



1. Identificación da programación

Centro educativo

Código	Centro	Concello	Ano académico
36018173	IES de Teis	Vigo	2025/2026

Ciclo formativo

Código da familia profesional	Familia profesional	Código do ciclo formativo	Ciclo formativo	Grao	Réxime
IFC	Informática e comunicacións	CSIFC03	Desenvolvemento de aplicacións web	Ciclos formativos de grao superior	Réxime de proba libre

Módulo profesional e unidades formativas de menor duración (*)

Código MP/UF	Nome	Curso	Sesións semanais	Horas anuais	Sesións anuais
MP0487	Contornos de desenvolvemento	2025/2026	0	107	0

(*) No caso de que o módulo profesional estea organizado en unidades formativas de menor duración

Profesorado responsable

Profesorado asignado ao módulo	DAVID GELPI FLETA,MANUEL ÁNGEL REIJA LAMAS,PATRICIA GONZÁLEZ PARDO,PEDRO DAVID GONZÁLEZ VÁZQUEZ,SERGIO RODRÍGUEZ CAL,GUSTAVO LUIS RODRIGUEZ GONZALEZ (Subst.),SOFÍA MARÍN GÓMEZ (Subst.)
Outro profesorado	

Estado: Pendente de supervisión equipo directivo



2. Resultados de aprendizaxe e criterios de avaliación

2.1. Primeira parte da proba

2.1.1. Resultados de aprendizaxe do currículo que se tratan

Resultados de aprendizaxe do currículo
RA1 - Recoñece os elementos e as ferramentas que interveñen no desenvolvemento dun programa informático, e analiza as súas características e as fases en que actúan ata chegar á súa posta en funcionamento.
RA2 - Avalía contornos integrados de desenvolvemento, e analiza as súas características para editar código fonte e xerar executables.
RA3 - Verifica o funcionamento de programas, para o que diseña e realiza probas.
RA4 - Optimiza código empregando as ferramentas dispoñibles no contorno de desenvolvemento.
RA5 - Xera diagramas de clases e valora a súa importancia no desenvolvemento de aplicacións, empregando as ferramentas dispoñibles no contorno.
RA6 - Xera diagramas de comportamento e valora a súa importancia no desenvolvemento de aplicacións, empregando as ferramentas dispoñibles no contorno.

2.1.2. Criterios de avaliación que se aplicarán para a verificación da consecución dos resultados de aprendizaxe por parte do alumnado

Criterios de avaliación do currículo
CA1.1 Recoñeceuse a relación dos programas cos compoñentes do sistema informático: memoria, procesador, periféricos, etc.
CA1.2 Identificáronse as fases de desenvolvemento dunha aplicación informática.
CA1.3 Diferenciáronse os conceptos de código fonte, obxecto e executable.
CA1.4 Recoñecéronse as características da xeración de código intermedio para a súa execución en máquinas virtuais.
CA1.5 Clasificáronse as linguaxes de programación.
CA1.6 Avaliouse a funcionalidade das ferramentas utilizadas en programación.
CA2.1 Instaláronse contornos de desenvolvemento, propietarios e libres.



ANEXO III
MODELO DE PROGRAMACIÓN DE PROBA LIBRE DE MÓDULOS PROFESIONAIS

Criterios de avaliación do currículo
CA2.2 Engadíronse e elimináronse módulos no contorno de desenvolvemento.
CA2.3 Personalizouse e automatizouse o contorno de desenvolvemento.
CA2.4 Configurouse o sistema de actualización do contorno de desenvolvemento.
CA2.5 Xeráronse executables a partir de código fonte de diferentes linguaxes nun mesmo contorno de desenvolvemento.
CA2.6 Xeráronse executables a partir dun mesmo código fonte con varios contornos de desenvolvemento.
CA2.7 Identificáronse as características comúns e específicas de diversos contornos de desenvolvemento.
CA3.1 Identificáronse os tipos de probas.
CA3.2 Definíronse casos de proba.
CA3.3 Identificáronse as ferramentas de depuración e proba de aplicacións ofrecidas polo contorno de desenvolvemento.
CA3.4 Utilizáronse ferramentas de depuración para definir puntos de ruptura e seguimento.
CA3.5 Utilizáronse as ferramentas de depuración para examinar e modificar o comportamento dun programa en tempo de execución.
CA3.6 Efectuáronse probas unitarias de clases e funcións.
CA3.7 Executáronse probas automáticas.
CA3.8 Documentáronse as incidencias detectadas.
CA4.1 Identificáronse os patróns de refactorización máis usuais.
CA4.2 Elaboráronse as probas asociadas á refactorización.
CA4.3 Revisouse o código fonte usando un analizador de código.
CA4.4 Identificáronse as posibilidades de configuración dun analizador de código.



Criterios de avaliación do currículo
CA4.5 Aplicáronse patróns de refactorización coas ferramentas que proporciona o contorno de desenvolvemento.
CA4.6 Realizouse o control de versións integrado no contorno de desenvolvemento.
CA4.7 Utilizáronse ferramentas do contorno de desenvolvemento para documentar as clases.
CA5.1 Identificáronse os conceptos básicos da programación orientada a obxectos.
CA5.2 Recoñecéronse os métodos de análise e modelaxe máis empregados no ámbito do desenvolvemento de aplicacións informáticas.
CA5.3 Identificáronse as ferramentas para a elaboración de diagramas de clases.
CA5.4 Interpretouse o significado de diagramas de clases.
CA5.5 Instalouse o módulo do contorno integrado de desenvolvemento que permite o uso de diagramas de clases.
CA5.6 Trazáronse diagramas de clases a partir das especificacións destas.
CA5.7 Xerouse código a partir dun diagrama de clases.
CA5.8 Xerouse un diagrama de clases mediante enxeñaría inversa.
CA6.1 Identificáronse os tipos de diagramas de comportamento.
CA6.2 Recoñeceuse o significado dos diagramas de casos de uso.
CA6.3 Interpretáronse diagramas de interacción.
CA6.4 Elaboráronse diagramas de interacción sinxelos.
CA6.5 Interpretouse o significado de diagramas de actividades.
CA6.6 Elaboráronse diagramas de actividades sinxelos.
CA6.7 Interpretáronse diagramas de estados.



Criterios de avaliación do currículo

CA6.8 Formuláronse diagramas de estados sinxelos.

2.2. Segunda parte da proba

2.2.1. Resultados de aprendizaxe do currículo que se tratan

Resultados de aprendizaxe do currículo

RA1 - Recoñece os elementos e as ferramentas que interveñen no desenvolvemento dun programa informático, e analiza as súas características e as fases en que actúan ata chegar á súa posta en funcionamento.

RA2 - Avalía contornos integrados de desenvolvemento, e analiza as súas características para editar código fonte e xerar executables.

RA3 - Verifica o funcionamento de programas, para o que diseña e realiza probas.

RA4 - Optimiza código empregando as ferramentas dispoñibles no contorno de desenvolvemento.

RA5 - Xera diagramas de clases e valora a súa importancia no desenvolvemento de aplicacións, empregando as ferramentas dispoñibles no contorno.

RA6 - Xera diagramas de comportamento e valora a súa importancia no desenvolvemento de aplicacións, empregando as ferramentas dispoñibles no contorno.

2.2.2. Criterios de avaliación que se aplicarán para a verificación da consecución dos resultados de aprendizaxe por parte do alumnado

Criterios de avaliación do currículo

CA1.1 Recoñeceuse a relación dos programas cos compoñentes do sistema informático: memoria, procesador, periféricos, etc.

CA1.2 Identificáronse as fases de desenvolvemento dunha aplicación informática.

CA1.3 Diferenciáronse os conceptos de código fonte, obxecto e executable.



Criterios de avaliación do currículo
CA1.4 Recoñecéronse as características da xeración de código intermedio para a súa execución en máquinas virtuais.
CA1.5 Clasificáronse as linguaxes de programación.
CA1.6 Avaliouse a funcionalidade das ferramentas utilizadas en programación.
CA2.1 Instaláronse contornos de desenvolvemento, propietarios e libres.
CA2.2 Engadíronse e elimináronse módulos no contorno de desenvolvemento.
CA2.3 Personalizouse e automatizouse o contorno de desenvolvemento.
CA2.4 Configurouse o sistema de actualización do contorno de desenvolvemento.
CA2.5 Xeráronse executables a partir de código fonte de diferentes linguaxes nun mesmo contorno de desenvolvemento.
CA2.6 Xeráronse executables a partir dun mesmo código fonte con varios contornos de desenvolvemento.
CA2.7 Identificáronse as características comúns e específicas de diversos contornos de desenvolvemento.
CA3.1 Identificáronse os tipos de probas.
CA3.2 Definíronse casos de proba.
CA3.3 Identificáronse as ferramentas de depuración e proba de aplicacións ofrecidas polo contorno de desenvolvemento.
CA3.4 Utilizáronse ferramentas de depuración para definir puntos de ruptura e seguimento.
CA3.5 Utilizáronse as ferramentas de depuración para examinar e modificar o comportamento dun programa en tempo de execución.
CA3.6 Efectuáronse probas unitarias de clases e funcións.
CA3.7 Executáronse probas automáticas.
CA3.8 Documentáronse as incidencias detectadas.



Criterios de avaliación do currículo
CA4.1 Identificáronse os patróns de refactorización máis usuais.
CA4.2 Elaboráronse as probas asociadas á refactorización.
CA4.3 Revisouse o código fonte usando un analizador de código.
CA4.4 Identificáronse as posibilidades de configuración dun analizador de código.
CA4.5 Aplicáronse patróns de refactorización coas ferramentas que proporciona o contorno de desenvolvemento.
CA4.6 Realizouse o control de versións integrado no contorno de desenvolvemento.
CA4.7 Utilizáronse ferramentas do contorno de desenvolvemento para documentar as clases.
CA5.1 Identificáronse os conceptos básicos da programación orientada a obxectos.
CA5.2 Recoñecéronse os métodos de análise e modelaxe máis empregados no ámbito do desenvolvemento de aplicacións informáticas.
CA5.3 Identificáronse as ferramentas para a elaboración de diagramas de clases.
CA5.4 Interpretouse o significado de diagramas de clases.
CA5.5 Instalouse o módulo do contorno integrado de desenvolvemento que permite o uso de diagramas de clases.
CA5.6 Trazáronse diagramas de clases a partir das especificacións destas.
CA5.7 Xerouse código a partir dun diagrama de clases.
CA5.8 Xerouse un diagrama de clases mediante enxeñaría inversa.
CA6.1 Identificáronse os tipos de diagramas de comportamento.
CA6.2 Recoñeceuse o significado dos diagramas de casos de uso.
CA6.3 Interpretáronse diagramas de interacción.



Criterios de avaliación do currículo
CA6.4 Elaboráronse diagramas de interacción sinxelos.
CA6.5 Interpreouse o significado de diagramas de actividades.
CA6.6 Elaboráronse diagramas de actividades sinxelos.
CA6.7 Interpretáronse diagramas de estados.
CA6.8 Formuláronse diagramas de estados sinxelos.

3. Mínimos exixibles para alcanzar a avaliación positiva e os criterios de cualificación

Os mínimos exixibles son os que se detallan a continuación :

BC1. Desenvolvemento de software

Concepto de programa informático.

Fases do desenvolvemento dunha aplicación: análise, deseño, codificación, probas, documentación, explotación e mantemento, etc.

Código fonte, código obxecto e código executable. Máquinas virtuais e linguaxe intermedia.

Tipos de linguaxes de programación.

Características das linguaxes máis difundidas.

Proceso de obtención de código executable a partir do código fonte: ferramentas implicadas.

BC2. Instalación e uso de contornos de desenvolvemento

Funcións dun contorno de desenvolvemento.

Instalación dun contorno de desenvolvemento.

Ferramentas e asistentes do contorno.

Instalación e desinstalación de módulos adicionais.

Personalización do contorno.

Mecanismos de actualización.



Contornos de desenvolvemento máis empregados (libres e comerciais).

Uso básico dun contorno de desenvolvemento: edición de programas e xeración de executables.

Características dos contornos de desenvolvemento.

BC3. Deseño e realización de probas

Planificación de probas.

Tipos de probas: funcionais, estruturais, regresión, etc.

Procedementos e casos de proba.

Mecanismos e ferramentas de depuración: puntos de ruptura, inspección de variables, etc.

Validacións.

Probas de código: cubrimento, valores límite, clases de equivalencia, etc.

Normas de calidade.

Probas unitarias: ferramentas.

Automatización e documentación das probas.

BC4. Optimización e documentación

Refactorización: concepto, limitacións e patróns máis usuais.

Refactorización e probas. Ferramentas de axuda á refactorización.

Uso e configuración de analizadores de código.

Control de versións. Estrutura das ferramentas de control de versións. Repositorio. Ferramentas de control de versións.

Cientes para control de versións. Integración no contorno de desenvolvemento.

Documentación. Uso de comentarios. Alternativas.

BC5. Elaboración de diagramas de clases

Clases: atributos, métodos e visibilidade.

Métodos de análise e modelaxe.

Obxectos: instanciación.

Relacións: herdanza, composición e agregación.

UML. Diagramas estruturais.

Notación dos diagramas de clases: interpretación.

Ferramentas de deseño de diagramas. Módulos integrados no contorno de desenvolvemento.



Creación de diagramas de clases.

Xeración de código desde o diagrama de clases.

Obtención do diagrama de clases a partir do código.

BC6. Elaboración de diagramas de comportamento

Tipos de diagramas de comportamento: campo de aplicación

Diagramas de casos de uso: actores, escenario e relación de comunicación.

Diagramas de interacción: tipos (diagramas de secuencia e de comunicación).

Diagramas de secuencia.

Diagramas de comunicación.

Diagramas de actividades.

Diagramas de estados.

CRITERIOS DE CUALIFICACIÓN:

- Cada unha das partes cualificarase de 0 a 10 puntos.
- Para superar cada unha das partes deberá obterse un mínimo de 5 puntos en cada unha delas.
- Cada unha das partes terá carácter eliminatorio.
- As persoas que non superen a primeira proba, serán cualificadas cun 0 na segunda parte.
- A cualificación final será a media aritmética das notas obtidas en cada unha das dúas partes, sempre e cando se acade un mínimo de 5 puntos en cada unha das partes.
- No caso das persoas aspirantes que suspendan a segunda parte da proba, a puntuación máxima que poderá asignarse será de catro puntos.
- A valoración de cada unha das preguntas ou cuestións de cada proba especificarase no propio exame.

4. Características da proba e instrumentos para o seu desenvolvemento

4.a) Primeira parte da proba

1. Terá carácter eliminatorio.
2. Será un exame escrito en papel que poderá ter cuestións de distintos tipos:



- Tipo test de multiresposta con só unha opción correcta e puntuando negativamente as incorrectas.
- Preguntas de resposta curta ou resolución/interpretación de pequenos supostos prácticos onde o examinado poderá demostrar os coñecementos relacionados cos resultados de aprendizaxe indicados.
- Preguntas que poderán incluír pseudocódigo e/ou organigramas.

3. A linguaxe de programación será Java.

REQUISITOS

- Será preciso levar DNI, ou outro documento acreditativo da identidade do/a aspirante.
- Non se poderá empregar ningún tipo de documentación.
- Non se permitirá a utilización de dispositivos electrónicos persoais.
- O único instrumento necesario e obrigatorio é un bolígrafo preferentemente azul ou negro. Recoméndase traer un de reposto.

4.b) Segunda parte da proba

- Só se poderá realizar esta proba, se se superou a primeira parte da proba con 5 puntos ou mais.
- Terá carácter eliminatorio.
- Consistirá na resolución ou interpretación dun ou varios supostos prácticos no ordenador ou papel onde a persoa examinada poderá demostrar os coñecementos relacionados cos resultados de aprendizaxe indicados.
- Os programas entregados deben compilar para ser corrixidos.
- A linguaxe de programación a utilizar será Java, usando coma mínimo a versión 21.
- Non se poderá empregar ningún tipo de documentación, salvo a API de Java que estará dispoñible no IDE e ademais en formato offline.

REQUISITOS

- Será preciso levar DNI, o outro documento acreditativo da identidade do/a aspirante.
- Non se poderá empregar ningún outro tipo de documentación que non se especificara anteriormente.
- Non se permitirá a utilización de dispositivos electrónicos persoais.
- O único instrumento necesario e obrigatorio é un bolígrafo preferentemente azul ou negro.