2.9 Programación: asistencia en tarefas seleccionadas

Aprender a programar sen que a IA escriba o código

Materia: Programación · **Curso:** 9.º (14–15 anos) · **Duración:** 1 sesión dobre (90 min) · **IA:** Asistencia **COA** IA

Competencias relacionadas coa IA (marco DUAL.AI.TEACHer) – nivel docente	**Obxectivos de aprendizaxe específicos da materia – nivel do alumnado**
AF-TL-2b *(Fundamentos e aplicacións da IA × Ensino e aprendizaxe, nivel 2 – Implementación reflexiva)*: «Os docentes poden organizar elementos da sesión que integren o uso da IA con funcións pedagóxicas claras e estratexias para verificar os resultados.» **AF-FC-2b** *(Fundamentos e aplicacións da IA × Fomento da competencia dixital (en IA) do alumnado, nivel 2 – Implementación reflexiva)*: «Os docentes poden aplicar unha orientación que axude o alumnado a desenvolver habilidades prácticas en IA, adaptando as tarefas aos coñecementos previos do alumnado e ás ferramentas de IA dispoñibles.»	Ao final da sesión, o alumnado é capaz de: • saber usar a IA para realizar tarefas repetitivas ou de apoio na programación, como sangrar o código ou escribir os comentarios; • pedirlle a un modelo de IA que propoña melloras no código sen que lle dea o propio código, para que o alumnado poida autoavaliar os seus coñecementos; • avaliar criticamente o modelo de IA cando lle pregunta como optimizaría o código dado.

> **Mensaxe clave:** O alumnado necesita coñecer os fundamentos da programación se quere avaliar os resultados que ofrece un modelo. Por iso, nos primeiros pasos da aprendizaxe da programación, o alumnado debe usar os modelos de IA para realizar tarefas de apoio e repetitivas relacionadas coa programación, pero non para producir código. Tamén pode usar o modelo para pedir posibles melloras no código e así poñer a proba as súas capacidades de programación.

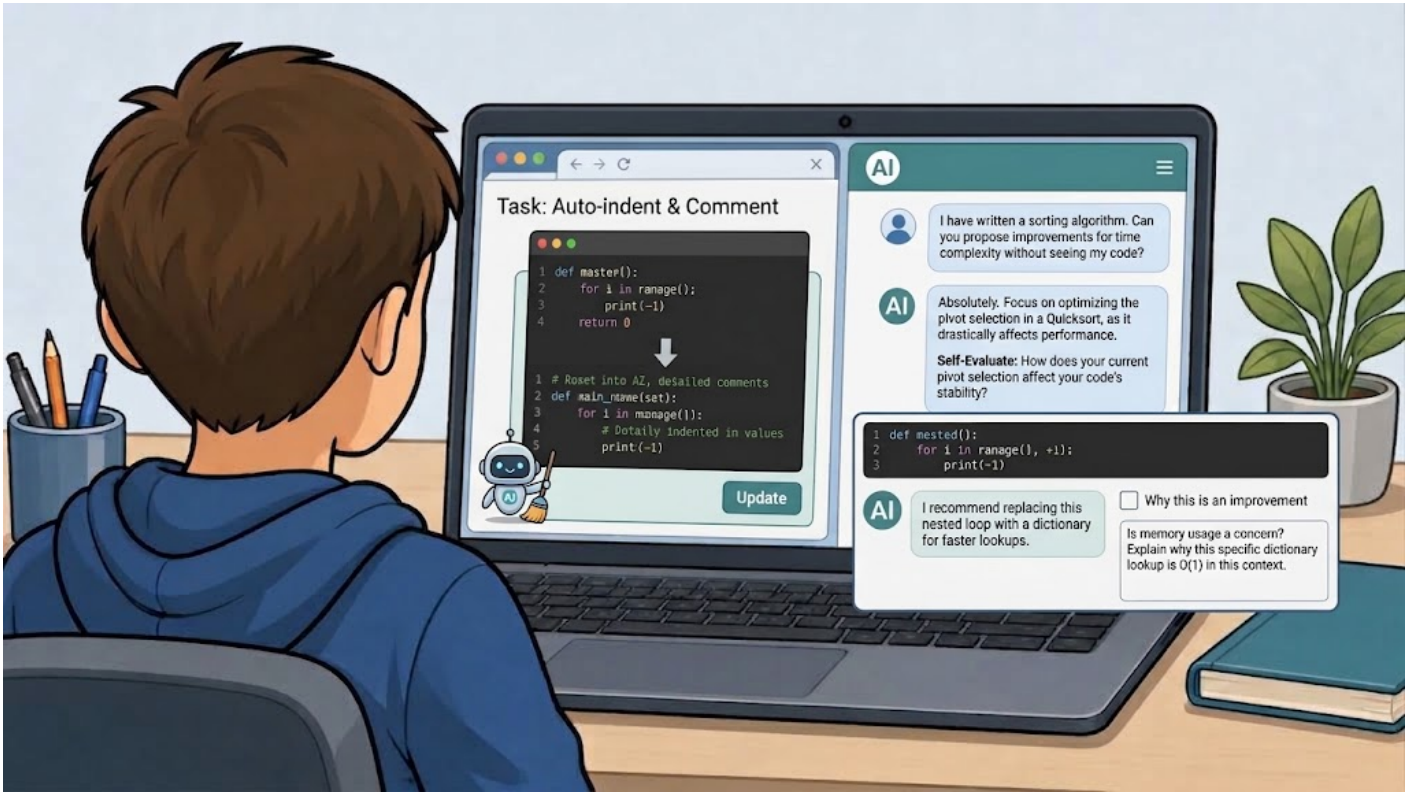


Fig. 1: «Aprender a programar sen que a IA escriba o código» (imaxe xerada con IA, Google Gemini 3.6 Thinking)

Contido

O ensino da programación está a cambiar nun contexto no que os sistemas de IA poden xerar, explicar e revisar código. Aínda que estas ferramentas ofrecen novas oportunidades de aprendizaxe, tamén introducen un desafío educativo importante: o alumnado necesita coñecementos de programación suficientes para avaliar se unha suxestión xerada por IA é correcta, eficiente ou mesmo pertinente para a tarefa en cuestión. Sen unha comprensión básica dos conceptos de programación, o alumnado pode aceptar solucións incorrectas, pasar por alto erros ou non recoñecer alternativas mellores. Por este motivo, aprender os fundamentos da programación segue sendo esencial, mesmo cando hai ferramentas de IA dispoñibles.

Nos cursos de iniciación á programación, a IA debería situarse como unha ferramenta de apoio á aprendizaxe e non como un programador substituto. O alumnado pode usar a IA de forma eficaz para tarefas repetitivas ou de baixo nivel que non reducen o esforzo cognitivo que supón aprender a programar. Por exemplo: dar formato e sangrar o código, xerar comentarios de documentación, explicar mensaxes de erro, resumir regras de sintaxe ou axudar a comprender a finalidade de determinadas construcións de programación. Estes usos permiten que o alumnado centre a súa atención na lóxica subxacente, nos algoritmos e nos procesos de resolución de problemas, que son centrais na competencia en programación.

Outro uso produtivo da IA é a revisión de código. En lugar de pedirlle a un modelo de IA que escriba unha solución, o alumnado pode escribir primeiro o seu propio código e despois pedirlle ao

modelo que identifique posibles melloras. A IA pode suxerir nomes de variables máis claros, unha mellor organización do código, menos redundancia, maior lexibilidade ou enfoques alternativos que o alumnado pode avaliar de forma autónoma. Isto converte a IA nunha compañeira de retroalimentación que permite ao alumnado comparar o seu propio razoamento con suxestións externas e valorar os puntos fortes e débiles das súas solucións.

Con todo, a retroalimentación xerada por IA non debe aceptarse sen espírito crítico. Os modelos poden recomendar cambios innecesarios, ineficientes ou baseados en supostos incorrectos sobre a finalidade do código. Por iso, un obxectivo central desta sesión é desenvolver a capacidade do alumnado para avaliar criticamente as suxestións da IA. O alumnado aprenderá a preguntarlle a un modelo de IA non só que melloras propón, senón tamén por que considera que eses cambios son beneficiosos. Ao comparar o razoamento do modelo cos principios de programación consolidados, o alumnado pode xulgar se a optimización proposta mellora de verdade o código ou se só representa unha opción de deseño diferente.

Esta sesión trata a IA como unha ferramenta que apoia a reflexión, a autoavaliación e a mellora da calidade do código, e non a xeración de código. O alumnado practica escribindo os seus propios programas, usando a IA para tarefas de apoio, pedindo retroalimentación sobre o seu traballo e analizando criticamente a calidade das respostas que recibe. O obxectivo final é axudar o alumnado a ser mellor programador e, á vez, un usuario máis informado dos sistemas de IA.

Programación da sesión

Fase	Tempo	Actividade
Introdución	10 min	O docente presenta dúas solucións breves en código para o mesmo problema, unha escrita por un alumno e outra xerada por un modelo de IA. O alumnado comenta que solución parece máis clara, máis eficiente ou máis fácil de entender.
Presentación	10 min	O docente presenta usos axeitados da IA no ensino da programación, salientando a distinción entre tarefas de apoio, revisión de código e xeración de código.
Exploración	15 min	Por parellas, o alumnado traballa nun pequeno exercicio de programación e identifica que tarefas se poderían delegar adecuadamente nun asistente de IA e cales deben seguir sendo responsabilidade súa.
Ciclo de mellora	35 min	O alumnado escribe a súa propia solución e completa de 2 a 3 roldas de escribir → preguntar → avaliar → revisar, usando a IA só para tarefas de apoio e suxestións de mellora.
Reflexión	20 min	Debate con toda a clase sobre a utilidade e as limitacións da retroalimentación da IA, centrado en como determinou o alumnado se as suxestións eran valiosas ou non.

Preguntas guía: engadir complexidade por capas

O alumnado realiza tantas roldas como permita o tempo durante o ciclo de mellora de 35 minutos. Cada rolda segue o mesmo proceso de catro pasos: escribir → preguntar → avaliar → revisar

1. **Rolda 1 — Usar a IA para tarefas de apoio**
   - Escribir: Complete de forma autónoma un breve exercicio de programación.
   - Preguntar: Pídalle á IA axuda cunha tarefa relacionada co código que non sexa de programación, como:
     - mellorar o sangrado,
     - xerar comentarios,
     - explicar unha mensaxe de erro,
     - describir o que fai unha función.
   - Avaliar: A explicación da IA describe con exactitude o código? Son útiles e comprensibles os comentarios xerados?
   - Revisar: Incorpore os cambios útiles, asegurándose de que a lóxica do programa segue sendo enteiramente obra súa.
2. **Rolda 2 — Pedir suxestións de mellora**
   - Escribir: Revise o programa completado e identifique partes que se poderían mellorar.
   - Preguntar: Pida suxestións de mellora sen permitir que a IA reescriba o código. Por exemplo:

«Suxire tres formas de mellorar a lexibilidade e a mantibilidade deste programa, pero non proporciones código de substitución.»

1.
   - Avaliar: Que suxestións parecen valiosas? Cales terían pouco impacto no programa?
   - Revisar: Aplique as melloras coas que estea de acordo e comprobe se o programa segue funcionando correctamente.
2. **Rolda 3 — Avaliar criticamente os consellos de optimización**
   - Escribir: Escolla unha sección do seu código que realice unha tarefa repetitiva ou que use bucles.
   - Preguntar: Pida suxestións de optimización e esixa que a IA xustifique as súas recomendacións.

Exemplo de instrución (prompt):

«Explica como se podería optimizar este código e por que cada optimización proposta melloraría o rendemento ou a lexibilidade. Non proporciones o código optimizado.»

-
  - Avaliar: Son realmente beneficiosas as optimizacións propostas? Cales se apoian nun razoamento claro? Cales parecen innecesarias ou difíciles de xustificar?
  - Revisar: Decida que recomendacións aplicar e explique o seu razoamento.

Preguntas para a reflexión:

- Que suxestións da IA foron máis útiles para mellorar o seu código?
- Fíxolle a IA algunha suxestión coa que non estivese de acordo? Por que?
- Como determinou se unha optimización pagaba a pena?

- En que momento do proceso de programación é máis útil a IA?
- Que tarefas de programación debería seguir realizando o alumnado?
- Por que son necesarios coñecementos de programación para avaliar a retroalimentación xerada por IA?
- Como pode o alumnado evitar depender en exceso das ferramentas de IA cando aprende a programar?

Pregunta final (para a fase de reflexión):

Cantos coñecementos de programación fan falta para determinar se unha suxestión da IA mellora realmente un programa, e que riscos xorden cando os usuarios confían nos consellos xerados por IA sen entender eles mesmos o código?

Materiais

- Un portátil por parella de alumnos cun contorno de programación
- Acceso a un asistente de chat de IA ou a un modelo de linguaxe aloxado localmente
- Un breve exercicio de programación axeitado ao nivel de experiencia do alumnado
- Unha ficha que siga o ciclo escribir → preguntar → avaliar → revisar
- Preguntas de reflexión centradas na alfabetización en IA e na avaliación crítica
- Exemplos opcionais de código que ilustren boas e malas prácticas de programación

---

Revision #3
Created 2026-10-20 07:19:17 UTC by Flavia
Updated 2026-10-29 14:15:29 UTC by Jannik