

CodeRunner

Correcció automàtica d'exercicis de programació a Moodle



Universitat
de les Illes Balears

Antoni Oliver
antoni.oliver@uib.cat

Introducció

- ◆ Assignatures on la programació és part central.
- ◆ Existeix un llistat d'exercicis per fer.
- ◆ Els alumnes han de fer els exercicis.
 - ◆ Però molts no els fan.
 - ◆ No adquireixen les competències.
 - ◆ No segueixen l'assignatura.
 - ◆ Suspenen.
- ◆ Per tant, hem de fer-los fer els exercicis.
 - ◆ I si no, si podem saber que els que els fan, aproven, ens llevarem un pes de sobre.

Eines per fer exercicis autocorregits de programació

- ◆ Eines externes:
 - ◆ **CodingBat**: massa senzill i molt poc flexible.
 - ◆ **DOMJudge**: molt complet, s'ha d'instal·lar en un servidor.
 - ◆ **Codewars**: molt complet, però és més una xarxa social.
 - ◆ **Qualified**: versió privada de *Codewars*, amb suport per a educació. De pagament.
 - ◆ **replit classroom**: molt complet, molt flexible, gratuït. Ja no funciona.
 - ◆ **replit Teams for Education**: molt complet, multiplayer. \$1000/any.

replit classroom

The screenshot displays the Replit Classroom interface. At the top, the Replit logo is on the left, and the title "T2E03 - Llegeix enter, és positiu?" is in the center. To the right of the title are links for "my repls", "teacher", "student", "notifications" (with a red badge showing "999"), "talk", "learn/teach", and a user profile "aoliver (0)". Below the title bar, there are three buttons: "share your program", "enhance your work after submitting", and "export to repl".

The main workspace is divided into two panes. The left pane shows a code editor with a file explorer on the left containing "Main.java" and "LT.java". The code in "Main.java" is as follows:

```
1 class Main {
2   public static void main(String[] args) {
3     int num;
4     num = LT.readInt();
5     System.out.println("5 és un nombre positiu");
6   }
7 }
```

The right pane contains the exercise instructions. It starts with "Due: --" and a green "submit" button. Below this, it says "Instructions from your teacher:" followed by the title "Tema 2 - exercici 3". The instructions read: "Escriu un programa que llegeix un nombre enter i mostra si el seu valor és positiu. Consideram que el nombre 0 és positiu." Below the instructions, it says "Per exemple:" and shows two input fields with the text "5 és un nombre positiu" and "-1 és un nombre negatiu".

At the bottom of the left pane is a terminal window showing the output of the program:

```
java version "1.8.0_31"
Java(TM) SE Runtime Environment (build 1.8.0_31-b13)
Java HotSpot(TM) 64-Bit Server VM (build 25.31-b07, mixed mode)
>
```

In the bottom right corner of the interface, there is a small icon for "Privacy - Cookies".

replit classroom

-1 - details

Failure reason: Output mismatch

Teacher's input:

```
-1
```

Teacher expected to see:

```
-1 és un nombre  
negatiu
```

Your output:

```
-1 és un nombre  
positiu
```

Difference:

```
-1 és un nombre negatiupositiu
```

Un petit estudi

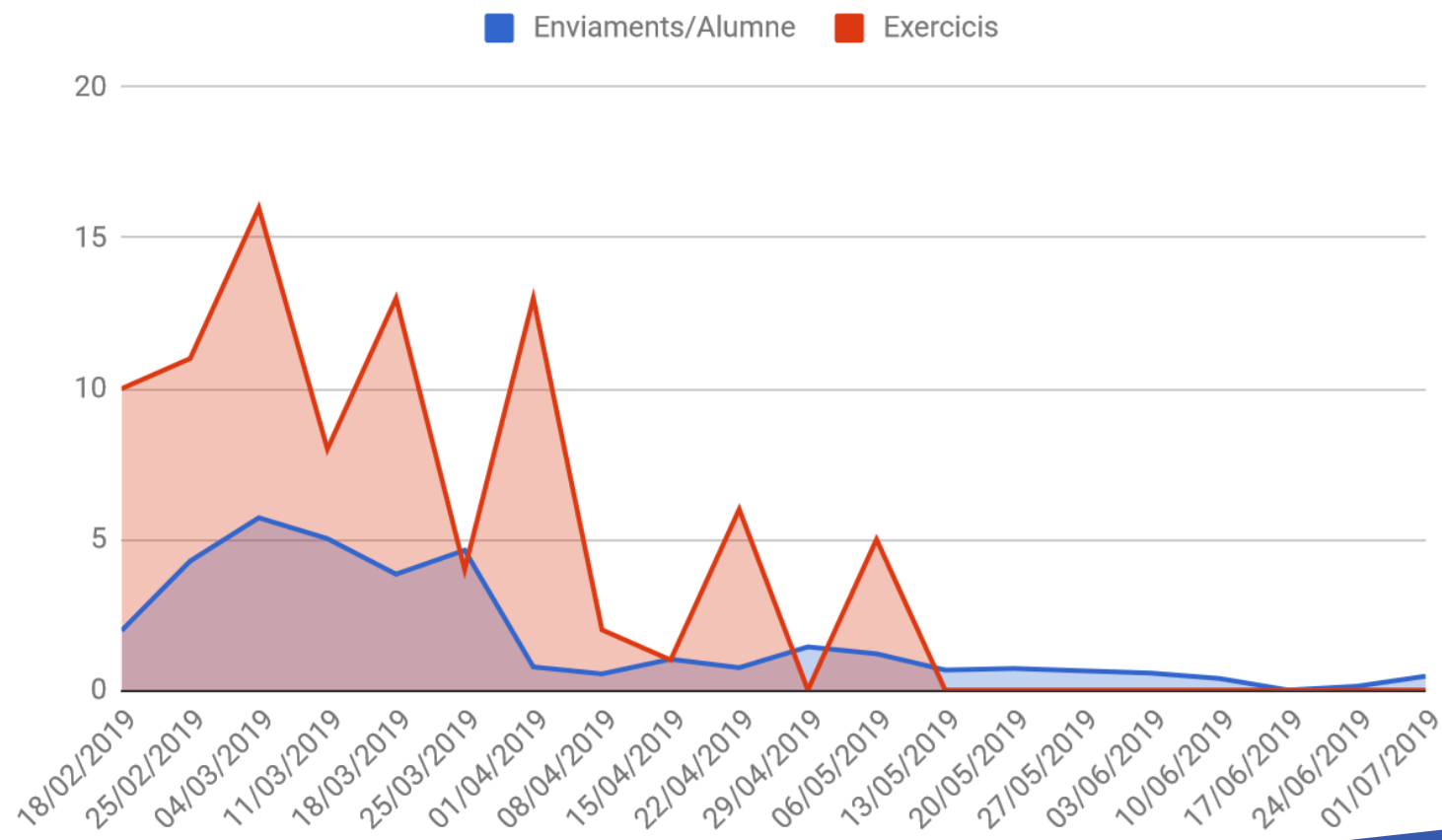
- ◆ Un grup de Programació I a Telemàtica.
- ◆ Exercicis per fer a una pàgina web externa (*replit*)
 - ◆ 2018-19 (55 alumnes, 90 exercicis, 1356 enviaments)
 - ◆ 2019-20 (57 alumnes, 104 exercicis, 2532 enviaments)
 - ◆ 2020-21 (77 alumnes, 102 exercicis, 3488 enviaments)
- ◆ Classificació d'exercicis fets
 - ◆ Feta de manera un poc pirata amb un Google Sheets.
- ◆ Gamificació (2019-20 i 2020-21)
 - ◆ Fer 6 d'aquests exercicis donava recompenses.

Classificació

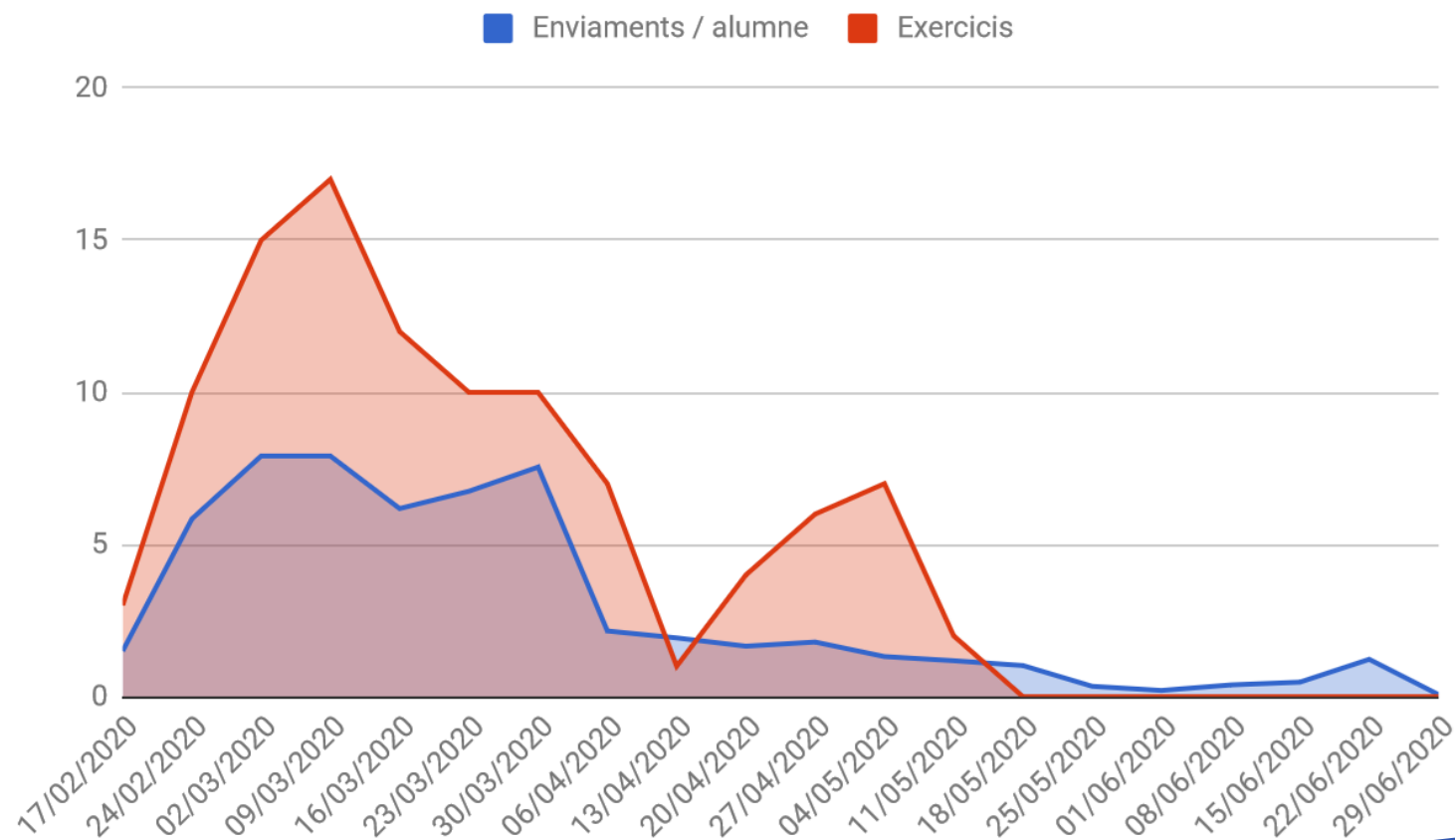
Classificació

| # | Alumne/a | P.G. | Ex. | T1 | |
|----|----------|------|-----|----|---|
| 1 | C | 6,5 | 99 | 1 | ^ |
| 2 | R | 19,5 | 99 | 1 | |
| 3 | M | 4,25 | 99 | 1 | |
| 4 | L | 3,25 | 99 | 1 | |
| 4 | P | 3,25 | 99 | 1 | |
| 6 | A | 0,75 | 97 | 1 | |
| 7 | B | 15,7 | 93 | 1 | |
| 8 | G | 10,7 | 96 | 1 | |
| 9 | C | 11 | 95 | 1 | |
| 10 | P | 31,5 | 88 | 1 | |
| 11 | B | 13 | 91 | 1 | v |
| | | | | | < > |
| | | | | | Resum Students Gamification log > |

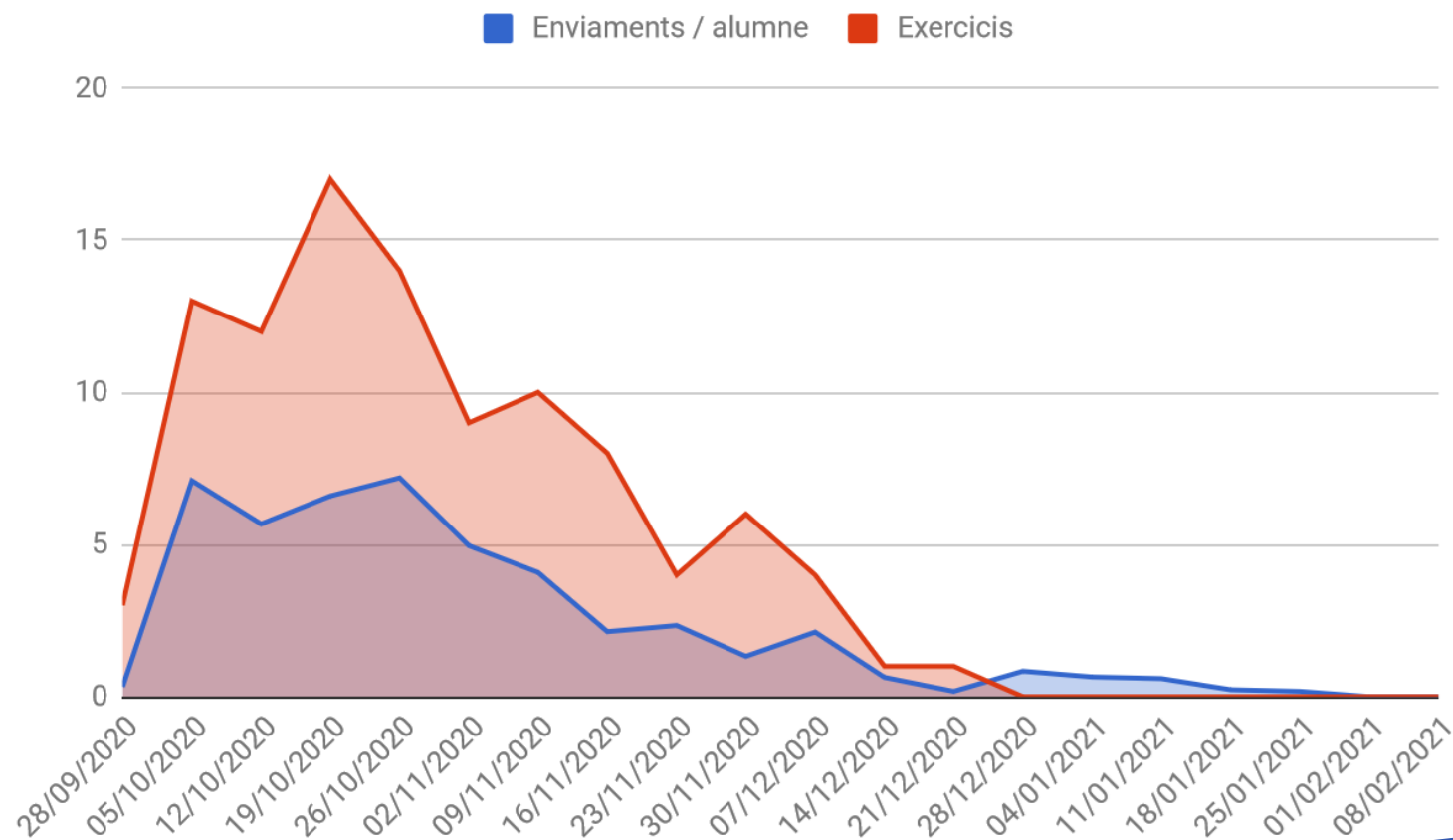
Resultats – timeline – 2018-19



Resultats – timeline – 2019-20

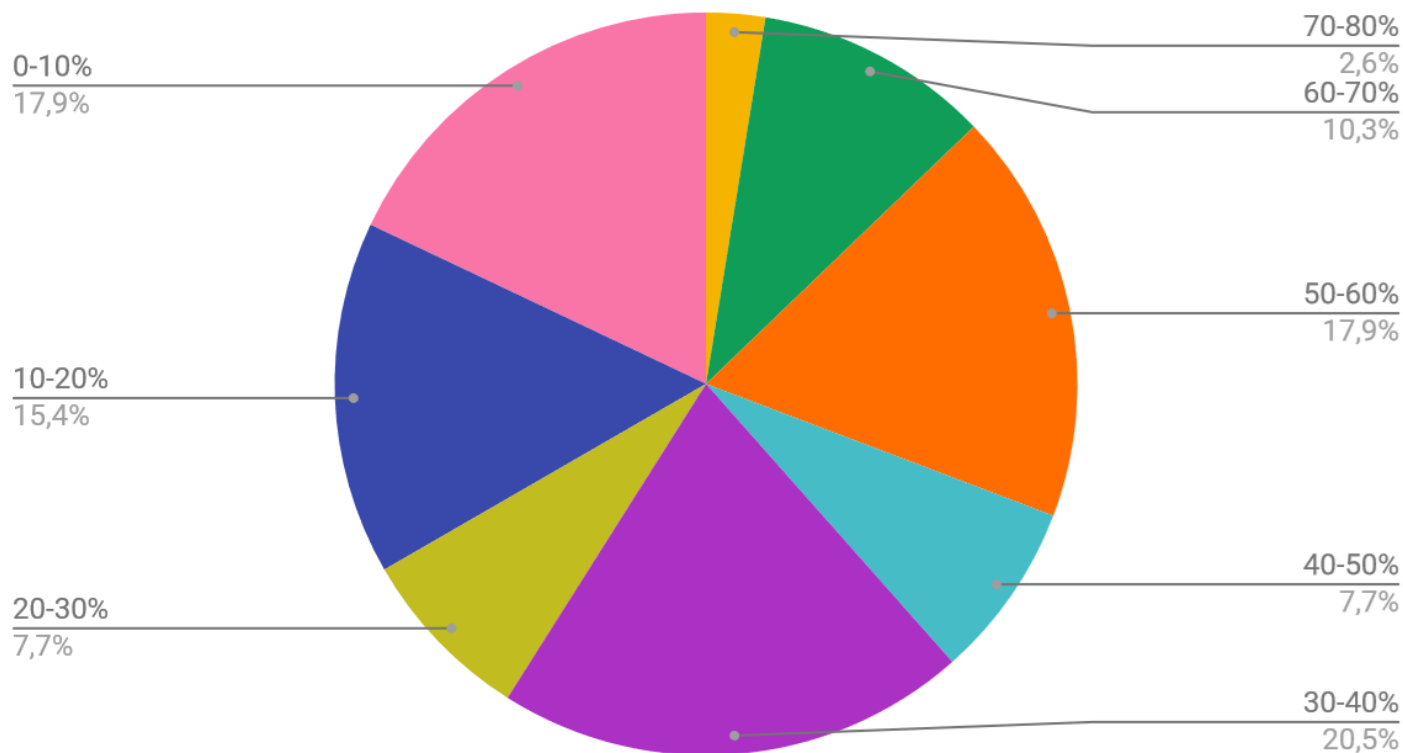


Resultats – timeline – 2020-21



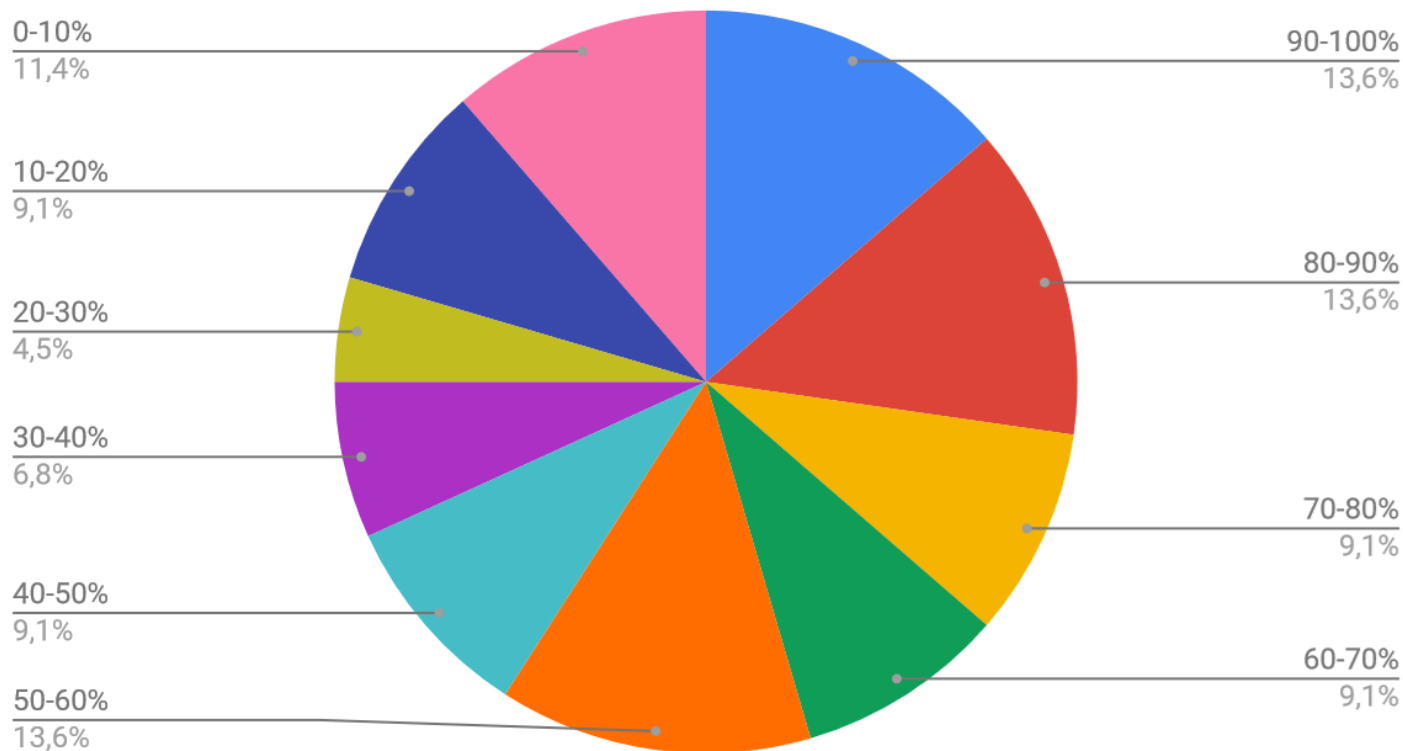
Resultats – feina dels alumnes – 2018-19

Alumnes que han fet el % dels exercicis



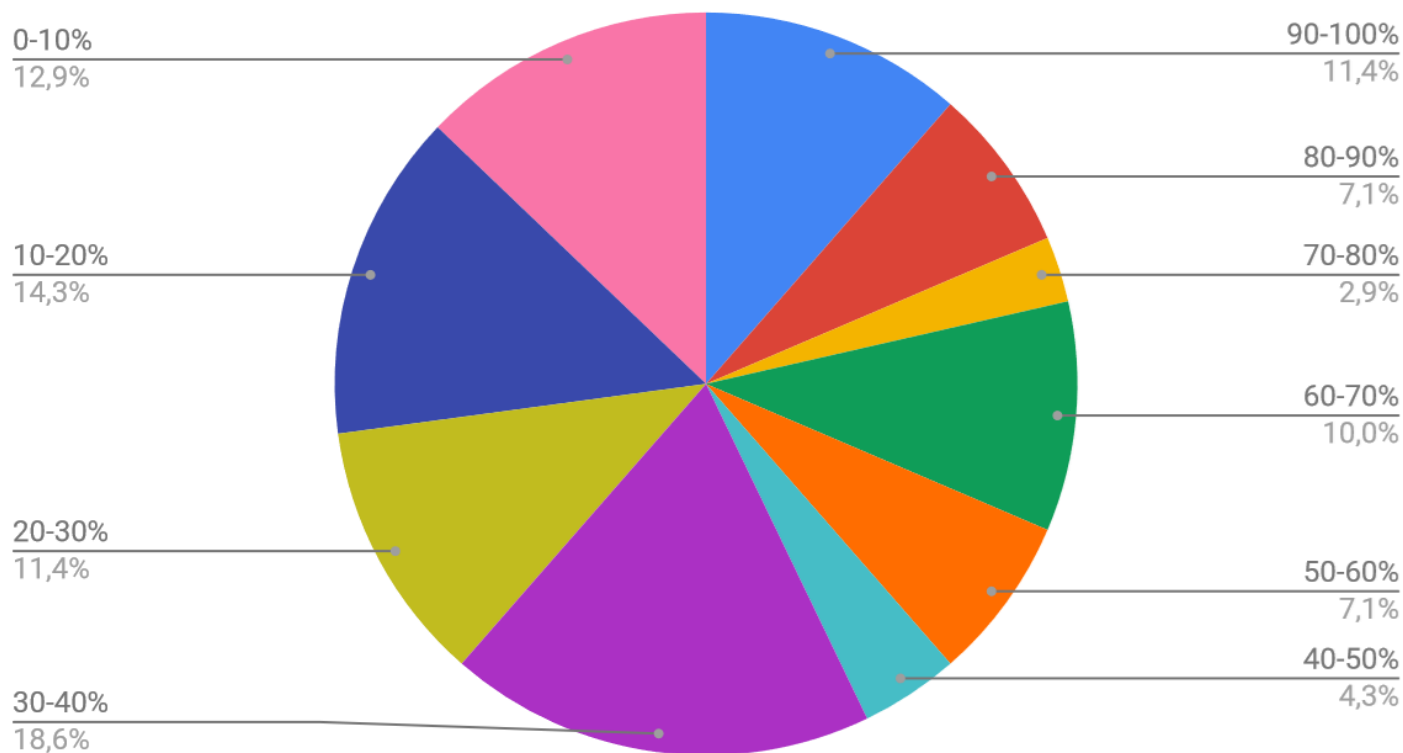
Resultats – feina dels alumnes – 2019-20

Alumnes que han fet el % dels exercicis



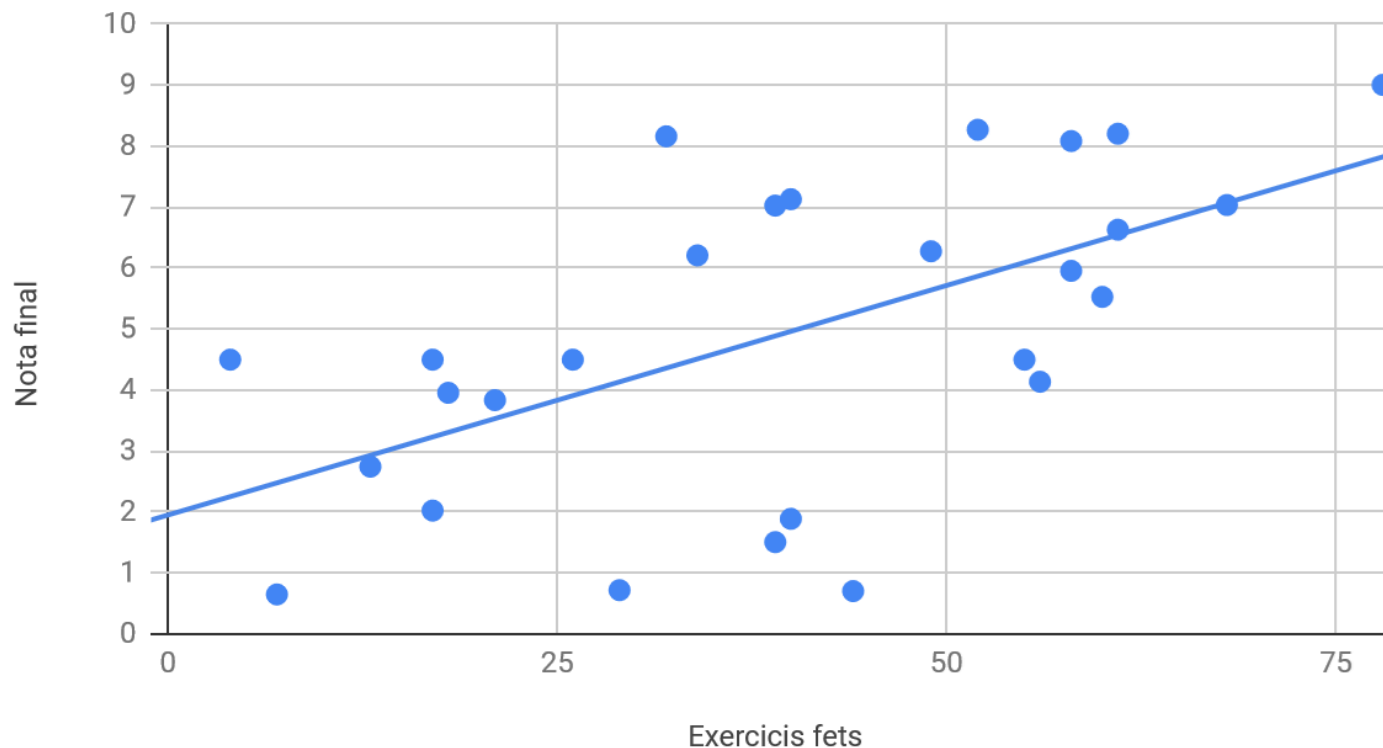
Resultats – feina dels alumnes – 2020-21

Alumnes que han fet el % dels exercicis



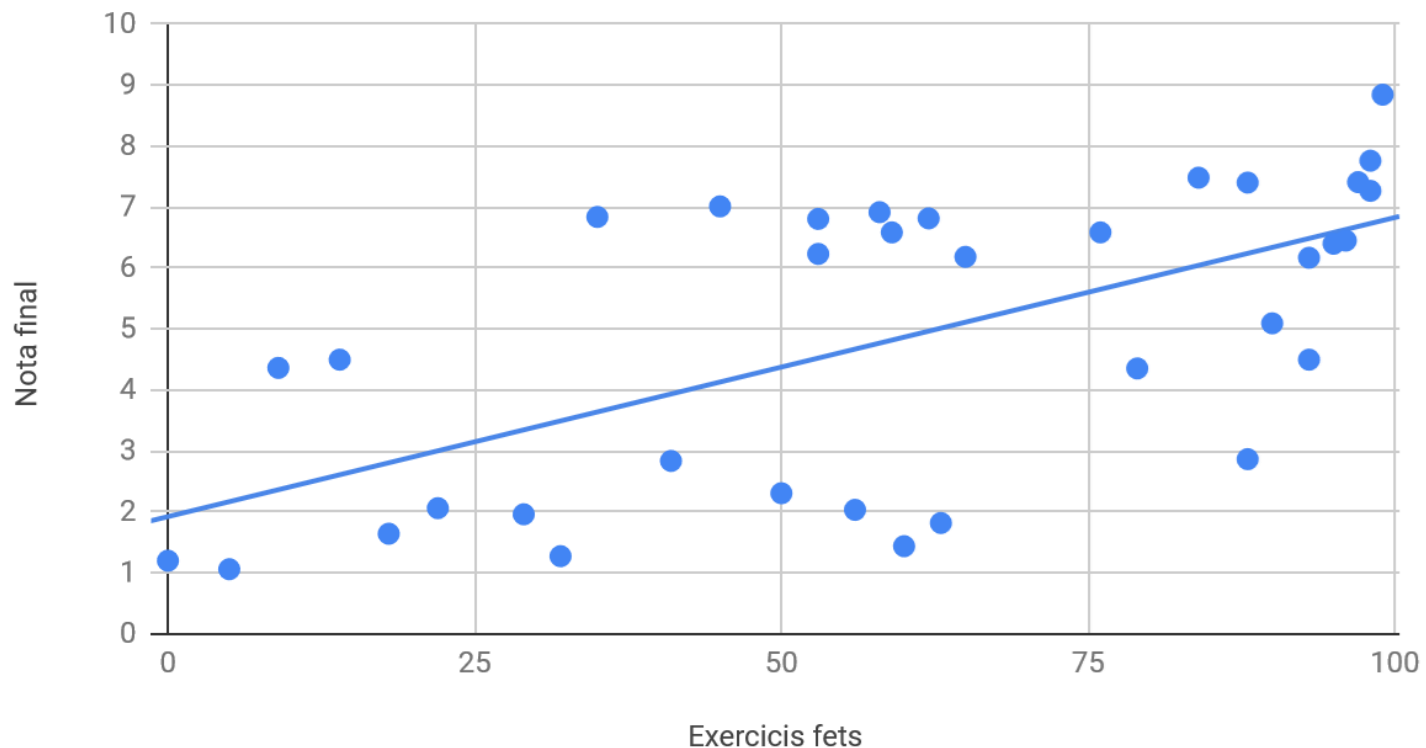
Resultats – exercicis fets i nota final – 2018-19

Nota final en funció dels exercicis fets 2018-19



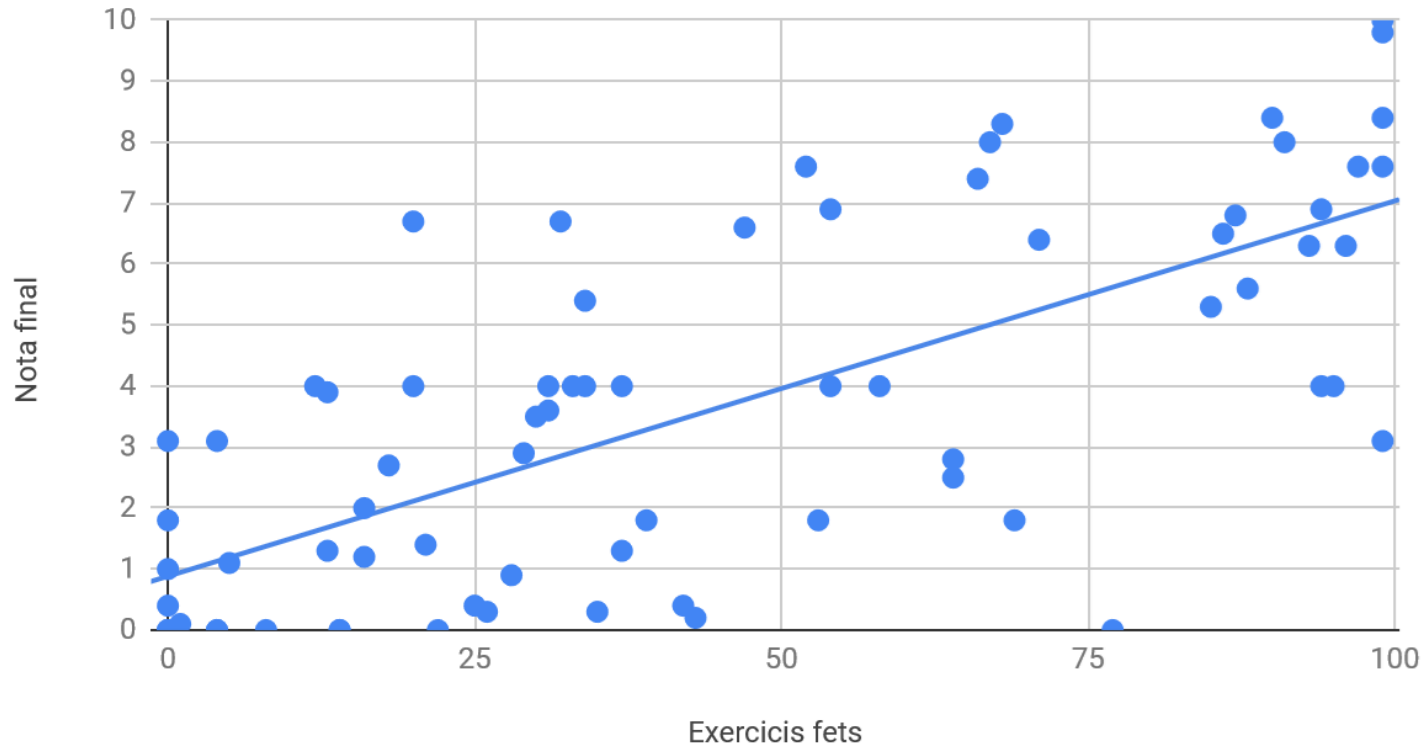
Resultats – exercicis fets i nota final – 2019-20

Nota final en funció dels exercicis fets 2019-20



Resultats – exercicis fets i nota final – 2020-21

Nota final en funció dels exercicis fets 2020-21



Conclusió

- ◆ Hi ha alumnes que volen fer exercicis.
- ◆ Agraeixen poder saber si el que fan és correcte.
 - ◆ No és només que funcioni correctament, però!
- ◆ Si els engrescam un poc, fan més feina.
- ◆ Els que fan més feina, treuen millor nota.
- ◆ Per tant, crec que és una cosa positiva i que tots podríem fer.

Però...

Però






























- ◆ *repl.it classroom* ja no es pot fer servir.
- ◆ *replit Teams for Education* no és gratuït.
- ◆ No es pot integrar *replit* amb *Moodle* (Aula Digital).
- ◆ Tot i això, *replit* (el producte estàndard) és una eina bastant útil per compartir codi tipus *Google Docs*.
 - ◆ Alguns alumnes l'han fet servir per fer la pràctica.

Solució pel curs 2021-22

- ◆ **CodeRunner**: extensió de Moodle.
 - ◆ Nou tipus de pregunta als qüestionaris.
 - ◆ Poden escriure el codi al navegador o penjar adjunts.
 - ◆ Suport integrat per a C, C++, Java, Python i php.
 - ◆ Molt flexible: si se n'instal·la el compilador, qualsevol.
 - ◆ Proves segons tres tipus de pregunta:
 - ◆ **Mètode**: paràmetres -> resultat.
 - ◆ **Classe**: programa -> resultat.
 - ◆ **Programa**: entrada -> sortida.
 - ◆ Tipus de pregunta personalitzats:
 - ◆ Molt flexible.
 - ◆ La resposta s'envia a un servidor de l'LTIM.
 - ◆ Si no funciona, digueu-me coses!

Exemple CodeRunner – Enunciat

▼ Paràmetres generals

| | |
|---------------------|--|
| Categoria actual | Exercicis (104) <input checked="" type="checkbox"/> Utilitza aquesta categoria |
| Desa a la categoria | Exercicis (104) ▾ |
| Nom de la pregunta | ! T2E04 - Llegeix real, és entre -1 i +1? |
| Text de la pregunta | ! <div><div><div>A ▾ B I       </div><div>               </div><div>     </div></div></div> <p>Escriu un programa que llegeix un nombre real i mostra si el seu valor es troba entre -1.0 i 1.0 (no inclosos).</p> <p>Feu servir la classe LT per obtenir valors del teclat. Per exemple, aquest tros de codi demana i llegeix un nombre del teclat:</p> <pre>System.out.print("Introdueix un nombre real: "); nombre = LT.readDouble();</pre> <p>Exemple d'execució 1:</p> <pre>Introdueix un nombre real: 0 0.0 és entre -1.0 i 1.0.</pre> <p>Exemple d'execució 2:</p> <pre>Introdueix un nombre real: -0.5</pre> |

Enunciat que veurà l'alumne

Exemple CodeRunner – Solució

▼ Answer

Answer

Sample answer attachments

```
1- /*
2 Aquest programa llegeix un nombre del teclat i diu si és positiu o negatiu.
3 1. Demanam i llegim el nombre real.
4 2. Si és entre -1.0 i 1.0 (no inclosos), ho deim; si no, deim que no ho és.
5 */
6 public class Main {
7     public static void main(String[] args) {
8
9         // Declaracions
10        double nombre;
11    }
```

Mida màxima per als fitxers nous: 100MB

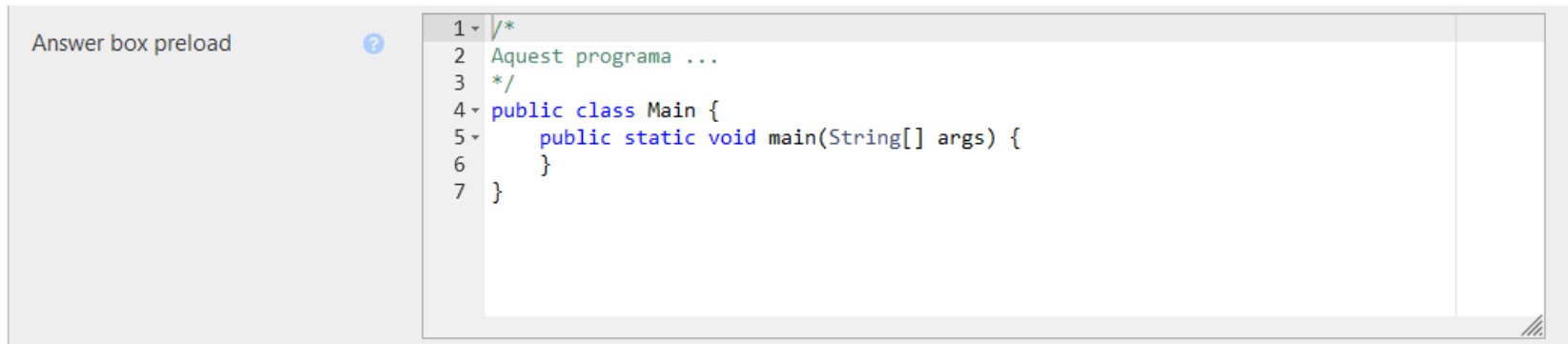
Solució que l'alumne pot veure si es marca "resposta correcta" a opcions de revisió.

Opcions de revisió

| Mentre es fa l'intent | Immediatament després de cada intent |
|---|---|
| <input checked="" type="checkbox"/> L'intent | <input checked="" type="checkbox"/> L'intent |
| <input checked="" type="checkbox"/> Si és correcta | <input checked="" type="checkbox"/> Si és correcta |
| <input checked="" type="checkbox"/> Punts | <input checked="" type="checkbox"/> Punts |
| <input checked="" type="checkbox"/> Retroacció específica | <input checked="" type="checkbox"/> Retroacció específica |
| <input checked="" type="checkbox"/> Retroacció general | <input checked="" type="checkbox"/> Retroacció general |
| <input type="checkbox"/> Resposta correcta | <input type="checkbox"/> Resposta correcta |
| <input type="checkbox"/> Retroacció global | <input checked="" type="checkbox"/> Retroacció global |

Exemple CodeRunner – Resposta inicial

Answer box preload



The screenshot shows a CodeRunner interface. On the left, there is a grey sidebar with the text "Answer box preload" and a question mark icon. The main area is a code editor with a light grey background. It contains the following Java code:

```
1  /*
2  Aquest programa ...
3  */
4  public class Main {
5      public static void main(String[] args) {
6      }
7  }
```

Podem començar la resposta de l'alumne perquè així sàpiga per on començar o que només hagi d'omplir els buits.

Exemple CodeRunner – Proves

Proves que s'executaran. Aquesta n'és una.

Test cases

The screenshot shows the 'Test cases' interface in CodeRunner. It contains several sections with annotations:

- Test case 1**: An annotation box contains the text 'Codi, si s'admet (l'alumne pot veure'l)'.
- Standard Input**: An annotation box contains the text 'Entrada estàndard'. The input field below it contains the value '0'.
- Expected output**: An annotation box contains the text 'Sortida estàndard'. The output field below it contains the text 'Introdueix un nombre real: 0.0 és entre -1.0 i 1.0.'. An orange arrow points from a text box to this output.
- Extra template data**: An annotation box contains the text 'Depèn del tipus de pregunta'.
- Test properties:** At the bottom, there are checkboxes for 'Use as example' and 'Hide rest if fail', a 'Display' button, a 'Show' button with a dropdown arrow, a 'Mark' field with the value '1.000', and an 'Ordering' field with the value '10'.

An orange text box with an arrow pointing to the 'Expected output' field contains the text: 'Si s'executés interactivament, aquí hi aniria el zero i el salt de línia que introduiria l'usuari.'

Exemple CodeRunner – Resposta de l'alumne

Answer:

Reset answer

```
1  /*
2  Aquest programa ...
3  */
4  public class Main {
5  public static void main(String[] args) {
6      // Declaracions
7      double nombre;
8
9      // Demanam i llegim el nombre
10     System.out.print("Introdueix un nombre real: ");
11     nombre = LT.readDouble();
12
13     // Mostrem el missatge adient
14     if (nombre > -1.0 && nombre < 2.0) {
15         System.out.println(nombre + " és entre -1.0 i 1.0.");
16     } else {
17         System.out.println(nombre + " no és entre -1.0 i 1.0.");
18     }
19 }
20 }
```

Mida màxima dels fitxers: 10KB, nombre màxim de fitxers: 1

Fitxers

Podeu arrossegar i deixar anar fitxers aquí per afegir-los

Fitxers java

Comprova

M'he equivocat!

Es pot amagar a *Hide check* a la pregunta

Exemple CodeRunner – Resultat

| | Input | Expected | Got |
|---|--------|---|---------------|
| ✓ | 0 | Introdueix un nombre real: 0.0 és entre -1.0 i 1.0. | Introdueix un |
| ✓ | -0.5 | Introdueix un nombre real: -0.5 és entre -1.0 i 1.0. | Introdueix un |
| ✗ | 1.0 | Introdueix un nombre real: 1.0 no és entre -1.0 i 1.0. | Introdueix un |
| ✓ | 0.999 | Introdueix un nombre real: 0.999 és entre -1.0 i 1.0. | Introdueix un |
| ✓ | -1.234 | Introdueix un nombre real: -1.234 no és entre -1.0 i 1.0. | Introdueix un |

Your code must pass all tests to earn any marks. Try again.

Show differences

< >

Incorrecte

Puntuacions per a aquesta tramesa: 0,00 / 1,00.

Exemple CodeRunner – Resultat

| | Input | Expected | Got |
|---|--------|---|---------------|
| ✓ | 0 | Introdueix un nombre real: 0.0 és entre -1.0 i 1.0. | Introdueix un |
| ✓ | -0.5 | Introdueix un nombre real: -0.5 és entre -1.0 i 1.0. | Introdueix un |
| ✓ | 1.0 | Introdueix un nombre real: 1.0 no és entre -1.0 i 1.0. | Introdueix un |
| ✓ | 0.999 | Introdueix un nombre real: 0.999 és entre -1.0 i 1.0. | Introdueix un |
| ✓ | -1.234 | Introdueix un nombre real: -1.234 no és entre -1.0 i 1.0. | Introdueix un |

Passed all tests! ✓

< >

Correcte

Puntuacions per a aquesta tramesa: 1,00 / 1,00.

Es poden atorgar punts parcials.

Es pot fer una penalització per intents múltiples.

Exemple CodeRunner – Solució

Question author's solution (Java):

```
1  /*
2  Aquest programa llegeix un nombre del teclat i diu si és positiu o r
3  1. Demanam i llegim el nombre real.
4  2. Si és entre -1.0 i 1.0 (no inclosos), ho deim; si no, deim que no
5  */
6  public class Main {
7      public static void main(String[] args) {
8
9          // Declaracions
10         double nombre;
11
12         // Demanam i llegim el nombre
13         System.out.print("Introdueix un nombre real: ");
14         nombre = LT.readDouble();
15
16         // Mostram el missatge adient
17         if (nombre > -1.0 && nombre < 1.0) {
18             System.out.println(nombre + " és entre -1.0 i 1.0.");
19         } else {
20             System.out.println(nombre + " no és entre -1.0 i 1.0.");
21         }
22     }
23 }
```

Si s'ha marcat mostrar la solució, l'alumne pot veure-la.

Exemple CodeRunner – Mètodes

| | | |
|---------------------|---|--|
| Test case 1 | ? | <pre>System.out.println(suma(3, 4));</pre> |
| Standard Input | ? | |
| Expected output | ? | 7 |
| Extra template data | ? | |

Personalització

- ◆ El tipus de pregunta és un *template* Twig.
- ◆ El resultat serà un programa que pugui entendre CodeRunner (C, Java, Python...).
- ◆ Podem fer un programa que interactuï amb els fitxers, compili el programa, l'executi i comprovi la resposta.
 - ◆ Així es poden incorporar més llenguatges (com R).
- ◆ Es pot fer el que sigui si sabem com fer-ho.
 - ◆ Realment és molt, molt potent.
- ◆ Però tampoc fa falta.

Personalització – package de Java

- ◆ Té problemes amb els *package* de Java.
- ◆ Solució: script python que:
 - ◆ Crea un directori *src*.
 - ◆ Hi escriu la solució:
 - ◆ De la solució, en cerca el nom de la classe.
 - ◆ Canvia el text *package* per *// package*.
 - ◆ L'escriu a *<classe>.java*.
 - ◆ Hi escriu els fitxers adjunts:
 - ◆ Els llegeix, canvia *package* per *// package* i els escriu.
 - ◆ Crida *javac -d dist <fitxers>*.
 - ◆ Crida *java -cp dist Main*.

Personalització – fitxers

- ◆ Volem que escrigui en un fitxer un contingut que depèn del d'un altre fitxer.
 - ◆ Posarem a sortida esperada el contingut que esperam.
 - ◆ Posarem a extra el contingut del fitxer d'entrada.
- ◆ Solució: script python que fa l'anterior i:
 - ◆ Escriu a *entrada.txt* l'*extra* de la prova.
 - ◆ Executa el programa.
 - ◆ Llegeix i imprimeix els continguts de *sortida.txt*.

Personalització – mètodes

- ◆ En comptes de posar a mà tots els casos a provar, podem iterar un bucle i provar el mètode moltes vegades.
- ◆ Solució: modificam `Java_Method` per:
 - ◆ incloure *Global Extra* a la classe.
 - ◆ executar l'*extra* de cada prova.
- ◆ *Global Extra* conté (una altra vegada) la resposta correcta i a l'*extra* de la prova hi podem posar el codi que vulguem que faci servir aquest extra.

Personalització – mètodes

| | |
|---------------------|--|
| Test case 1 | <pre>// Aquí provarem 1000 nombres. // Si alguna cosa falla, se't dirà què ha fallat.</pre> |
| Standard Input | |
| Expected output | |
| Extra template data | <pre>for (int i = 1; i <= 1000; i++) { boolean expected = resposta(i); boolean got = esPrimer(i); if (expected != got) { System.out.printf("S'esperava esPrimer(%1\$d) = %2\$b, però esPrimer(%1\$d) = %3\$b.%n", i, expected, got); break; } }</pre> |

resposta és a Global extra i conté la resposta correcta.

esPrimer és la resposta de l'alumne.

Per $i = 1 \dots 1000$, si la resposta no és l'esperada, mostrem l'error i interrompem la prova.

Classificació d'exercicis fets

- ◆ Això no ho tenim, i crec que els va motivar molt.
- ◆ Vaig fer una extensió senzilla de Moodle.
 - ◆ Classificació d'una *categoria d'avaluació*.
 - ◆ Podem crear categories que agrupen alguns elements.
 - ◆ Cada exercici de cada qüestionari val un punt.
 - ◆ A la categoria *exercicis_fets* es sumen tots aquests punts.
 - ◆ També es pot fer servir per altres coses (p. ex.: gamificació).
 - ◆ Però el CTI s'estima més no posar-la en marxa si no ha estat validada per Moodle.
 - ◆ I Moodle encara no ho ha pogut fer.
 - ◆ Si ens interessa a uns quants...

Classificació d'exercicis fets

Exercicis fets   ▼

| # | Participant | Exercicis |
|---|-----------------|-----------|
| 1 | Antoni 2 Alumne | 2 |
| 2 | Antoni Alumne | 1 |

Punts de gamificació   ▼

| # | Participant | Punts |
|---|-----------------|-------|
| 1 | Antoni Alumne | 10.0 |
| 2 | Antoni 2 Alumne | 0.0 |

Més informació

1. replit: <https://replit.com/>.
2. Moodle de CodeRunner: <https://coderunner.org.nz/> (exemples, fòrum).
3. Documentació tècnica: https://github.com/trampgeek/moodle-qtype_coderunner.
4. L'extensió CodeRunner: https://moodle.org/plugins/qtype_coderunner.
5. L'extensió GradeRanking: https://moodle.org/plugins/block_graderanking (encara no ha estat validada).

Vos hi animau?