نموذج وصف المادة الدراسية

معلومات المقرر الدراسية			
اسم المقرر	أساسيات البرمجة I	أسلوب التدريس	
نوع المقرر	اساسي	• النظري • العملي	
رمز المقرر	CYS1103		
عدد الوحدات	7		
عدد ساعات المقرر	175		

مستوى المقرر الدراسي	1	الفصل الدراسي	1
القسم الأكاديمي	قسم الامن السيبراني	الكلية	كلية علوم الحاسوب وتكنولوجيا المعلومات
مسؤول المادة	د. علي كريم عبدالرحيم	الايميل	alialmujab@uowa.edu.iq
اللقب العلمي	مدرس	الشهادة الاكاديمية	دكتوراه
مدرس المادة	د. علي كريم عبدالرحيم	الايميل	alialmujab@uowa.edu.iq
اسم مراجع النظير	م.م نبيل صادق	الايميل	nabeel@uowa.edu.iq
تاريخ موافقة اللجنة العلمية	2025-12-24	اصدار	1.0

العلاقة مع المواد الدراسية الأخرى			
	الفصل الدراسي	لا شيء	**وحدة المتطلبات الأساسية**
	الفصل الدراسي	لا شيء	**وحدة المتطلبات المشتركة**

رئيس القسم

عميد الكلية

أهداف المادة الدراسية ونتائج التعلم والمحتويات الإرشادية	
أهداف المادة الدراسية	1. مقدمة في البرمجة: تهدف الوحدة إلى تعريف الطلاب بالمفاهيم والمبادئ الأساسية للبرمجة. يقدم نظرة عامة على لغات البرمجة، وغرضها، ودورها في تطوير البرمجيات. 2. البنى الأساسية البرمجية: تهدف الوحدة إلى تعريف الطلاب بالبنى البرمجية الأساسية مثل المتغيرات، وأنواع البيانات، والمشغلات، والتعبيرات. تركز على تعليمهم كيفية استخدام هذه البنى لكتابة برامج بسيطة. 3. هياكل التحكم: تهدف الوحدة إلى تعريف الطلاب بهياكل التحكم مثل الحلقات والشروط الشرطية. تعلمهم كيفية استخدام هذه الهياكل للتحكم في سير تنفيذ البرنامج واتخاذ القرارات بناء على شروط معينة. 4. الوظائف والإجراءات: تهدف الوحدة إلى تعليم الطلاب عن الوظائف والإجراءات، وغرضها، وكيفية تعريفها واستخدامها في البرمجة. يركز على البرمجة المعيارية وقابلية إعادة استخدام الكود. 5. عمليات الإدخال/الإخراج: تهدف الوحدة إلى تعريف الطلاب بعمليات الإدخال/الإخراج في البرمجة. يغطي تقنيات قراءة المدخلات من المستخدم، وعرض المخرجات، والتفاعل مع الملفات. 6. مهارات حل المشكلات: تهدف الوحدة إلى تطوير مهارات حل المشكلات لدى الطلاب من خلال تقديم تحديات وتمارين برمجية لهم. يؤكد على أهمية تقسيم المشكلات إلى خطوات أصغر، وتصميم الخوارزميات، وتنفيذ الحلول باستخدام بنى البرمجة. 7. التصحيح وحل المشكلات: تهدف الوحدة إلى تزويد الطلاب بمهارات في تحديد وحل الأخطاء البرمجية الشائعة. يعلمهم تقنيات تصحيح الأخطاء، وتتبع تنفيذ البرامج، والتعامل مع الأخطاء بفعالية.
مخرجات التعلم للمادة الدراسية	1. فهم المفاهيم الأساسية للبرمجة: يجب أن يكون الطلاب قادرين على شرح المفاهيم الأساسية للبرمجة، بما في ذلك المتغيرات، وأنواع البيانات، وهياكل التحكم، والوظائف. 2. اكتب وشغل برامج بسيطة: يجب أن يكون الطلاب قادرين على كتابة برامج بسيطة باستخدام لغة برمجة، مما يدل على فهم أساسيات النحو والدلالات. يجب أن يكونوا قادرين على تجميع أو تفسير برامجهم وتنفيذها بنجاح. 3. تطبيق تقنيات حل المشكلات: يجب أن يكون الطلاب قادرين على تحليل وتقسيم المشكلات البسيطة إلى مهام أصغر وأكثر قابلية للإدارة. يجب أن يظهروا القدرة على تصميم الخوارزميات وتنفيذ الحلول باستخدام البنى البرمجية المناسبة. 4. استخدام تراكيب البرمجة بفعالية: يجب أن يكون الطلاب قادرين على استخدام بنى البرمجة مثل الحلقات، والشرطيات، والدوال للتحكم في تدفق البرنامج، واتخاذ القرارات، وأداء المهام المتكررة. 5. تصحيح الأخطاء وحل المشكلات البرامج: يجب أن يكون الطلاب قادرين على التعرف على و تصحيح الأخطاء الشائعة في برامجهم. يجب أن يكونوا قادرين على استخدام تقنيات التصحيح واستخدام استراتيجيات لاستكشاف أخطاء الكود الخاص بهم

	فعليا. 6 . اظهر مهارات معالجة البيانات الأساسية: يجب أن يكون الطلاب قادرين على العمل مع هياكل البيانات الأساسية مثل المصفوفات أو القوائم أو السلاسل النصية. يجب أن يظهروا كفاءة في التلاعب والوصول إلى البيانات المخزنة في هذه الهياكل. 7 . تطبيق عمليات الإدخال/الإخراج: يجب أن يكون الطلاب قادرين على دمج عمليات الإدخال/الإخراج في برامجهم. يجب أن تظهر القدرة على قراءة مدخلات المستخدمين، وعرض المخرجات، والتفاعل مع الملفات حسب الحاجة. 8 . فهم مبادئ تطوير البرمجيات الأساسية: يجب أن يكون لدى الطلاب وعي بمبادئ تطوير البرمجيات مثل تنظيم الكود، وقابلية إعادة استخدام الكود، والقابلية للنمطية. يجب أن يكونوا قادرين على كتابة كود واضح وسهل القراءة وصيانة وفقا لقواعد البرمجة وأفضل الممارسات. 9 . التعاون الفعال في مشاريع البرمجة: يجب على الطلاب إثبات القدرة على العمل بشكل تعاوني في مشروع برمجي، والتواصل الفعال مع أعضاء الفريق، ومشاركة الكود، واستخدام أنظمة التحكم في الإصدارات.
المحتويات الإرشادية	تشمل المحتويات الإرشادية لأساسيات البرمجة المواضيع التالية: 1 . مقدمة في البرمجة: o تعريف وأهمية البرمجة o نظرة عامة على لغات البرمجة واستخداماتها o مقدمة عن لغة برمجة محددة (مثل بايثون) وميزاتها 2 . المتغيرات وأنواع البيانات: o مقدمة عن المتغيرات وغرضها o أنواع البيانات الأساسية (مثل الأعداد الصحيحة، الأعداد العائمة، السلاسل النصية، البوليانات) o إعلان المتغيرات والتعيين 3 . هياكل التحكم: o مقدمة في هياكل التحكم (مثل عبارات if، الحلقات) o العبارات الشرطية (مثل if-else، عبارات إذا المتداخلة) o هياكل التكرار (مثل، حلقة while، حلقة لحلقة) 4 . الوظائف والإجراءات: o تعريف وغرض الوظائف o إعلان الوظيفة والاستدعاء o تمرير الوسائط إلى الدوال وإرجاع القيم o مقدمة في الدوال والمكتبات المحددة مسبقا 5 . حل المشكلات والتفكير الخوارزمي: o فهم وتعريف المشكلات o تقسيم المشكلات إلى مهام أصغر o تطوير الخوارزميات والحلول خطوة بخطوة o ترجمة الخوارزميات إلى كود

استراتيجيات التعلم والتعليم	
الاستراتيجيات	عند تدريس أساسيات البرمجة لطلاب الصف الأول في قسم تكنولوجيا المعلومات، من المهم استخدام استراتيجيات تناسب أعمارهم ومستوى تعلمهم. إليك بعض الاستراتيجيات الفعالة: 1. الأنشطة العملية: استخدم الأنشطة التفاعلية والعملية لإشراك الطلاب بنشاط في عملية التعلم. على سبيل المثال، توفير ألغاز أو ألعاب أو أشياء مادية تمثل مفاهيم البرمجة مثل المتغيرات أو الحلقات. يساعد هذا النهج في جعل المفاهيم المجردة أكثر ملموسة ومتعة. 2. التمثيلات البصرية: استخدم وسائل بصرية مثل الرسوم البيانية أو الرسوم الانسيابية أو الرسوم التوضيحية لمساعدة الطلاب على تصور مفاهيم البرمجة. يمكن للتمثيلات البصرية أن تساعد في فهم تدفق تنفيذ البرامج، والعلاقة بين البنى البرمجية المختلفة، والمنطق وراء الخوارزميات. 3. التلعيب (Gamification): دمج عناصر التلعيب في تمارين البرمجة والواجبات. أنشئ تحديات برمجية، أو مسابقات، أو ألعاب تعليمية تحفز الطلاب على تطبيق مفاهيم البرمجة بشكل إبداعي. يعزز هذا النهج التعلم النشط، وحل المشكلات، والمنافسة الصحية بين الطلاب. 4. التعلم التعاوني: شجع التعلم التعاوني من خلال تسهيل المشاريع الجماعية أو أنشطة البرمجة الزوجية. يعزز التعلم التعاوني التواصل والعمل الجماعي وتبادل الأفكار بين الطلاب. كما تتيح للطلاب التعلم من بعضهم البعض وحل مسائل البرمجة بشكل جماعي. 5. نهج خطوة بخطوة: قسم مفاهيم البرمجة إلى خطوات صغيرة يمكن التحكم بها. ابدأ بأمثلة بسيطة وملموسة قبل الانتقال إلى مواضيع أكثر تعقيدا. قدم تعليمات وشرح واضح، موضحا كل خطوة في العملية. يساعد هذا النهج التدريجي الطلاب على فهم المفاهيم تدريجيا وبناء مهاراتهم البرمجية بفعالية. 6. أمثلة واقعية: اربط مفاهيم البرمجة بسيناريوهات واقعية يمكن للطلاب أن يتفهموا بها. استخدم أمثلة من المواقف اليومية، مثل إنشاء برنامج لحساب التكلفة الإجمالية للعناصر في عربة التسوق أو محاكاة نظام إشارات المرور. ربط البرمجة بالتطبيقات الواقعية يجعلها أكثر صلة وجاذبية للطلاب. 7. الموارد التفاعلية عبر الإنترنت: استخدم الموارد التفاعلية عبر الإنترنت، ألعاب البرمجة التعليمية، أو منصات برمجة صديقة للأطفال مصممة خصيصا للمتعلمين الصغار. غالبا ما توفر هذه الموارد دروسا تفاعلية، وبيئات برمجة بصرية، وتغذية راجعة فورية، مما يجعل تجربة التعلم أكثر تفاعلية ومتعة. 8. الدعم الفردي: قدم دعما فرديا وملاحظات للطلاب. قدم المساعدة لمن يواجهون صعوبات وقدم تحديات إضافية لمن يفهمون المفاهيم بسرعة. بانتظام

	تقييم تقدم الطلاب ومعالجة احتياجاتهم التعليمية الخاصة لضمان تقدمهم المستقر. 9. شجع الإبداع: عزز الإبداع من خلال تشجيع الطلاب على التفكير الإبداعي وإيجاد حلول مبتكرة لمشاكل البرمجة. وفر لهم فرصا لتطبيق مفاهيم البرمجة في المشاريع الإبداعية، مثل تصميم ألعاب بسيطة أو إنشاء الرسوم المتحركة. يشجع هذا النهج على التفكير النقدي، وحل المشكلات، واستكشاف أفكارهم الخاصة. 10. الممارسة التأملية: دمج أنشطة التأمل والتقييم الذاتي في عملية التعلم. شجع الطلاب على مراجعة كودهم الخاص، وتحديد مجالات التحسين، والتأمل في أساليبهم في حل المشكلات. تساعد هذه الممارسة التأملية الطلاب على تطوير فهم أعمق لمفاهيم البرمجة وتحسن قدرتهم على تحليل وتصحيح الشيفرة.

الحمل الدراسي للطالب محسوب لـ 15 اسبوعا

SWL الهيكلي (h/sem) الحمل الدراسي المنتظم للطالب خلال الفصل	78	**SWL المنظم (h/w)** الحمل الدراسي المنتظم للطالب أسبوعيا	5
SWL منظم غير (h/sese) الحمل الدراس غير المنتظم للطالب خلال الفصل	97	**SWL منظم غير (h/w)** الحمل الدراسي غير المنتظم للطالب أسبوعيا	7
SWL إجمالي (الدراسية الحصص) الحمل الدراسي الكلي للطالب خلال الفصل	175		

تقييم المقرر الدراسي

		الوقت/العدد	الوزن (الدرجات)	الأسابيع	مخرجات التعلم
التقويم التكويني	اختبارات	5	8% (8)	4,11	4,6
	مشاريع وتقارير	1	5% (5)	13	10,12
	مختبر	10	15% (15)	3,5,7,9,12	3,4,5,8,9
	واجبات بيتية	5	7% (7)	6	1,2,4,10
	نشاطات صفية	5	5% (5)		
التقييم التلخيصي	امتحان منتصف الفصل	1	10% (10)	7	1 – 13
	امتحان النهائي	3	50% (50)	16	الكل
إجمالي التقييم				100	

المنهاج الاسبوعي النظري	
	المواد المغطاة
الأسبوع 1	مقدمة في أساسيات البرمجة
الأسبوع 2	تعلم كيف تغير المشكلة إلى خوارزمية
الأسبوع 3	رمز زائف ومخطط تدفق
الأسبوع 4	المتغيرات، عبارات التكليف، والتعبيرات
الأسبوع 5	مشغلات التعيين المعززة
الأسبوع 6	ترميز الأحرف باستخدام ASCII
الأسبوع 7	**امتحان منتصف الفصل**
الأسبوع 8	دراسة حالة: الحد الأدنى لعدد العملات المعدنية
الأسبوع 9	كتابة التعبيرات البوليانية باستخدام مؤثرات المقارنة
الأسبوع 10	عبارات if والأخطاء الشائعة في بيانات الاختيار
الأسبوع 11	المؤثرات المنطقية وتوليد الأعداد العشوائية
الأسبوع 12	الحلقات والحلقة المتداخلة
الأسبوع 13	الكلمات المفتاحية انكسر واستمر
الأسبوع 14	**التحضير للامتحان النهائي**

المنهاج الاسبوعي للمختبر	
	المواد المغطاة
الأسبوع 1	المختبر 1: مقدمة في أساسيات البرمجة
الأسبوع 2	المختبر 2: البدء مع C++
الأسبوع 3	المختبر 3: قراءة المدخلات من وحدة التحكم
الأسبوع 4	المختبر 4: تعلم المتغيرات
الأسبوع 5	المختبر 5: تعلم عبارات الواجبات والتعبيرات
الأسبوع 6	المختبر 6: قراءة السلاسل النصية من لوحة المفاتيح وترميز الأحرف باستخدام ASCII
الأسبوع 7	**امتحان منتصف الفصل**
الأسبوع 8	المختبر 8: كتابة التعبيرات البوليانية باستخدام مؤثرات المقارنة
الأسبوع 9	المختبر 9: تنفيذ عبارات if

الأسبوع 10	المختبر 10: تعلم كيفية قراءة الأخطاء الشائعة في بيانات الاختيار
الأسبوع 11	المختبر 11: تعلم المشغلات المنطقية
الأسبوع 12	المختبر 12: توليد الأرقام العشوائية
الأسبوع 13	المختبر 13: حلقة while، حلقة ل، وحلقة متداخلة
الأسبوع 14	**التحضير للامتحان النهائي**

مصادر التعلم والتدريس

	نص	**متوفر في المكتبة؟**
النصوص المطلوبة	C++: المرجع الكامل، الطبعة الرابعة هربرت شيلدت ماكجرو-هيل/أوزبورن نيويورك، شيكاغو، سان فرانسيسكو، لشبونة لندن، مدريد، مكسيكو سيتي، ميلانو، نيودلهي، سان خوان سيول سنغافورة سيدني تورونتو	كلا
النصوص الموصى بها	OqeiliSalch, prof. Department of IT-AL-Balqa Applied University	كلا
المواقع الإلكترونية	https://www.w3schools.com/cpp/cpp_intro.asp	

مخطط الدرجات

المجموعة	**الدرجة**	**التقدير**	**الدرجات (٪)**	**التعريف**
مجموعة النجاح (100 - 50)	**أ** - ممتاز	امتياز	100 - 90	أداء متميز
	ب - جيد جدا	جيد جدا	89 - 80	فوق متميز مع بعض الأخطاء
	ج - جيد	جيد	79 - 70	أداء جيد مع أخطاء ملحوظة
	د - متوسط	متوسط	69 - 60	عادل لكنه مع عيوب كبيرة
	هـ - مقبول	مقبول	59 - 50	الجهد يفي بالحد الأدنى من المعايير
مجموعة الرسوب (0 –49)	**و** – راسب	راسب (قيد المعالجة)	(49-45)	يتطلب الأمر المزيد من الجهد لكن تم منح الساعات المعتمدة
	ف – راسب	راسب	(44-0)	يتطلب النجاح جهداً كبيراً

ملاحظة: النقاط العشرية فوق أو تحت 0.5 ستقربها إلى العلامة الكاملة الأعلى أو الأدنى (على سبيل المثال، علامة 54.5 ستقربها إلى 55، بينما العلامة 54.4 ستقربها إلى 54). لدى الجامعة سياسة لا تبرر "الرسوب القريب من النجاح"، لذا فإن التعديل الوحيد على الدرجات الممنوحة من قبل العلامة الأصلية سيكون التقريب التلقائي المذكور أعلاه.