The map from student number s to exam number e is given by the array

array	dimensions	description
$e(s)$	N_s	exam number corresponding to student number s

This array is determined so that

$$K(e(s)) = k(s). \tag{2}$$

The points awarded to each student can be determined using the grading key for any exam, not just the particular exam that the student actually took. To make this clear, we consider the arrays:

array	dimensions	description
$P_{\text{seq}}(s,e,q)$	$N_{\text{s}} \times N_{\text{e}} \times N_{\text{Q}}$	points awarded to student s on question q scored for exam e
$P_{\text{se}}(s,e)$	$N_{\text{s}} \times N_{\text{e}}$	total points awarded to student s scored for exam e
$P_{\text{sq}}(s,q)$	$N_{\text{s}} \times N_{\text{Q}}$	points award to student s on question q using the correct exam
$P_{\text{s}}(s)$	N_{s}	total points awarded to student s scored using the correct exam
$P_{\text{sQ}}(s,Q)$	$N_{\text{s}} \times N_{\text{Q}}$	points awarded to student s for library question Q

We define an array for partial credit $\sigma(n)$ (the variable `SCORE_PER_ANSWERS` in `randexam`) which is the score value multiplier when n answers are selected (e.g., $\sigma(1) = 1$ always, but $\sigma(n) = 1/n$ can be used to award partial credit, or $\sigma(n) = 0$ for $n > 1$ will allow no partial credit). The first array in the table above is computed by:

$$P_{\text{seq}}(s,e,q) = \begin{cases} \sum_{a=1}^{N_{\text{A}}} P\Big(Q(e,q), V(e,q), A(e,q,a)\Big) b(s,q,a) \sigma\left(\sum_{a'=1}^{N_{\text{A}}} b(s,q,a') \right) & \text{if } P_{\text{sQ}}^{\text{over}}\big(s, Q(e,q)\big) < 0 \\ P_{\text{sQ}}^{\text{over}}\big(s, Q(e,q)\big) & \text{otherwise} \end{cases} \tag{3}$$

The subsequent arrays in the table above are then computed by:

$$P_{\text{se}}(s,e) = \sum_{q=1}^{N_{\text{Q}}} P_{\text{seq}}(s,e,q) \tag{4}$$

$$P_{\text{sq}}(s,q) = P_{\text{seq}}\big(s, e(s), q\big) \tag{5}$$

$$P_{\text{s}}(s) = P_{\text{se}}\big(s, e(s)\big) \tag{6}$$

$$P_{\text{sQ}}\Big(s, Q\big(e(s), q\big)\Big) = P_{\text{seq}}\big(s, e(s), q\big). \tag{7}$$

We expect that $e(s)$ maximizes $P_{\text{se}}(s,e)$ for each s, as grading with the "wrong" exam key should give random choices for each answer. Indeed, for $\hat{e} \neq e(s)$, $P_{\text{se}}(s,\hat{e})$ should be approximately $N_{\text{Q}}/N_{\text{A}}$, assuming that each question is worth one point if correct. Some significant deviations are likely, however, when the number of students N_{s} is much larger than the number of questions N_{Q}.

5 Curving

Curved score values are stored in the array:

array	dimensions	description
$P_{\text{s}}^{\text{curve}}(s)$	N_{s}	total curved points awarded to student s

The default is for no curving, in which case $P_{\text{s}}^{\text{curve}}$ is equal to P_{s}. If score curving is enabled (by the `curve_scores` configuration parameter) then $P_{\text{s}}^{\text{curve}}$ is computed using the piecewise linear function shown in Figure 6 with the following parameters:

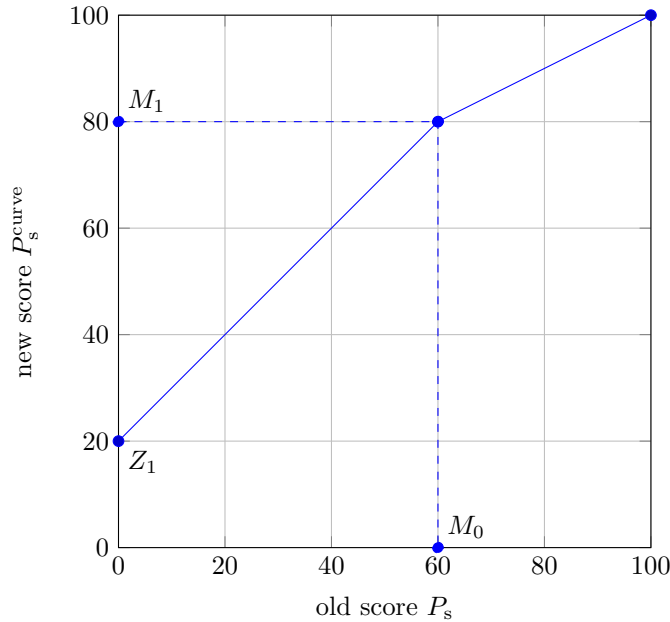| quantity	config option	description
M_0	—	median of old (uncurved) scores
M_1	`curve_new_median`	desired median of new (curved) scores
Z_1	`curve_new_zero`	desired new (curved) score for old zero score

Figure 6: Curving function that transforms old scores to new scores.

6 Statistics

As well as computing the scores for each student, we can assess how the students as a group performed on the different questions and variants. Basic statistics for this assessment are listed below.

array	dimensions	description
$n_{\mathrm{e}}^{\mathrm{s}}(e)$	N_{e}	number of students taking exam e
$n_{\mathrm{QV}}^{\mathrm{s}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	number of students given variant V of question Q
$n_{\mathrm{QVA}}^{\mathrm{a}}(Q,V,A)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}} \times N_{\mathrm{A}}$	number of students giving answer A for variant V of question Q
$n_{\mathrm{QV}}^{\mathrm{a}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	number of students giving an answer for variant V of question Q
$n_{\mathrm{QV}}^{\mathrm{na}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	number of students not giving an answer for variant V of question Q
$n_{\mathrm{Q}}^{\mathrm{a}}(Q)$	N_{Q}	number of students giving an answer for question Q
$r_{\mathrm{QVA}}^{\mathrm{a}}(Q,V,A)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}} \times N_{\mathrm{A}}$	fraction of students giving answer A for variant V of question Q
$r_{\mathrm{QV}}^{\mathrm{na}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	fraction of students not giving an answer for variant V of question Q
$P_{\mathrm{QV}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	total points awarded for variant V of question Q
$P_{\mathrm{Q}}(Q)$	N_{Q}	total points awarded for question Q
$\bar{P}_{\mathrm{QV}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	average points per student for variant V of question Q
$\bar{P}_{\mathrm{Q}}(Q)$	N_{Q}	average points per student for question Q
$\bar{P}$	1	average points per student for entire exam
$P_{\mathrm{QV}}^{\max}(QV)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	maximum points available for variant V of question Q
$P_{\mathrm{Q}}^{\max}(Q)$	N_{Q}	maximum points available for question Q
$P^{\max}$	1	maximum points available for the entire exam
$\hat{P}_{\mathrm{Q}}(Q)$	N_{Q}	normalized average points per student for question Q

array	dimensions	description
$R_{\mathrm{QV}}(Q,V)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}}$	ratio of average points for variant V to average for question Q
$n_{\mathrm{QVC}}(Q,V,C)$	$N_{\mathrm{Q}} \times N_{\mathrm{V}} \times N_{\mathrm{A}}$	number of students who bubbled in C answers for variant V of question Q
$r_{QQ}^{\mathrm{s}}(Q,\hat{Q})$	$N_{\mathrm{Q}} \times N_{\mathrm{Q}}$	correlation between student scores on questions Q and $\hat{Q}$
$r_{\mathrm{ss}}^{Q}(s,\hat{s})$	$N_{\mathrm{s}} \times N_{\mathrm{s}}$	correlation between question points for students s and $\hat{s}$
$n_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s})$	$N_{\mathrm{s}} \times N_{\mathrm{s}}$	number of mutually incorrect exam answers for students s and $\hat{s}$
$r_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s})$	$N_{\mathrm{s}} \times N_{\mathrm{s}}$	proportion of identical incorrect answers for students s and $\hat{s}$
$D_{\mathrm{Q}}(Q)$	N_{Q}	"difficulty" of question Q: one minus normalized averaged points $\hat{P}_{\mathrm{Q}}(Q)$
$r_{\mathrm{Q}}^{\mathrm{P}}(Q)$	N_{Q}	"discrimination" of question Q: correlation coefficient of $P_{\mathrm{sQ}}(s,Q)$ and $\big(P_{\mathrm{s}}(s) - P_{\mathrm{sQ}}(s,Q)\big)$
$q_{\mathrm{sQ}}(s,Q)$	$N_{\mathrm{s}} \times N_{\mathrm{Q}}$	exam question number q in which library question Q appears for student s
$V_{\mathrm{sQ}}(s,Q)$	$N_{\mathrm{s}} \times N_{\mathrm{Q}}$	variant number of library question Q given to student s
$c_{\mathrm{sQ}}(s,Q)$	$N_{\mathrm{s}} \times N_{\mathrm{Q}}$	correct answer position for library question Q given to student s

These quantities are computed by:

$$n_{\mathrm{e}}^{\mathrm{s}}(e) = \sum_{s=1}^{N_{\mathrm{s}}} \delta\big(K(e), k(s)\big) \tag{8}$$

$$n_{\mathrm{QV}}^{\mathrm{s}}(\hat{Q},\hat{V}) = \sum_{s=1}^{N_{\mathrm{s}}} \sum_{q=1}^{N_{\mathrm{Q}}} \delta\Big(\hat{Q}, Q\big(e(s),q\big)\Big)\, \delta\Big(\hat{V}, V\big(e(s),q\big)\Big) \tag{9}$$

$$n_{\mathrm{QVA}}^{\mathrm{a}}(\hat{Q},\hat{V},\hat{A}) = \sum_{s=1}^{N_{\mathrm{s}}} \sum_{q=1}^{N_{\mathrm{Q}}} \sum_{a=1}^{N_{\mathrm{A}}} \delta\Big(\hat{Q}, Q\big(e(s),q\big)\Big)\, \delta\Big(\hat{V}, V\big(e(s),q\big)\Big)\, \delta\Big(\hat{A}, A\big(e(s),q,a\big)\Big)$$
$$\cdot\, b(s,q,a)\sigma\Big(\sum_{a'} b(s,q,a')\Big) \tag{10}$$

$$n_{\mathrm{QV}}^{\mathrm{a}}(Q,V) = \sum_{A=1}^{N_{\mathrm{A}}} n_{\mathrm{QVA}}^{\mathrm{a}}(Q,V,A) \tag{11}$$

$$n_{\mathrm{QV}}^{\mathrm{na}}(Q,V) = n_{\mathrm{QV}}^{\mathrm{s}}(Q,V) - n_{\mathrm{QV}}^{\mathrm{a}}(Q,V) \tag{12}$$

$$n_{\mathrm{Q}}^{\mathrm{a}}(Q) = \sum_{V=1}^{N_{\mathrm{V}}} n_{\mathrm{QV}}^{\mathrm{a}}(Q,V) \tag{13}$$

$$r_{\mathrm{QVA}}^{\mathrm{a}}(Q,V,A) = \frac{n_{\mathrm{QVA}}^{\mathrm{a}}(Q,V,A)}{n_{\mathrm{QV}}^{\mathrm{s}}(Q,V)} \tag{14}$$

$$r_{\mathrm{QV}}^{\mathrm{na}}(Q,V) = \frac{n_{\mathrm{QV}}^{\mathrm{na}}(Q,V)}{n_{\mathrm{QV}}^{\mathrm{s}}(Q,V)} \tag{15}$$

$$P_{\mathrm{QV}}(\hat{Q},\hat{V}) = \sum_{s=1}^{N_{\mathrm{s}}} \sum_{q=1}^{N_{\mathrm{Q}}} P_{\mathrm{sq}}(s,q)\, \delta\Big(\hat{Q}, Q\big(e(s),q\big)\Big)\, \delta\Big(\hat{V}, V\big(e(s),q\big)\Big) \tag{16}$$

$$P_{\mathrm{Q}}(Q) = \sum_{V=1}^{N_{\mathrm{V}}} P_{\mathrm{QV}}(Q,V) \tag{17}$$

$$\bar{P}_{\mathrm{QV}}(Q,V) = \frac{P_{\mathrm{QV}}(Q,V)}{n_{\mathrm{QV}}^{\mathrm{s}}(Q,V)} \tag{18}$$

$$\bar{P}_{\mathrm{Q}}(Q) = \frac{P_{\mathrm{Q}}(Q)}{N_{\mathrm{s}}} \tag{19}$$

$$\bar{P} = \sum_{Q=1}^{N_{\mathrm{Q}}} \bar{P}_{\mathrm{Q}}(Q) = \frac{1}{N_{\mathrm{s}}} \sum_{s=1}^{N_{\mathrm{s}}} P_{\mathrm{s}}(s) \tag{20}$$

$$P_{\mathrm{QV}}^{\max}(Q,V) = \max_{A \in \{1,\ldots,N_{\mathrm{A}}\}} P(Q,V,A) \tag{21}$$

$$P_{\mathrm{Q}}^{\max}(Q) = \max_{V \in \{1,\ldots,N_{\mathrm{V}}\}} P_{\mathrm{QV}}^{\max}(Q,V) \tag{22}$$

$$P^{\max} = \sum_{Q=1}^{N_{\mathrm{Q}}} P_{\mathrm{Q}}^{\max}(Q) \tag{23}$$

$$\hat{P}_{\mathrm{Q}}(Q) = \frac{\bar{P}_{\mathrm{Q}}(Q)}{P_{Q}^{\max}(Q)} \tag{24}$$

$$R_{\mathrm{QV}}(Q,V) = \frac{\bar{P}_{\mathrm{QV}}(Q,V)}{\bar{P}_{\mathrm{Q}}(Q)} \tag{25}$$

$$n_{\mathrm{QVC}}(Q,V,C) = \sum_{s=1}^{N_{\mathrm{s}}} \delta\left(C - \sum_{a=1}^{N_{\mathrm{A}}} b\big(s, q_{\mathrm{sQ}}(s,Q), a\big)\right) \delta\big(V - V_{\mathrm{sQ}}(s,Q)\big) \tag{26}$$

$$r_{\mathrm{QQ}}^{\mathrm{s}}(Q,\hat{Q}) = \frac{\sum_{s=1}^{N_{\mathrm{s}}} (P_{sQ}(s,Q) - \bar{P}_Q(Q))(P_{sQ}(s,\hat{Q}) - \bar{P}_Q(\hat{Q}))}{\sqrt{\sum_{s=1}^{N_{\mathrm{s}}} (P_{sQ}(s,Q) - \bar{P}_Q(Q))^2} \sqrt{\sum_{s=1}^{N_{\mathrm{s}}} (P_{sQ}(s,\hat{Q}) - \bar{P}_Q(\hat{Q}))^2}} \tag{27}$$

$$r_{\mathrm{s}}^{\mathrm{Q}}(s,\hat{s}) = \frac{\sum_{Q=1}^{N_{\mathrm{Q}}} (P_{sQ}(s,Q) - \bar{P}_Q(Q))(P_{sQ}(s,\hat{Q}) - \bar{P}_Q(\hat{Q}))}{\sqrt{\sum_{Q=1}^{N_{\mathrm{Q}}} (P_{sQ}(s,Q) - \bar{P}_Q(Q))^2} \sqrt{\sum_{Q=1}^{N_{\mathrm{Q}}} (P_{sQ}(s,\hat{Q}) - \bar{P}_Q(\hat{Q}))^2}} \tag{28}$$

$$n_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s}) = \sum_{q=1}^{N_{\mathrm{Q}}} \delta\big(0, P_{\mathrm{sq}}(s,q)\big)\, \delta\big(0, P_{\mathrm{sq}}(\hat{s},q)\big) \tag{29}$$

$$r_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s}) = \frac{1}{n_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s})} \sum_{q=1}^{N_{\mathrm{Q}}} \delta\big(a(s,q), a(\hat{s},q)\big)\, \delta\big(0, P_{\mathrm{sq}}(s,q)\big)\, \delta\big(0, P_{\mathrm{sq}}(\hat{s},q)\big) \tag{30}$$

$$D_{\mathrm{Q}}(Q) = 1 - \hat{P}_{\mathrm{Q}}(Q) \tag{31}$$

$$r_{\mathrm{Q}}^{\mathrm{P}}(Q) = \frac{\sum_{s=1}^{N_{\mathrm{s}}} \Big(P_{sQ}(s,Q) - \bar{P}_Q(Q)\Big)\Big(\big(P_s(s) - P_{sQ}(s,Q)\big) - \big(\bar{P} - \bar{P}_Q(Q)\big)\Big)}{\sqrt{\sum_{s=1}^{N_{\mathrm{s}}} \Big(P_{sQ}(s,Q) - \bar{P}_Q(Q)\Big)^2} \sqrt{\sum_{s=1}^{N_{\mathrm{s}}} \Big(\big(P_s(s) - P_{sQ}(s,Q)\big) - \big(\bar{P} - \bar{P}_Q(Q)\big)\Big)^2}} \tag{32}$$

$$q_{\mathrm{sQ}}(s,\hat{Q}) = \sum_{q=1}^{N_{\mathrm{Q}}} \delta\Big(\hat{Q}, Q\big(e(s),q\big)\Big)q \qquad \Longrightarrow \qquad Q\big(e(s), q_{\mathrm{sQ}}(s,\hat{Q})\big) = \hat{Q} \tag{33}$$

$$V_{\mathrm{sQ}}(s,Q) = V\big(e(s), q_{\mathrm{sQ}}(s,Q)\big) \tag{34}$$

$$c_{\mathrm{sQ}}(s,Q) = c\big(e(s), q_{\mathrm{sQ}}(s,Q)\big) \tag{35}$$

where $\delta(x,y)$ is the Kronecker delta function, equal to one with equal arguments and zero otherwise.

We expect that $R_{\mathrm{QV}}(Q,V) \approx 1$, indicating that all variants were equally difficult for question Q. We also expect that $r_{\mathrm{ss}}^{\mathrm{ai}}(s,\hat{s}) \approx 1/N_{\mathrm{A}}$ for $e(s) = e(\hat{s})$, indicating only random association between

incorrect answers given by students with identical exams, although this may be legitimately higher due to correlations between student mistakes.

To understand how students of different ability levels perform it is helpful to group students by their exam scores. We consider N_{g} almost-equal-sized *groups* of students, where all students in group g perform worse than all students in group $g+1$, etc. We can then compute the following per-group quantities:

array	dimensions	description
$s(r)$	N_{s}	student number at rank r
$g(s)$	N_{g}	group number of student s
$n_{\mathrm{g}}^{\mathrm{s}}(g)$	N_{g}	number of students in group g
$P_{\mathrm{gQV}}(g,Q,V)$	$N_{\mathrm{g}} \times N_{\mathrm{Q}} \times N_{\mathrm{V}}$	total points awarded for variant V of question Q to students in group g
$P_{\mathrm{gQ}}(g,Q)$	$N_{\mathrm{g}} \times N_{\mathrm{Q}}$	total points awarded for question Q to students in group g
$\bar{P}_{\mathrm{gQ}}(g,Q)$	$N_{\mathrm{g}} \times N_{\mathrm{Q}}$	average points awarded for question Q to students in group g
$\hat{P}_{\mathrm{gQ}}(g,Q)$	$N_{\mathrm{g}} \times N_{\mathrm{Q}}$	normalized average points awarded for question Q to students in group g

The definitions for the above quantities are:

$$P_{\mathrm{s}}\big(s(r)\big) \le P_{\mathrm{s}}\big(s(r+1)\big) \tag{36}$$

$$g\big(s(r)\big) = \left\lfloor \frac{(r-1)\,N_{\mathrm{g}}}{N_{\mathrm{s}}} \right\rfloor + 1 \tag{37}$$

$$n_{\mathrm{g}}^{\mathrm{s}}(\hat{g}) = \sum_{s=1}^{N_{\mathrm{s}}} \delta\big(\hat{g}, g(s)\big) \tag{38}$$

$$P_{\mathrm{gQV}}(\hat{g},\hat{Q},\hat{V}) = \sum_{s=1}^{N_{\mathrm{s}}} \sum_{q=1}^{N_{\mathrm{Q}}} P_{\mathrm{sq}}(s,q)\; \delta\big(\hat{g}, g(s)\big)\; \delta\big(\hat{Q}, Q(e(s),q)\big)\; \delta\big(\hat{V}, V(e(s),q)\big) \tag{39}$$

$$P_{\mathrm{gQ}}(g,Q) = \sum_{V=1}^{N_{\mathrm{V}}} P_{\mathrm{gQV}}(g,Q,V) \tag{40}$$

$$\bar{P}_{\mathrm{gQ}}(g,Q) = \frac{P_{\mathrm{gQ}}(g,Q)}{n_{\mathrm{g}}^{\mathrm{s}}(g)} \tag{41}$$

$$\hat{P}_{\mathrm{gQ}}(g,Q) = \frac{\bar{P}_{\mathrm{gQ}}(g,Q)}{P_{\mathrm{Q}}^{\mathrm{max}}(Q)} \tag{42}$$

7 Implementation notes

The randomized exam procedure described in this document is implemented in the `randexam` Python script. The configuration section at the top of the file should be edited as appropriate.

The internal variables in the script follow the notation in this document and use `numpy` for array handling. The only significant departure is that all variables are numbered from zero rather than from one, as Python uses zero-indexed arrays. The conversion between one-based and zero-based quantities occurs during reading and writing. All internal variables and computations are zero-based.

In this document overloaded notation is used for index variables and arrays, so that we write both $Q(e,q)$ and $C(Q,V)$, where Q is understood to be a dummy variable in the latter expression. In the code, plain variable names like `Q` refer to the array and names like `Qi` and `Qj` are used for dummy index variables.

All variables that represent answers are string-valued and store values A, B, etc, rather than an integer. These correspond to indexes A $\leftrightarrow$ 0, B $\leftrightarrow$ 1, etc. To convert between zero-based integer

indexes and string representations, the functions `chr2ind()` and `ind2chr()` can be used. For example, to access a value in the $A(e, q, a)$ array we can write:

```
e = 7
q = 0
a = "D"
print(A(e, q, chr2ind(a)))
```

The bubbled-in answers of students are stored in a "bubble" array b, which uses 0 for unselected answers and 1 for selections. There are a three conversion routines that deal with the bubble list format: `binary2bubble()`, `string2bubble()`, and `bubble2string()`, which do exactly what they say. The conversions to and from strings make use of the `ch2ind()` and `ind2chr()` routines. The controlling of the parsing is through the `MULTIPLE_ANSWERS_PER_QUESTION` variable, and the partial credit scoring (the $\sigma(n)$ array) is stored as `SCORE_PER_ANSWERS`.

8 Key generation

The exam keys are generated by encoding the exam number e in base-N_A using N_D digits and then appending two or three checksum digits. The exam-number digits are in little-endian order, with least significant digit first, so for zero-based digits K_i and zero-based indexes i they are

$$K_i = \left\lfloor \frac{e}{(N_A)^i} \right\rfloor \mod N_A, \text{ for } i = 0, \ldots, (N_D - 1). \tag{43}$$

The N_C checksum digits are the *parity digit* K_P, a *Fletcher-type* checksum K_F, and possibly a modified Fletcher-type checksum K_M, defined for zero-based digits K_i and zero-based indexes i by

$$K_P = \sum_{i=0}^{N_D-1} K_i \mod N_A \tag{44}$$

$$K_F = \sum_{i=0}^{N_D-1} w_{F,i} K_i \mod N_A \qquad w_{F,i} = \left(i \mod (N_A - 1) \right) + 1 \tag{45}$$

$$K_M = \sum_{i=0}^{N_D-1} w_{M,i} K_i \mod N_A \qquad w_{M,i} = \left(\left\lfloor \frac{i}{N_A - 1} \right\rfloor \mod (N_A - 1) \right) + 1. \tag{46}$$

Using the first two checksum digits ensures a minimum Hamming distance of 3 for at most $(N_A - 1)$ non-checksum digits (up to $N_A^{(N_A-1)}$ different random exams). If more digits are required to encode the exam numbers then the third checksum digit is appended to the key, ensuring a Hamming distance of at least 3 for up to $(N_A - 1)^2$ non-checksum digits, allowing up to $N_A^{(N_A-1)^2}$ different random exams. In the case of $N_A = 5$ the key lengths are:

number of exams	exam-number digits N_D	checksum digits	key length $N_K = N_D + N_C$
5	1	2	3
25	2	2	4
125	3	2	5
625	4	2	6
3125	5	3	8
15625	6	3	9

9 Contributors

The following people have contributed to the development of `randexam`:

Matt West: Primary developer.

Lee DeVille: Concept development, testing, individualized student feedback code.

Rick Laugesen: Documentation fixes.

Dallas Trinkle: Support for multiple answers, partial credit, and multicolumn answers.

Craig Zilles: Bug fixes.