تصميم استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات برمجة اللغات الحديثة لدى طالبات تكنولوجيا التعليم

نجوى عزام أحمد فهمي*

أ.د / عبداللطيف الصفي الجزار د / سماح محمد صابر*****

المستخلص

يهدف هذا البحث إلى تصميم استراتيجية تعليمية جديدة قائمة على تكامل أنماط عرض المشكلة البرمجية لتنمية كفايات برمجة اللغات الحديثة (C++) ببيئة تعلم مدمج، وتقوم الإستراتيجية المطورة على دمج ثلاثة أنماط لعرض المشكلة البرمجية: الألغاز والمصفوفة والإكمال، وقد استخدم الباحثون منهج البحث الوصفي التحليلي ؛ حيث قاموا بمراجعة وتحليل أدبيات وبحوث تكنولوجيا التعليم ذات الصلة لصياغة الأسس النظرية للاستراتيجية المقترحة لكفايات C++ المستهدفة، واستنادًا إلى تلك الأسس النظرية للاستراتيجية، صاغ الباحثون النموذج الأولي للاستراتيجية المقترحة من خمس مراحل: مرحلة الإعداد والتهيئة في بيئة التعلم التقليدية، مرحلة عرض وتكوين المفاهيم والقواعد البرمجية في البيئة التقليدية، مرحلة التدريب على تكوين المفاهيم والقواعد البرمجية من خلال نمط عرض الألغاز (Puzzle) في بيئة التعلم الإلكتروني، مرحلة التدريب على بناء المشكلة وتركيبها الصحيح من خلال نمط المصفوفة (Matrix) في بيئة التعلم الإلكتروني، وأخيرًا، مرحلة التقويم من خلال نمط الإكمال (Completion) في بيئة التعلم الالكتروني. وقد تمت مراجعة هذه الاستراتيجية الأولية المكونة من خمس مراحل، وتم التحُكم عليها من قبل (٩) محكمين مشاركين من أعضاء هيئة التدريس في تكنولوجيا التعليم. تم تنفيذ التعديلات والإقتراحات على النموذج الأولي للاستراتيجية وتمالتوصل إلى الشكل النهائي لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج لتنمية كفايات البرمجة والمكونة من (٥) مراحل و (٤٧) خطوة إجرائية. ثم تحليل بيانات التحكيم ومناقشتها وعرضها، وأظهرض نتائج البحث أن متوسط إتفاق المحكمين علي الإستراتيجية ككل بلغ (٩٥%). وكذلك تم عرض الشكل النهائي للاستراتيجية المطورة وخطة تنفيذها، وتم تضمين الأشكال والمراجع والتوصيات وقائمة الأبحاث المقترحة في هذا التقرير البحثي.

الكلمات المفتاحية: التعلم المدمج، مراحل الإستراتيجية، المشكلة البرمجية، تكامل أنماط العرض، كفايات البرمجة C++، نمط عرض الألغازPuzzle، نمط عرض المصفوفة Matrix.

*معيدة بقسم تكنولوجيا التعليم والمعلومات - كلية البنات - جامعة عين شمس

**أستاذ تكنولوجيا التعليم بقسم تكنولوجيا التعليم والمعلومات كلية البنات - جامعة عين شمس

***مدرس تكنولوجيا التعليم بقسم تكنولوجيا التعليم والمعلومات - كلية البنات - جامعة عين شمس

البريد الالكتروني: nagwa.azzam@women.asu.edu.eg

مقدمــة البحــث:

تهتم البحوث التطويرية في تكنولوجيا التعليم بتصميم الاستراتيجيات وتطوير بيئات التعلم والتدريب وفق متغيرات تصميم مبتكرة تؤدي إلى فعاليتها، وتركز مخرجات تلك البحوث التطويرية على إنتاج المعرفة التي تثري مجال التخصص، والحصول على المنتوجات مثل الاستراتيجيات والبرامج وبيئات التعلم التي تحسن الواقع التطبيقي بتوظيف هذه المخرجات، وتحتل كفايات البرمجة مكانة واضحة في إعداد الأخصائيين في مجال تكنولوجيا التعليم عامةً وتكنولوجيات التعلم الإليكتروني بصفة خاصة. وتعد كفايات برمجة الكمبيوتر Computer Programming حقل من حقول المعرفة التي يعتبرها الطلاب مهمة تعلم صعبة، حيث تعتبر المقررات التي تحمل اسم مثل " مقدمة في البرمجة " مرارات إلزامية للعديد من الطلاب الذين يدرسون في التعليم العالي لتأسيس الطلاب ذوى الكفاءة في التعامل مع تكنولوجيا الاتصال والمعلومات.

وقد انتقل في السنوات الحالية مفهوم البرمجة إلى المدارس الثانوية وحتى الابتدائية، تأكيدًا على أهميتها فهي من متطلبات القرن الحادي و العشرين، وذلك لمواكبة عصر المعلومات في المجتمع العالمي، ولذلك منذ السبعينيات كان الباحثون يقدمون طرقًا مختلفة لتعلم البرمجة، ومع ذلك كانت معظم هذه الأساليب تعتمد على نظرية أو خبرة المعلم، بالإضافة إلى ذلك قام البعض بتطوير نماذج وتم استخدامها محليًا لتقديم نماذج إحصائية تثبت نماذجهم، ولا يوجد دليل على أن هذه النماذج تستخدم في أي مكان أخر، ثم قدم المطورون أدوات ونظم تعمل كمنهج بديل لتدريس واحد لواحد (One To One) لخفض التكلفة، ولكن أيضًا لا يبدو أن هذه الأدوات بدأ استخدامها أو تم استبدال التدريس التقليدي، وفي ضوء هذه المحاولات تم تصنيف البحوث البرمجية إلى ثلاث فئات، كالتالى: أولًا: **مدخل التعليم** Teaching Approach**:** ويتم فيه اعتماد اسلوب أو مجموعة من الأساليب التي تساعد في الوصول إلى نتائج تعليم ناجحة مثل اعتماد أسلوب لنفعلها معًا let us do it (Ngo-ye & Park 2014)،[١] والذي تم إعداده من عدة طرق مثل برمجة الأقران، والتدريبات التفاعلية في الفصول الدراسية، والتدريس العملي، ثانيًا**: نموذج التعليم** Teaching Model**:** ويتكون من منهج واحد أو عدة مناهج، بالإضافة إلى العمليات اللازمة لتطبيق هذه المناهج (How To)، وكيفية تقييم عملية التعلم وكيف يتم دمج التغذية الراجعة في النموذج لتحسين التعلم، بعبارة أخرى المنهج التعليمي جزء من نموذج التعليم مثل منهج (دعونا نفعلها معًا) فإذا تم استخدام هذه الطريقة فالنموذج سيوضح العمليات اللازمة لكيفية تنفيذها. وثالثًا: **أدوات التعليم** Teaching Tool**:** وهي عبارة عن أدوات يتم استخدامها لتعلم البرمجة مثل أداة Scratch والتي يتم استخدامها عبر الإنترنت، وأدوات التعليم لوحدها ليست فعالة فهي لا تحتوى عادة إلا على التمارين والمبادئ التوجيهية وإذا تم استخدامها مع نموذج ستكون أكثر فعالية لتعليم لغات البرمجة (Alajmi & Alkhatib, 2015).

يتطلب تعلم البرمجة الجهد ونهج خاص للتعلم لكي يصبح الطالب مبرمج جيد ، حيث يجب على الطالب الحصول على سلسلة من القدرات التي تتجاوز بكثير معرفة بناء جملة بعض لغات البرمجة، وعلى الرغم من اقتراح العديد من المداخل والأدوات تهدف إلى دعم تعلم البرمجة بطرق مختلفة ووجود تقارير عن نتائج إيجابية نتيجة بعض الأدوات كما تم الإشارة سابقًا، إلا أنه ليس لأي منها استخدام عام في الواقع، ولذلك تبقى

[١] استخدم الباحثون في التوثيق وكتابة المراجع الإصدار السادس من نظام جمعية علم النفس الأميركية APA Style، ولكنه في المراجع العربية فيتم كتابة أسم المؤلف كاملا بالترتيب العربي: الأسم الأول ثم الثاني ثم اللقب في التوثيق وفي كتابة المراجع.

المشكلة كما هي دون تغيير نسبيًا وهذا ما تثبته التقارير عن الصعوبات التي يواجهها العديد من الطلاب عند تعلم البرمجة الأساسية (Mendes & Gomes, 2007).

كما تبين الدراسات أن المشكلة تبدأ بالنسبة للكثيرين من الطلاب في المرحلة الأولى من التعلم التي تتطلب الفهم وتطبيق مفاهيم البرمجة المجردة وذلك لأن التعلم يتم من خلال نمط واحد تعرض به كافة المشكلات دون التركيز على نوع المشكلة ومدى مناسبتها للنمط، ومكونات اللغة ومراحل تعلمها، أي توحيد طريقة الحصول على المعرفة والقدرات البرمجية، وقد قسمت الأدبيات مشكلات تعلم البرمجة إلى:

(١) **طرق التعليم methods of teaching:** فقد أوضح كلًا من (Alturki, 2016; Kori et al., 2015; Gomes & Mendes, 2014) أن طرق التعليم التقليدية غير ملائمة لاحتياجات العديد من الطلاب، لأسباب مختلفة منها: (أ) طرق التعليم ليست شخصية. (ب) لا تدعم استراتيجيات المعلمين جميع أساليب تعلم الطلاب وتفضيلاتهم. (ج) تدريس المفاهيم الديناميكية من خلال المواد الساكنة. (د) يركز المعلمون بدرجة أكبر على تدريس لغة البرمجة وتفاصيلها وقواعدها اللغوية بدلاً من الترويج لحل المشكلات باستخدام لغة البرمجة، فالغرض من مقررات البرمجة التمهيدية هو زيادة قدرات البرمجة لدى الطلاب. ومع ذلك يركز المعلمون والطلاب على التفاصيل اللغوية في لغة البرمجة. كذلك يجب اختيار اللغة المستخدمة في دورات البرمجة التمهيدية مع الأخذ في الاعتبار الملاءمة التربوية وليس الشعبية في الصناعة أو لسبب آخر.

(٢) **أساليب الدراسة The study methods**: وذلك كما أوضح أيضًا كلًا من (Alturki, 2016; Kori et al., 2015; Gomes & Mendes, 2014) أن أساليب الدراسة التي يتبعها العديد من الطلاب ليست مناسبة لتعلم البرمجة حيث يستخدم الطلاب منهجيات دراسة غير صحيحة: كاستخدام العديد من الطلاب لحل المشكلات من التخصصات الأخرى طريقة حفظ الصيغ أو الإجراءات وحتي في بعض الأحيان يحفظ الطلاب الصيغ دون فهم كامل للمفاهيم الأساسية، مع العلم فقط أن صيغة معينة يجب أن أن تستخدم لنوع من المشكلة، لذلك فالبرمجة تتطلب طريقة دراسة مختلفة، حيث يجب أن تكون عملية بشكل أساسي ومكثفة للغاية ومختلفة تمامًا عما هو مطلوب في العديد من المقررات الأخرى التي تستتند أكثر إلى المعرفة النظرية التي تتطلب القراءة والحفظ.

(٣) **قدرات الطلاب وسلوكهم The student's abilities and attitudes:** حيث يرى كلًا من (Alturki, 2016; Gomes & Mendes, 2014; Shih, 2007; Çakıroğlu, 2014) أن الطلاب لا يعرفون كيفية حل المشكلات: فالسبب الأكثر أهمية للصعوبات التي يشعر الكثير من المبتدئون بتعلم البرمجة فيها هو افتقارهم إلى مهارات عامة في حل المشكلات حيث لا يعرف الطلاب كيفية إنشاء خوارزميات، وذلك أساسًا لأنهم لا يعرفون كيفية حل المشكلات، ويتطلب حل المشكلات قدرات متعددة لا يملكها الطلاب غالبًا، وهي: (أ) فهم المشكلة: ففي كثير من الأحيان يحاول الطلاب حل مشكلة ما دون فهمها تمامًا ويحدث هذا في بعض الأحيان لأن الطالب يواجه صعوبات في تفسير بيان المشكلة والبعض الآخر لمجرد أن الطلاب حريصون على بدء كتابة التعليمات البرمجية وعدم قراءة وتفسير وصف المشكلة بشكل صحيح. (ب) العلاقة بين المعرفة: لا يقوم العديد من الطلاب بإنشاء مقارنات صحيحة مع المشكلات السابقة ولا ينقلون المعرفة السابقة إلى المشكلات الجديدة فهم يميلون إلى تجميع المشكلات التي لها نفس الخصائص الظاهرة بدلاً من نفس المبدأ، وبالتالي يقوم الطلاب في كثير من الأحيان ببناء حلولهم على مشاكل ليس لها صلة بالمطلوب مما يؤدي إلى حلول غير صحيحة.(ج) التفكير في المشكلة والحل: يميل الطلاب إلى كتابة إجابة

قبل التفكير بعناية في الحل، ففي كثير من الأحيان يتم الاختبار بشكل سريع يكون فيه الطالب راض لمجرد أن البرنامج يعمل مع مجموعة بيانات دون إجراء اختبارات أكثر شمولًا. (د)عدم الثبات: غالبًا ما يتخلى الطلاب عن حل مشكلة ما إذا لم يجدوا حلاً سريعًا، فعادة يتطلب حل مشاكل البرمجة الجهد والمثابرة ومع ذلك، عند مواجهة أي صعوبة يفضل العديد من الطلاب أن يطلبوا الحل من زميل لهم أو الاستسلام ، بدلاً من الاستمرار في محاولة حل مشكلة، وهذه النقطة مهمة جدا؛ لأن التعلم أكثر فعالية عندما يجد الطلاب الحل بأنفسهم بعد المثابرة منهم للوصول للحل، بدلاً من مجرد قراءة الحل بعد الحصول عليه جاهزًا من المعلم أو أي زميل.(هـ) العديد من الطلاب ليس لديهم ما يكفي من المعرفة الرياضية والمنطقية، فقد أجريت بعض التجارب لاستكشاف العلاقات بين كفاءات حل المشكلات الرياضية ونقص قدرات البرمجة التي أبدتها مجموعة من الطلاب. هذه التجربة نفذت خلال في الفصل الدراسي الثاني من عام ٢٠٠٥/٢٠٠٦ وخلص الباحثون إلى أن الطلاب المشاركين واجهوا صعوبات عميقة في العديد من المجالات مثل: حساب التفاضل والتكامل الأساسي ونظرية الأرقام أو المفاهيم الهندسية المثلثية البسيطة. (و) أيضا هناك صعوبات في تحويل مشكلة نصية إلى صيغة رياضية تحلها. كما تم تحديد القيود في مستوى التجريد والتفكير المنطقي. لذلك فإن المعرفة الرياضية مهمة جدًا للبرمجة وهناك العديد من الدراسات التي أكدت على دليل وجود علاقة بين مهارات البرمجة والخبرة في الرياضيات.(ي) يفتقر الطلاب إلى خبرة برمجية محددة: تنجم صعوبات البرمجة لدى العديد من الطلاب أيضًا عن أخطاء البرمجة والمفاهيم الخاطئة المحددة، في بعض الأحيان نجد طلابًا لا يعرفون كيف تعمل هياكل البرمجة الشائعة أو لديهم مفاهيم خاطئة عنهم، ومن الشائع أيضًا أن يظهر الطلاب صعوبات في اكتشاف أخطاء البرمجة اللغوية والمنطقية البسيطة.

(٤) **طبيعة البرمجة The nature of programming:** فيرى كلًا من (Alammary, Carbone, & Sheard, 2012; Gomes, & Mendes, 2014; Alario-Hoyos, et al., 2016; Topalli, & Cagiltay, 2018) أن البرمجة تتطلب مستوى عال من التجريد، حيث يتطلب تعلم البرمجة مهارات مثل التجريد والتعميم والنقل والتفكير النقدي وخاصة في مراحل التعلم الأولي حيث يتوقع من الطلاب فهم وتطبيق بعض مفاهيم البرمجة التجريدية مثل هياكل التحكم لحل المشكلات، فلغات البرمجة لها بناء جملة معقد للغاية حيث تم تطوير لغات البرمجة للاستخدام الاحترافي وليس لدعم التعلم، كما اللغات الشائعة واسعة المحتوى ولديها العديد من التفاصيل الغوية المعقدة لحفظها، لذلك يتطلب هذا التعقيد أن يضطر الطلاب إلى التركيز في وقت واحد في بناء الخوارزمية والقواعد اللغوية.

(٥) **الآثار النفسية Psychological Effects : (أ)** الطلاب ليس لديهم الدافع والرضا الكافي: فلا يوجد لدى الكثير من الطلاب الدافع الكافي لدراسة البرمجة، لأن هناك ارتباط سلبي للغاية بالبرمجة ينتقل من طالب لآخر. فهناك الصورة العامة للمبرمج باعتباره الطالب غير الملائم اجتماعيًا، بالإضافة إلى أن مقررات البرمجة تكتسب سمعة كونها صعبة، لذلك من الصعب تخيل الطلاب الذين يتطلعون إلى هذه الصورة. وإذا حاول الطلاب التسجيل في المقرر دائمًا يتوقعون أن يكون الأمر صعبًا، كما أن هناك صورة سلبية لأولئك الذين يتفوقون في دراستها، لذلك ينقص الطلاب التحفيزووجود الدافع والرضا. (ب) يتعلم الطلاب البرمجة في فترة صعبة من حياتهم، حيث يتم تدريس البرمجة عادةً كموضوع أساسي في بداية مقررات التعليم العالي بالتزامن مع فترة انتقالية وعدم استقرار في حياة الطالب؛ لأن هذا هو الوقت الذي فيه العديد من الصعوبات والتحديات لحياة جديدة ومستقلة، وهنا يمثل المقرر البرمجي تحديًا صعبًا، لذلك يفضل عدم وضعها في فترة انتقالية أو التخفيف من توصيف المقرر وتبسيطه (Gomes& Mendes, 2007&2014).

وقد تتبع الباحثون الدراسات التى استخدمت أشكال أو أنماط وتكنولوجيات فعالة يمكن أن تساعد في عرض المشكلات البرمجية بشكل يمكن أن يعالج صعوبات التعلم السابقة وتنمي تعلم البرمجة بكافة جوانبها بشكل يبدأ من تعلم الأساسيات التي تتمثل في فهم مفاهيم البرمجة والتمييز بينها والتعرف على تراكيبها ورموزها وقواعد كتابتها إلى فهم بنية العبارات والجمل والمشكلات كاملة، ثم تقييم ما تم تعلمه، وتم التوصل إلى الأنماط التالية:

١- نمط عرض الألغاز Puzzles:

يهدف التعلم من خلال الألغاز إلى تعليم الطلاب التفكير الناقد وتقنيات حل المشكلة، وتعد الألغاز بشكل مبسط يساعد على سهولة تذكر التعلم، والمشكلة هنا تعد تحديًا للمتعلم القائم على حلها، فهى تعد بحيث لا يكون الحل مرتب بشكل واضحًا، ويمكنه إعادة المحاولة أكثر من مرة للوصول للحل.

في مقررات برمجة الكمبيوتر، تتم عملية حل اللغز على عدة خطوات ، في البداية يتم طرح المشكلة تبعًا لمحتوى اللغة الذى يتم تدريسه، ثم يتم تقسيم حل المشكلة البرمجية إلى عدة قطع أو أجزاء، ويعتمد ذلك على مدى صعوبة المشكلة وتفرعها، ويتم تجميع قطع اللغز Puzzle التي تمثل أكواد الحل على خط واحد، ثم يقوم الطالب بمحاولة حل البرنامج من خلال اختيار قطعة الكود المناسبة، وبعد نجاحه يتم تقييمه، وإجراءات حل اللغز يقوم المعلم بإعلامها للطلاب أو من خلال نظام التعلم المستخدم بشكل آلى (Mohorovičić, & Strčić, 2011)

وقد أوضحت العديد من الدراسات أهمية استخدام الألغاز Puzzle في التدريب على حل المشكلات ومدى مناسبته لتدريس مقرر البرمجة التمهيدية، ولكن مع وجود العديد من الملاحظات والمشكلات التي تمثلت في كونه مناسب فقط للمشكلات الصغيرة، ولا يناسب كل الفئات العمرية، وكذلك حاجة الطلاب وتوصيتهم بمزيد من الأمثلة التى تدعم تعلمهم لفهم بينة وهيكل المشكلة البرمجية وذلك في بعض الدرسات التي استخدمته كأداة فقط للمراجعة قبل بدء التعلم الجديد، كما أوصوا بضرورة البحث فيه مستقبلًا بدمجه مع أليات أخرى تعوض حاجة الطلاب إلى فهم بنية المشكلة وتوفره بشكل دائم، إلى جانب البحث عن أنماط أخرى توفر مزيد من الأمثلة وطرق أخرى في التقييم (Oyelere et al., 2019; Ihantola & Karavirta, 2011; Kulikov & Pevzner, 2018; Parsons& Haden, 2006; Merrick, 2010; Sunday et al., 2020). هذا بالإضافة إلى جانب العديد من الدراسات التي أوصت نصًا بضرورة استخدامه في التدريب على مفاهيم وقواعد وبناء الجملة البرمجية كما ورد في (Parsons& Haden, 2006, Oyelere et al., 2019; Merrick, 2010; Sunday et al., 2020)

نمط عرض الإكمال Completion problem :

يعد هذا النمط إحدى الأنماط التي تتبع نهج السقالات التعليمية Scaffolding كما أشار رحمان Rahman (2012) ، ويتم فيها إعطاء مشكلة محلولة بالكامل ثم إتباعها بمشكلة أخرى تكون فيه إحدى خطواتها ناقصة، وعلى المتعلم إكمال هذه الخطوة الناقصة، ومن ثم إيجاد الحل المناسب، وفي المشكلة التالية يمكن إنقاص خطوتين ولحل المشكلة على المتعلم إكمال الخطوتين، وهكذا يستمر حتى يتم التأكيد على التعلم المطلوب.

يلعب نمط الإكمال دورًا إيجابيًا في عرض المشكلات البرمجية وتعلم الطلاب للبرمجة وفقًا للأدبيات والدراسات، مثل دراسات (Van Merriënboer, 1990; Paas, 1992; Van Merriënboer et al. , 2002; Griffin et al., 2008; Redford et al., 2012; Thiede et al., 2010; Koriat & Ackerman, 2010; Koriat, Ma'ayan, & Nussinson, 2006; Baars et al., 2013; Rahman, 2012) ووفقًا لهذه الدراسات فإن نمط الإكمال لا يتم استخدامه بمفرده بل لابد أن يسبقه شرح يدعم هذا النمط، ثم يتم التعلم أو التدريب من خلاله، وقد ترجع فعالية هذا النمط إلي مزاياه المعروضة في الدراسات السابقة لهذه لنقطة وهي فعالية مهام نمط الإكمال التي تؤدي إلى التنشيط، فالمهمة بطبيعتها تقوم على معالجة خطوات الحل المقدمة لأنها تحتوي على معلومات أساسية يحتاج إليها المتعلم قبل أن يتمكن من الاستمرار. بالإضافة إلى ذلك ، خطوات الحل المقدمة هي أمثلة على منهجية نظامية صحيحة لحل المشكلة، كما يمكن أن تكون أنماط الممارسة متعددة مراحل لتحفيز دوافع التعلم، و يمكن أن تقتصر الإجابة الصحيحة على مساحة صغيرة لتبسيط عملية التشخيص المعقدة لبرنامج التتبع.

نمط عرض المصفوفات Matrices:

تعد المصفوفة إحدى الأنماط التي تفرعت من التعلم بالأمثلة المحلولة و يتم استخدامها بغرض تقليل الحمل المعرفي الذى تسببه عرض المشكلة بالطريقة التقليدية، وأيضًا لتحسين التعلم، والمصفوفة Matrix هي عبارة عن " جدول مكون عمودين يسمح بالمقارنة بين فئة واحدة أو أكثر " (Kauffman & Kiewra 2010) ، كما تعرض الروابط الهرمية من خلال الأشكال، والكشف عن العلاقات والأنماط التي يسهل فهمها من قبل المتعلم (Robinson & Skinner 1996).

وقد استخدمت المصفوفات في عرض المشكلات والتعلم كاحدى الأنماط التي تقدمها استراتيجية الأمثلة العاملة Worked example، وذلك وفقًا للدراسات (Hancock-Niemic et al. , 2016; Atkinson et al., 1999; Atkinson, Merrill-Lusk, and Bietzel, 2007; Jairam et al., 2012; Kauffman, & Kiewra, 2010; Langan-Fox, Waycott, & Albert, 2000) ، وكان ضمن المشكلات التي واجهها الطلاب وذكرها الباحثون ان من عيوبها عدم قدرتها على عرض المفاهيم والقواعد البرمجية وكثرة الأخطاء الإملائية والبرمجية أثناء الكتابة، وضرورة تقديم الشرح قبل استخدامها.

ومن ثم كانت الحاجة إلي البحث عن استراتيجية تعالج ما سبق من مشكلات وذلك في بيئة التعلم المدمج، حيث يؤكد تشن (٢٠١٣) Chen, أن التعلم المدمج أصبح أحد الطرق التعليمية التي تساهم في إنجاح العملية التعليمية، وذلك من خلال إكتساب المعارف والمعلومات وجها لوجة، وإستخدام المنصات التعليمية، والتعلم الذاتي والتعاوني، وتحمل الطالب مسؤلية تعلمة من خلال الأنشطة التعليمية والأساليب المختلفة عن الطرق التقليدية. وتؤكد العديد من الدراسات على أهمية وإيجابيات بيئات التعلم المدمج في العملية التعليمية مثل: (Ja'ashan, 2015 ; Mirriahi , Alonzo & Fox , 2015) والتي من أهمها: زيادة فاعلية التعلم، تنويع طرق ووسائل المعرفة، تحقيق التعلم النشط. وفي الفقرات التالية يتم عرض مصادر الشعور بمشكلة البحث وصياغتها.

مشكلة البحث:

نبعت مشكلة البحث من عدة محاور هي:

أولًا: من خلال إطلاع الباحثون على الأبحاث والأدبيات المرتبطة بتعلم البرمجة، ومشكلاتها، وأنماط عرض المشكات البرمجية المختلفة، تبين التالي:

أدت نسب الفشل المرتفعة في مقررات البرمجة التمهيدية إلى مشكلة عالمية دفعت كثير من الباحثين إلى اقتراح منهجيات وأدوات لمساعدة الطلاب. وعلى الرغم من أنه تم الإبلاغ عن أن بعض هذه الأدوات لها تأثير إيجابي في تعلم الطلاب ، إلا أن المشكلة لا تزال دون حل. وذهب بعض الباحثين للإعتقاد بأن هناك عديد من الأسباب التي تسبب مشكلة التعلم هذه. ربما كان الأهم هو الافتقار إلى قدرات حل المشكلات التي يظهرها العديد من الطلاب. فهم لا يعرفون كيفية البرمجة ، وربما يرجع ذلك أساسًا إلى طرق عرض المشكلات وأنماط تقديمها للطلاب (Mendes ,Gomes, 2007,2014).

ومن أهم الأسباب في إيجاد عديد من المتعلمين صعوبة في تعلم البرمجة، طبيعة المهمة البرمجية المجردة، فالمفاهيم المجردة مثل المتغيرات وأنواع البيانات والذاكرة الديناميكية لا ترتبط ارتباطًا وثيقًا بالمواقف الحياتية مما يتسبب بصعوبة فهم المتعلمين لهذه المفاهيم (Dunican, 2002)، ووفقًا لبيركينز ومارتن كما ذكر جيلمور (1990) Gilmore أن الصعوبات الرئيسية ذات شقين: الشق الأول، وهو المعرفة الهشة ، والثاني، هو إهمال الاستراتيجيات، فالأول هو توفر المعرفة لدى الطالب، ولكنها لا تستخدم عند الحاجة إليها، وفي الشق الثاني تعتبر الاستراتيجيات غير كافية مما يؤدي إلى فشل المتعلمين أثناء محاولتهم لحل المشكلة، وذكر أيضًا أن الصعوبات تحدث في ثلاث نقاط مختلفة: المرحلة الأولى من تصميم الحل، وأثناء الترميز أو تكويد الحل، وأخيرًا في أخطاء التصحيح وقت التشغيل.

وكذلك قد أوضح كلًا من روزمانا و زمزوري (2019) Rosminah & Zamzuri في دراستهم عن صعوبات تعلم البرمجة من خلال توضيح رؤية الطلاب أن لديهم مشاكل في القراءة والتتبع والكتابة وتصميم بسيط جزء التعليمات البرمجية، هذا يتسبب في حصول الطلاب على موقف الاعتماد على الآخرين لإكمال المهمة، فالإتجاه الحالي في تعلم البرمجة هو التعلم التعاوني أو برمجة الأقران وبالتالي أصبح الطالب كسول وغير مهتم بالتعلم، وهذا من أهم الاسباب التي جعلت هذه الدراسة تعتمد وترتكز على التعلم الفردي للطلاب، ومن الدراسات التي أكدت نفس الصعوبات (McCracken et al.,2001; Lister et al., 2004; Tenenberg et al., 2005)

كما أوضحت هذه الدراسات والأدبيات (Rosminah & Zamzuri, 2019; Albrecht, Gumz & Grabowski ,2018; Linden & Lederman, 2011; Ismail, Ngah & Umar, 2010; Rudder, Bernard& Mohammed, 2007) أن النقاط الصعبة التي يواجهها الطلاب هي المفاهيم المتعلقة بالذاكرة والتي لم يتمكن الطلاب من استخدامها في خلق نموذج عقلي واضح للذاكرة، وتتفاقم المشكلة عندما يتم استخدام الوسائط الثابتة هي تدريس المفاهيم الديناميكية للبرمجة وبناء عليه تم تحديد أربعة مصادر رئيسية للمشكلة التي يواجهها الطلاب في تعلم البرمجة. هي: (١) عدم فهم وإتقان بناء الجملة والبرمجة، (٢) افتقار الطلاب إلى المهارة في تحليل بنية المشكلة البرمجية، (٣)) الاستخدام غير الفعال لتقنيات وأنماط عرض المشكلات (٤) الإستخدام غير الصحيح لاستراتيجية حل المشكلات في البرمجة.

كما أثبتت الأدبيات السابقة أن لغة البرمجة تشبه في تعلمها أي لغة من اللغات التي ينطق بها العالم فهي تتكون من مجموعة من المفردات والتراكيب والقواعد التي تحكمها، وأوصت الدراسات بضرورة البدء في تعليم مفردات اللغة ومفاهيمها أولًا والتدريب عليها بشكل ديناميكي يضمن ثبات المفاهيم وبناء الجملة، ثم ضرورة توضيح بناء المشكلة والتي تتكون من مجموعة من العبارات والأسطر التي تم التدريب على كل منها على حدا في المرحلة الأولى فالطالب المبتدئ مع عدم وجود خبرة في البرمجة يحتاج لتعلم بناء لغة البرمجة قبل تطبيقها مهارات في أسلوب الهيكلة والبناء والبرمجة ثم ضرورة تقييم ما تم تعلمه للتأكد من تحقق نتائج التعلم المطلوبة.

كما أوضحت هذه الدراسات ضرورة وجود أمثلة كثيرة يرجع اليها الطالب في وقت للتعلم وللتدريب. مع ضرورة وجود التغذية الراجعة الألية وضمان التقييم الذاتي الذي يعوض وجود المعلم أثناء التدريب في المنزل، الأمر الذى يضمن وجود الدعم الدائم وقت عدم توفر المعلم خارج الفصل الدراسي ورده بشكل دائم على الطالب (Albrecht, Gumz, & Grabowski, 2018).

وقد وجد لانغان فوكس وآخرون أن المبتدئين يربطون تعلمهم بكل ما هو سطحى كأساس لحل المشكلة، وربما يكون هذا هو السبب في عدم تعرفهم على بنية المشكلة الأساسية، وأيضًا أشار (كيليسي ومايرر، ١٩٩٦) وفقًا لدراسة لانغان فوكس أن المتعلمين وخاصة المبتدئين بحاجة إلى عرض بنية المشكلة بأكثر من طريقة أو مثال، أي عدم الثبات على نمط واحد في عرض البنية (بنية متغيرة). (Langan-Fox et al., 2000).

ثانيًا: خبرة الباحثين في تدريس مقررات البرمجة التمهيدية بشقيها النظري والعملي بقسم تكنولوجيا التعليم والمعلومات بكلية البنات جامعة عين شمس تبين من خلالها مدى حاجة الطلاب لإستراتيجية تدريس تضمن تعلم الطلاب لكل جوانب البرمجة ومراحل تعلمها بدءًا من قواعدها ومفرادتها إلى تعلم بنية المشكلة، وحاجة الطلاب لمزيد من الشرح والتدريب، والتنويع في طرق عرض المشكلة، والذي يرجع لطبيعة المشكلات واختلافها ومستوى صعوبتها، وتباين الفروق الفردية واختلافها بين الطلاب ، ومن هنا كانت الحاجة إلى التدريب على قواعد اللغة، وأنواع الأخطاء بشكل جديد مناسب لكل مرحلة من مراحل تعلم اللغة بشكل مركز كل على حدا، الأمر الذي لا تسمح به الطريقة التقليدية، وأيضًا حاجة الطلاب إلى حل الكثير من المشكلات للتمكن من مهارات الحل المطلوبة، والتي لا يسمح وقت التعلم النظامي والمباشر بالكلية بتنفيذها جميعًا.

ثالثًا: توصيات المؤتمرات، مثل :

- مؤتمر العلوم والمعلومات بلندن (٢٠١٦) Science and Information Conference والذى أشاد بأهمية البرمجة، وأوضح مدى صعوبتها لطلاب علوم الكمبيوتر، ولذلك أوصي باستخدام طرق جديدة في تعليم البرمجة، وتغيير طرق العرض المعتادة، وتطوير برمجيات جديدة من شأنها تحسين مستوى الطلاب، وتنمية مهارات البرمجة لديهم.
- المؤتمر الدولى لتكنولوجيات وتطبيقات التعليم الإليكتروني الناشئة (٢٠١٦) International conference on Emerging E-learning Technologies and Application والذى أوضح أهمية البرمجة القائمة على الممارسات، وحاجة الطلاب للحصول إلى تحديث مناهج البرمجة الحديثة،

وتقديمها بشـــكل يقدم المهام البرمجية بشـــكل مبســـط مألوف للمتعلمين، وأكدوا على اســـتخدام طرق التدريس من خلال اللعبة.

رابعًا: قلة الدراسات، وخاصة العربية التى تناولت تصميم استراتيجيات قائمة على تكنولوجيات تعالج مشكلات تعلم اللغة وحل المشكلات البرمجية، فمعظم الدراسات تكون لتقصي أثر استراتيجيات على توابع تربوية معينة، مثل: دراسة (إبراهيم أحمد السيد عطية، ٢٠١٠)، والتي هدفت إلى بحث أثر التفاعل بين استراتيجية حل المشكلات مفتوحة النهاية والسعة العقلية على الحلول الإبتكارية لمشكلات البرمجة التعليمية لدى طلاب الدبلوم المهنية، والتى توصلت نتائجها إلى فعالية كلاً من استراتيجية حل المشكلات مفتوحة النهاية، والسعة العقلية، على الحلول الإبتكارية، ودراسة (**علي محمد علي الزعبي**، ٢٠١٤)، والتى هدفت إلى تقصى أثر استراتيجية تدريسية قائمة على حل المشكلات في تنمية مهارات التفكير الإبداعى الرياضي لدى طلبة معلم الصف فـي جامعـــة اليرموك . وقد أظهرت النتائج تحسنًا فـي مهـارات التفكيـر الإبداعي الرياضي (الطلاقـة، والمرونـة، والأصـالة) لـدى طلبـة المجموعـة التجريبيـة، كما وجدت الدراسة فروقا دالة إحصائيا بـين المجمـوعتين التجريبيـة والضـابطة فـي مهـارات الطلاقـة والمرونـة والأصـالة وفـي الدرجـة الكليـة للاختبـار لصـالح المجموعة التجريبيـة، وأظهرت النتـائج تحسـنا فـي مسـتويات التفكيـر الإبداعي للمجموعـة التجريبية الضابطة ، مقارنة مع مستويات المجموعة الضابطة.

مما سبق أمكن الباحثون صياغة مشكلة البحث الحالي في العبارة التالية :

" توجد حاجة إلى تصميم استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة للغات الحديثة (C++)"

أسئلة البحث:

وإنطلاقاً من صياغة المشكلة ، **قام الباحثون بطرح السؤال الرئيسي** التالي:

" كيف يمكن تصميم استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة C++؟ "

ويتفرع من هذا السؤال الأسئلة الفرعية التالية:

١. ما أسس ومراحل استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج؟
٢. ما مراحل اســـتراتيجية تكامل أنماط عرض المشـــكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة للغات الحديثة؟

أهداف البحث:

يسعي البحث الحالي إلى التوصل للتالى:

١. أسس ومراحل استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج.
٢. اسـتراتيجية لتكامل أنماط عرض المشـكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة C++.

فروض البحث:

١. يمكن وضع أسس لتطوير استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة.
٢. يمكن وضع استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة للغات الحديثة وخاصة لغة C++.

منهج البحث:

يتبع البحث الحالى المنهج الوصفي التحليلي، في عرض البحوث والدراسات وتحليلها، لإستخلاص الأسس والمراحل الأساسية للإستراتيجية المقترحة وكفايات برمجة C++، وتطوير استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة C++، وخطة تطبيقها لتنمية كفايات البرمجة للغات الحديثة ببيئة التعلم المدمج.

أهمية البحث:

تكمن أهمية البحث الحالي في النقاط التالية :

١. تقديم استراتيجية جديدة لبيئة التعلم المدمج لتعلم البرمجة قائمة على تكامل أنماط عرض المشكلة.
٢. قد تفيد هذه الإستراتيجية في بناء استراتيجيات أخرى مشابهة لتعلم البرمجة في بيئات أخري وباستخدام أنماط مختلفة أو طرق أخرى لإحداث التكامل.
٣. توجيه أنظار الباحثين المهتمين بمجال تكنولوجيا التعليم لإجراء دراسات وبحوث تسعى لتقديم استراتيجيات جديدة لتعلم البرمجة قائمة على توظيف التكنولوجيات الحديثة وإنتاج الأنظمة الموجهة للتعلم.
٤. يتوقع من تطبيق الاستراتيجية ارتقاء مستوى الطالبات في مادة البرمجة، وتكوين ميول إيجابية نحو مقرر لغات برمجة حديثة (٢) بشكل خاص وتعلم البرمجة عمومًا.
٥. يتوقع من تطبيق الاستراتيجية تنمية مهارات حل المشكلة البرمجية لدي الطالبات بشكل كبير يؤثر إيجابياً علي مستوى تحصيلهم بمقرر لغات برمجة حديثة (٢).
٦. يمكن تطوير بيئة تعليمية مدمجة قائمة على استراتيجية تكامل أنماط عرض المشكلات البرمجية.
٧. يقدم البحث استراتيجية جديدة تثبت مدي فعالية وأهمية التعلم المدمج عند تعلم البرمجة.
٨. يمكن استخدام الاستراتيجية لخدمة مقررات برمجية أخري قائمة علي حل المشكلات مع مراعاة المحتوى وخصائص المتعلمين.

المشاركون في البحث Participants:

المشاركون في البحث عينة قصدية مكونة من عدد (٩) من المحكمين تخصص تكنولوجيا التعليم من أعضاء هيئة التدريس بأقسام تكنولوجيا التعليم بالجامعة (ثلاثة أساتذة، وأربعة أستاذ مساعد، واثنين مدرس).

خطوات البحث:

إتبع الباحثون الخطوات التالية:-

أولا: الجانب النظري: ويتمثل في إعداد الإطار النظري للبحث من خلال الإطلاع على الأدبيات والدراسات السابقة المتعلقة بتعلم البرمجة والتعلم المدمج وأنماط عرض المشكلات البرمجية، واستخلاص الأسس النظرية والعلمية للإستراتيجية.

ثانيا: الجانب التطبيقي الميداني: والذي يتمثل في:

١. التوصل الى المراحل الأساسية للإستراتيجية المقترحة لتكامل أنماط عرض المشكلات البرمجية.
٢. التوصل إلى الإجراءات الفرعية للإستراتيجية المقترحة لتكامل أنماط عرض المشكلات البرمجية.
٣. وضع مخطط أولى لمراحل الإستراتيجية لتنمية كفايات البرمجة للغات البرمجة الحديثة.
٤. العرض على مجموعة من المحكمين.
٥. إجراء التعديلات اللازمة في ضوء آراء المحكمين.
٦. التوصل الى المخطط النهائي لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج
٧. وضع مخطط لتطبيق الإستراتيجية لتنمية كفايات البرمجة للغات البرمجة الحديثة ببيئة التعلم المدمج.

حدود البحث:

اقتصر البحث الحالى على التالى

١. لغات البرمجة الحديثة علي البرمجة بلغة C++
٢. التوصل إلى الشكل النهائي لإستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج.
٣. الجانب المعرفي والآدائي لكفايات البرمجة بلغة C++.
٤. محكمين تخصص تكنولوجيا التعليم من أعضاء هيئة التدريس بأقسام تكنولوجيا التعليم بالجامعة.

مصطلحات البحث:

١. التعلم المدمج :Blended Strategy

- التعلم المدمج هو نظام متكامل يهدف الى مساعدة المتعلم خلال كل مرحلة من مراحل التعلم ويقوم على الدمج بين التعلم التقليدي والتعلم الالكتروني بأشكال مختلفة داخل قاعة الدرس(محمد عطية خميس، ٢٠٠٣، ص ٢٥٥).

- التعريف الإجرائي: توليف بين التعلم الإلكتروني والتعلم المباشر التقليدي في منظومة تعلم لتحقيق مخرجات التعلم المستهدفة بتنظيم أحداث التعلم والتدريب اللازمة لتلبية الاحتياجات التعليمية حيث يتم تعلم المفاهيم والقواعد البرمجية في بيئة التعلم المباشر، والتدريب على التعلم من خلال أنماط عرض المشكلات البرمجية في بيئة التعلم الإلكتروني وفق استراتيجية تكامل أنماط العرض المقترحة بشكل يضمن التفاعل النشط بين المعلم والطلاب وبين الطلاب وأنشطة التعلم وتوفير الدعم اللازم لإرشاد الطلاب أثناء التعلم، وكذلك الوقت الكافي للتعلم والتدريب داخل الصف الدراسي وخارجه.

٢. الاستراتيجية Strategy:

- الإستراتيجية: بمعناها العام هي خطة منظمة، تتكون من مجموعة محددة من الأنشطة والإجراءات، مرتبة في تسلسل معين، لتحقيق أهداف معينة في فترة زمنية محددة ". (محمد عطية خميس، ٢٠٠٣، ص ١٠٩).

- **التعريف الإجرائي:** خطة قائمة على نهج منظومي علمي تتكون من مجموعة من المراحل وإجراءاتها يتبعها المتعلم في بيئة التعلم المدمج حيث يتم تعلم المفاهيم والقواعد البرمجية في بيئة التعلم المباشر، والتدريب على التعلم من خلال أنماط عرض المشكلات البرمجية في بيئة التعلم الإلكتروني وفق استراتيجية تكامل أنماط العرض المقترحة للوصول إلى الهدف المطلوب وهو التمكن من كفايات البرمجية للغة C++.

٣. المشكلة البرمجية Programming Problem:

- **التعريف الإجرائي:** المشكلة البرمجية هي عبارة عن مهمة تمثل تحدى عقلي للمبرمج في كتابة برنامج مكون من مجموعة من الأوامر وفق لغة معينة، وعند تنفيذ هذه الأوامر بنجاح تتم المهمة، وتحل المشكلة البرمجية وتستخدم في هذا البحث ضمن الإستراتيجية لتنمية كفايات البرمجة.

٤. أنماط عرض المشكلة Problem view styles:

- **التعريف الإجرائي:** بقصد بأنماط عرض المشكلة البرمجية الطريقة أو الإسلوب الذي يتم به التعلم، أي طريقة وشكل سرد المشكلة والحل بشكل ما يؤدي إلى التعلم المطلوب.

٥. نمط عرض الألغاز Puzzles:

- الألغاز Puzzles هي عبارة عن طريقة لتعليم الطلاب التفكير الناقد وتقنيات حل المشكلة، يتم إعدادها بشكل مبسط يساعد على سهولة تذكر التعلم، واللغز أو المشكلة هنا تعد تحديًا للمتعلم القائم على حلها، فهى تعد بحيث لا يكون الحل مرتب بشكل واضحًا، ويمكنه إعادة المحاولة أكثر من مرة للوصول للحل (Mohorovičić & Strčić, 2011).

- **التعريف الإجرائي:** نمط للتعلم يختلف تصميمه وتكنولوجياته ومحتواه باختلاف المحتوى والفكرة والتعلم الذي يقدمه، وهو في هذه الدراسة عبارة عن نمط ديناميكي يستخدم في تعلم المفاهيم البرمجية من خلال السحب والإلقاء في مربعات مفرغة لأماكن الأكواد والرموز البرمجية.

٦. نمط عرض الإكمال Completion problem :

- **نمط عرض الإكمال** للمشكلة البرمجية هو عبارة عن إحدى أنواع السقالات التعليمية Scaffolding، ويتم فيها إعطاء مشكلة برمجية محلولة بالكامل للمتعلم ثم إتباعها بمشكلة أخرى تكون فيه إحدى خطواتها ناقصة، وعلى المتعلم إكمال هذه الخطوة الناقصة، ومن ثم إيجاد الحل المناسب، وفي المشكلة التالية يمكن إنقاص خطوتين ولحل المشكلةعلى المتعلم إكمال الخطوتين، وهكذا يستمر حتى يتم التأكيد على التعلم المطلوب (Rahman, 2012).

- **التعريف الإجرائي:** نمط لعرض وتقييم المشـــكلة البرمجية، يتم فيه تقديم مشـــكلة برمجية محلولة ينقصـها عدة خطوات يقوم الطالب بكتابتها وحلها في المحرر الخاص مكان كل خطوة ناقصـة، ثم تقديم الحل كاملًا كتغذية راجعة بعد الإنتهاء من الحل وإكمال الجمل البرمجية الناقصة.

٧. نمط عرض المصفوفات Matrices:

- المصـــفوفة Matrix هي عبارة عن جدول مكون عمودين يســـمح بالمقارنة بين فئة واحدة أو أكثر (Kauffman & Kiewra, 2010) .كما تعرض الروابط الهرمية من خلال الأشكال، والكشف عن العلاقات والأنماط التي يسهل فهمها من قبل المتعلم (Robinson & Skinner, 1996).

- **التعريف الإجرائي:** نمط لعرض تعلم بناء المشـــكلة البرمجية وهوعبارة عن جدول مكون عمودين، الأول به مشـــكلة محلولة والأخر به مشـــكلة مكافئة بها ســـطر برمجى ناقص، يكمله المتعلم بإختيار الإجابة الصـــحيحة من بين عدة إجابات أسـفل المصـفوفة، ثم عرض الحل الصـــحيح كتغذية راجعة بعد الإنتهاء من الحل.

٨. الكفايات البرمجية :

- **الكفاية:** هي قدرة الفرد على فهم المعرفة وإكمال مهمة محددة، كما يمكن تعريفها بأنها قوة الحافز لدى الشخص على فهم المعرفة وتطبيقها بنجاح. (Ilahi, Belcadhi & Braham 2013).
- **التعريف الإجرائي:** الكفـايـات البرمجيـة هي قـدرة المتعلم - معرفيـا وأدائيـا - على فهم وتطبيق المعرفة البرمجية ومهارات اللغة المسـتخدمة في حل المشـكلة البرمجية التي تعد بمثابة إكمال مهمة التعلم والوصول للهدف منه.

الإطار النظري للبحث

يتناول الإطار النظري لهذا البحث عدة محاور رئيسية، تبدأ بالمحور الأول، وهو بعنوان بيئة التعلم المدمج وتصميمها لإستراتيجية تكامل أنماط عرض المشكلة البرمجية، والمحور الثاني بعنوان تعلم البرمجة وكفايات البرمجة لحل المشكلات البرمجية للغات الحديثة، ثم المحور الثالث بعنوان أسس تطوير الاستراتيجية القائمة على تكامل أنماط عرض المشكلات البرمجية لتنمية كفايات البرمجة (C++) في بيئة التعلم المدمج (الإجابة عن السؤال الأول).

محاور الإطار النظري

المحور الأول: بيئة التعلم المدمج وتصميمها لإستراتيجية تكامل أنماط عرض المشكلة البرمجية:

يتكون التعلم المدمج **Blended Learning** من مجموعة واسعة من أساليب التعلم ، التي تجمع بين التعلم الإلكتروني والتعلم الحقيقي المباشر، والتقنيات ، والموارد ، ويوفر الفوائد الجوهرية التالية (Atef & Medhat 2015; Lin et al., 2016): **(١)** إنه يسمح بالتدريس والتعلم والتقييم للاستمرار حتى عندما الجامعة مغلقة. (٢) يعزز التعلم النشط لدى الطلاب لأن الطلاب يمكنهم توصيل احتياجاتهم إلى معلميهم وأقرانهم.**(٣)** يمكن أن تخفف من التأثير السلبي لمقررات الإنترنت المصممة بشكل سيئ مع المقررات عالية الجودة التي يقودها المدربون.**(٤)** يمكن أن يحسن من جودة تجربة التعلم بين الطلاب والمعلمين.**(٥)**

يوفر فرصة لدعم التشغيل في سياق عالمي بمزيد من الكفاءة.(٦) تعزيز المشاركة الطلابية والإلهام والتعلم الذاتي التنظيم مع التزويد المتأصل للتركيز على الطالب وتحسين الإنجازات الأكاديمية.

إن جميع الطرق والأساليب والاستراتيجيات التعليمية الحديثة التي تعتمد على الإنترنت والتعلم الإلكتروني تكفي لمهمات تعلم ولكن هناك مهمات تعلم تتطلب طرق التعليم التقليدي، تلك التي تحتاج إلي مختبرات حقيقية أو مصانع أو مزارع أو ورش ميكانيكية وما في شاكلتها، حيث لابد من أن يكمل كل منهما الآخر بحيث تحقق شروط التعلم لتلك المهام والمشاركة الأدائية للطلاب للتعلم وجعل البيئة التعليمية أكثر جاذبية، ولا نترك أيضا التعليم التقليدي بكل فوائده، ويتطلب ذلك استراتيجيات جديدة للتعليم المدمج، وتهدف هذه الاستراتيجيات إلى جعل العملية التعليمية ذات معنى؛ لأنها تربط الواقع الحقيقي الذي يعيشه الطلاب بالمواقف التعليمية المختلفة. (جمال مصطفي عبدالرحمن الشرقاوي، ٢٠١٢).

ومن استراتيجيات التعليم المدمج:

أولًا: استراتيجية التعليم المدمج كما حددها كل من محمد عطية خميس (٢٠٠٣)، وحسن عبد الله النجار (٢٠٠٩)، وهي تقوم على:

١. إثارة الدافعية في أنشطة ما قبل العرض، وتبصير المتعلمين بالأهداف المرجوة.
٢. تقديم التعلم في تتابع وصياغة المحتوى وعرضه وفق استراتيجية تناسب المهام والطلاب.
٣. تفاعل المتعلمين من خلال التدريبات والتطبيقات والأنشطة الموزعة والتوجيه، وتقرير تعلمهم.
٤. تقويم الأداء من خلال اختيار السلوك من خلال أدوات القياس.
٥. التطبيق على المتعلمين يكون من خلال الواجبات والممارسات التعليمية .

ثانيًا: استراتيجية التعليم المدمج كما حددها حسن زيتون (٢٠٠٥):

ويستخدم التعلم المدمج في التعليم والتعلم وفقًا لاستراتيجية معينة من جملة استراتيجيات لخصها حسن زيتون (٢٠٠٥) في الاستراتيجيات التالية:

الاستراتيجية الأولى: تتأسس على أن يتعلم درس أو أكثر بأسلوب التعلم الصفي، ويتعلم درس آخر أو أكثر بأحد أشكال التعلم الإلكتروني، ويقوم تعلم الطلاب بأي من وسائل التقويم التقليدية أو الإلكترونية.

الاستراتيجية الثانية: تتأسس على أن يتشارك فيها التعلم الصفي والتعلم الإلكتروني تبادليا في تعليم و تعلم درس واحد، غير أن بداية التعليم والتعلم تتم بأسلوب التعلم الصفي ويليه التعلم الإلكتروني، ويقوم تعلم الطلاب ختاميا بأي من وسائل التقويم التقليدية أو الإلكترونية، وهي الإستراتيجية المتبعة في هذا البحث.

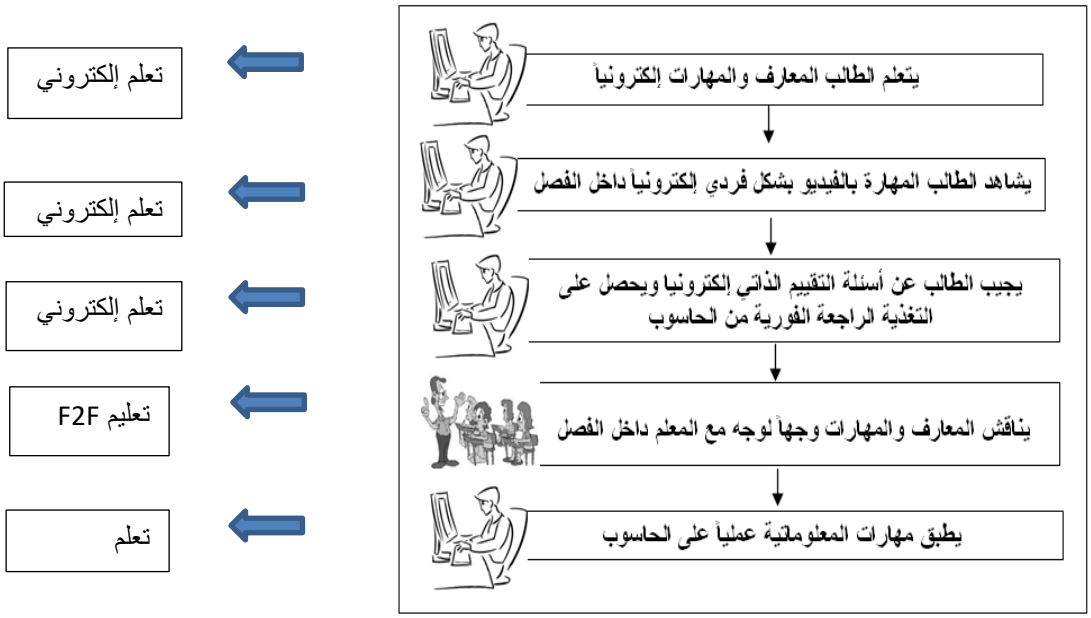
الاستراتيجية الثالثة: تتأسس على أن يتشارك فيها التعلم الصفي والتعلم الإلكتروني تبادليا في تعليم و تعلم درس واحد، غير أن بداية التعليم والتعلم تتم بأسلوب التعلم الإلكتروني، ويعقبه التعلم الصفي، ويقوم تعلم الطلاب ختاميا بأي من وسائل التقويم التقليدية أو الإلكترونية.

الاستراتيجية الرابعة: تتأسس على أن يتشارك فيها التعلم الصفي والتعلم الإلكتروني تبادلا في تعليم و تعلم درس واحد، بحيث يتم التناوب بين أسلوب التعلم الإلكتروني والتعلم الصفي أكثر من مرة للدرس الواحد ويقوم تعلم الطلاب ختاميا بأي من وسائل التقويم التقليدية أو الإلكترونية.

كما يمكن للمعلمين أن يلعبوا دورًا فعالًا في تقرير الكيفية والدرجة التي يمكن بها دمج وتكامل الكمبيوتر والتكنولوجيا مع المناهج التقليدية ونظرًا لأن المشكلات البرمجية تتطلب وقتًا أطول لحلها، وهذا لا يتاح مع بيئة التعليم التقليدية. حيث يجب الإنتباه إلى توفير فرص لحل مشكلات يعطى لها الوقت الكافي في الصف (Tucker, 2012) ولذلك تم استخدام التعلم المدمج الذي يدمج بين التعليم التقليدي والتعليم الإلكتروني، فبيئة التعلم المدمج بيئة غنية بأدواتها التي يمكن استخدامها كوسيلة للتعلم في مراحل الإستراتيجية المقترحة.

ثالثًا: التعلم المدمج التتابعي Sequential Blended Learning Design للإستراتيجية :

كذلك اعتمد الباحثون على تصميم التعلم المدمج التتابعي Sequential Blended Learning Design في هذا البحث (Alraghaib, Elgazzar, & Nouby, 2015; Ismaeil, 2009)، لبساطة التصميم ومناسبته وإكماله لفكرة البحث الحالي. فهو يعد ملخصًا زمنيًا لموارد التعلم الإلكتروني وموارد التعلم المباشر التقليدي بشكل يؤدي إلى التتابع والترابط بين مصادر المعلومات المقدمة للمتعلم في البيئات التعليمية المختلفة؛ اعتمادًا على الإنسياب الخطي للمعلومات من مورد واحد إلى المورد اللاحق في المخطط الزمني للتصميم. يجمع التعلم المدمج بين مصادر المعرفة المتعددة للمتعلم ويعتمد على: خصائص الطالب ، طبيعة عملية التعلم ، ومصادر التعلم المتاحة ، والبنية التحتية لتكنولوجيا المعلومات في المؤسسة. لذا ، فإن العنصر الرئيسي في تصميم التعلم المدمج التتابعي هو جمع مصادر المعرفة المنفصلة المقدمة في المخطط الزمني / التدفق الخطي للمعلومات إلى المتعلم. استنادًا إلى وجهة النظر هذه حول التعلم المدمج التتابعي، تم تصميم إستراتيجية جديدة للتعلم المدمج التتابعي من قبل المؤلفين، وهو التلخيص الخطي اذ لابد أن يوضح خطوات ومصادر تقديم المعرفة والتدريب على المهارات العملية ثم التقييم وتوفير التغذية الراجعة وذلك بشكل تسلسلي منطقي (Alraghaib et al. , 2015) كما في شكل (١).



شكل (١) تصميم التعلم المدمج التتابعي بعد (Alraghaib, Elgazzar, & Nouby, 2015, p. 34)

ثانيًا: المحور الثاني تعلم البرمجة وكفايات البرمجة لحل المشكلات البرمجية للغات الحديثة:

أولًا: تعلم البرمجة للغات الحديثة:

تعتبر البرمجة عملية معقدة للغاية ومتعددة المراحل ، حيث تتطلب كل عملية فرعية معرفة محتوى مختلف ، واستدعاء عمليات معرفية أخرى (Ambrosio et al., 2011) والعديد من المهارات التي يجب معالجتها في تدريس البرمجة (Howard, 2002; Lehman, Bruning, & Horn, 1983). على سبيل المثال ، فهم القراءة ، التفكير المنطقي ، التفكير المنهجي ، اكتساب المكونات المعرفية في تحديد المشكلة ، تخطيط وإنتاج الحلول ، الإبداع ، الفضول الفكري ، المهارات الرياضية ، مهارات التفكير الظرفية ، التفكير الإجرائي ، التفكير المؤقت ، التفكير التحليلي والكمي ، استخدام مختلف المصادر ، والتدريب على تلك المهارات أو بعضها هو ما يجعل البرمجة والتعامل مع المشكلات البرمجية عملية مبدعة ومرنة في إنتاج الحلول (Ambrosio et al., 2011; Lau & Yuen, 2011).

أسس تعلم جوانب كفايات البرمجة:

تعتمد كفاءات البرمجة على المعرفة والمهارات؛ تتكون المعرفة من تعريفات وحقائق وبنية اللغة (أي بناء الجملة) وخوارزميات محددة من المعرفة البرمجية (Caspersen, 2007). ، وقد أطلق أوليفر (Oliver, 1993) بنية هرمية على معرفة البرمجة بناء على تجربته، حيث قال أن هناك ثلاثة أنواع أساسية من المعرفة البرمجية:

١. فهم بناء الجملة أو بنية البرنامج.

٢. القدرة على تطبيق عبارات (وهذا يعني الجمع بين البيانات وتنظيمها في الخوارزمية).

٣. القدرة على حل المشكلة.

بالنسبة للنوع الأول من المعرفة استنتج أوليفر أن الطلاب يجب أن يكونوا قادرين على تتبع الخوارزميات وفهم أي عبارات في البرامج ، بدلاً من فهم محتوى العمليات. أما فيما يتعلق بالنوع الثاني من المعرفة ، يعتقد أوليفر أن المبتدئين يجب أن يعرفوا معنى العمليات. عند مواجهة مشكلة في حل مماثل لبرنامج نموذجي، يجب على الطلاب استخدام البرنامج النموذجي لحل هذه المشكلة. بالنسبة للنوع الثالث من المعرفة ، نظر أوليفر للمعرفة المحدودة للمبتدئين. لذلك ، لم يؤكد الطلاب على كتابة برنامج بشكل مستقل، بل طلب بدلاً من ذلك أن يعيدوا صياغة مطالب حل المسألة. لإثبات البنية الهرمية لمعرفة البرمجة ، وقد صمم أوليفر اختبارًا يسمى اختبار العمليات الحسابية (APT). تنقسم العملية في APT إلى ثلاث خطوات لفهم ثلاثة أنواع من معرفة الطلاب بالبرمجة في كل خطوة.

وتتكون مهارات البرمجة من بعض الإجراءات المطلوبة التي توضحها الاستراتيجيات في تطبيق هذه المعرفة (Caspersen, 2007). ومن خصائص مهارات البرمجة كما يرى (حسن زيتون، ١٩٩٩) وفقًا لما ورد في (أحمد أبو العلا بهنساوى، وأخرون، ٢٠١٢) :

- المهارة تعبر عن القدرة على أداء عمل Action أو عملية Process، وهذا العمل أو العملية يتكون في الغالب من مجموعة من الآداءات أو العمليات الأصغر، وهي الآداءات أو العمليات Operation البسيطة الفرعية، أو المهارات البسيطة Sub-skills أو الإستجابات البسيطة Simple Responses، التي تتم بشكل متسلسل ومتناسق فتبدو مؤتلفة مع بعضها البعض.

- تتكون المهارة عادة من خليط من الاستجابات العقلية (وهي التي يغلب عليها الأداء العقلي مثل مهارات حل المشكلات، ومهارات التفكير الإبتكاري) والإجتماعية (وهي التي يغلب عليها الأداء الإجتماعي ومن أبرزها المهارات الإجتماعية المنزلية والمدرسية).
- يتأسس الأداء المهاري على المعرفة Knowledge أو المعلومات، إذ تكون المعرفة أو المعلومات جزءًا لا غنى عنه من هذا الأداء.
- ينمي الأداء المهاري للفرد ويحسن من خلال عملية التدريب Training أو الممارسة Practice، ويعتبر التدريب على أداء المهارة شرطا أساسيًا لتعلمها، ويجدر التنويه على أن التدريب ليس مجرد تكرار عشوائي للأداء، بل هو تكرار واع وهادف مصحوب بالتعزيز، أي أن التدريب يكون هنا نوعًا من الممارسة المعززة والموجهة لغرض معين والذي يؤدي لتحسن الأداء.
- يتم تقييم الأداء المهاري عادة بكل من معياري الدقة في القيام به والسرعة في الإنجاز معًا.

خطوات تعلم البرمجة:

كما اقترح موفت وديميل (1982) Deimel & Moffat أربع خطوات لتعليم البرمجة:

١. دع الطلاب ينفذون البرنامج من أجل مراقبة تنفيذها وتقييم فوائدها وقيودها.

٢. اطلب من الطلاب أن يدرسوا برامج جيدة التنظيم.

٣. يشارك المتنافسون في تعديل البرنامج وتمديده.

٤. دع الطلاب يقومون بتصميم وكتابة برنامج جديد بشكل مستقل.

و ذكر أحمد أبو العلا بهنساوى وأخرون (٢٠١٢) أن عملية إتقان مهارات البرمجة تتطلب مرور الطلاب بعدد من المراحل ، يمكن إجمالها فيما يلي:

- يتم تزويد الطالب بالمهارات المعرفية عن كيفية أداء تلك المهارة كأن يقدم له شـــرح نظري لخطوات أداء مهارة من مهارات البرمجة.
- يقدم للطالب نموذجًا عمليًا يوضـــح التنفيذ الفعلي لأداء المهارة، كأنه يشـــاهد مثال لخطوات حل مشكلة برمجية
- يقوم الطالب بتنفيذ الخطوات الســـابقة واللازمة لحل المشـــكلة البرمجية بهدف تحســـين أداء تلك المهارة، وحتى يصل لدرجة التمكن في أداء تلك المهارة.
- تقويم أداء الطالب على تنفيذ مهارة حل المشكلة البرمجية.

مراحل تعلم البرمجة في بيئة التعلم المدمج:

اقترح ســـعيد هاجرويت (2008) Hadjerrouit ثلاثة مراحل لتعلم البرمجة في بيئة التعلم المدمج، وكانت هذه المراحل كالتالي:

١. مرحلة تعلم المفاهيم:

حيث تؤكد نظرية التعلم المعرفي على مخطط المتعلم كهيكل معرفي منظم (Bruner, 1990; Gagne et al., 1993). على عكس المدرسة السلوكية، تعترف المدرسة المعرفية أن العقل البشري ليس مجرد متلق سلبي للمعرفة. بدلًا من ذلك ، فإن المتعلم يفسر المعرفة ويعطي معنى لذلك. ثم دمج المعرفة الجديدة مع المعرفة السابقة. يشير المنظور المعرفي للتعلم إلى النشاط العقلي، مثل التفكير التحليلي والتفكير النقدي. فعندما يطبق المعلمون منهجًا معرفيًا ، فهم يركزون على فهم المفاهيم وعلاقاتهم. وإذا كان المتعلمون قادرين على فهم الروابط بين المفاهيم ، وكسر المعلومات وإعادة بنائها من خلال الاتصالات المنطقية ، فإن فهمهم سيزداد.

٢. مرحلة البناء:

وهنا ترى نظرية التعلم البنائية المعرفة ككيان مبني مصنوع من قبل كل متعلم من خلال عملية التعلم. والهيكل التنظيمي للبناء يتم تعلمه كمنتج للانتقال من التلقي السلبي إلى عملية البناء النشط حيث يقوم المتعلمون ببناء معرفتهم الخاصة بناءً على المعرفة والخبرة السابقة (Duffy et al., 1993; Piaget, 1971; Steffe & Gale, 1995) ويتطلب التعلم البنائي من المتعلمين إظهار مهاراتهم من خلال بناء معرفتهم الخاصة عند حل مشكلات العالم الحقيقي. لذلك ، يدعو النموذج البنائي إلى تعليم يركز على المتعلمين ، لأن المفترض أن يتعلم الطلاب بشكل أفضل عندما يجبرون على استكشاف واكتشاف الأشياء بأنفسهم.

٣. مرحلة التقييم القائم على الحوار:

وفيها يمكن النظر إلى نظرية التعلم الإجتماعية على أنها تصحيح للبنائية ، حيث يتم فصل التعلم في البنائية عن السياق الاجتماعي. ففي حين أنه من المفترض أن يحدث التعلم البنائي بشكل فردي حيث يتفاعل المتعلم الفردي مع مادة الدراسة ، فإن المنظور الاجتماعي يعتبر التعلم كموقف اجتماعي والمعرفة موزعة اجتماعيًا (Vygotsky, 1978; Wengler, 1998). وفيها يحدث التعلم أثناء ممارسة المتعلمين واختبارهم وتحسين معرفتهم وتقييمها من خلال المناقشة والحوار والتعاون ومشاركة المعلومات والتفاعل مع الآخرين. وقد أكد فيجوتسكي Vygotsky (1978) بأن الطريقة التي يقوم بها المتعلمون في بناء المعرفة والتفكير والعقل والتأثير على شكل فريد تتم من خلال علاقاتهم مع الآخرين. وقال إن التوجيهات المقدمة من قبل أشخاص أكثر قدرة ، تسمح للمتعلم بالانخراط في مستويات النشاط التي لا يمكن إدارتها بمفردها.

ويمكن للباحثون التعقيب على المرحلة الأخيرة حيث يمكن أن يتم التعلم والتقييم بشكل فردى، فمن أنواع الحوار وفقًا للأدبيات حوار الطالب التفاعلي مع مصادر التعلم كحواره مه جهازه الشخصي والموقع الإلكتروني بشكل مفرد، وليس بالضرورة اجتماعه بالأشخاص فقط، حيث يجتمع الطالب بجهازه ويتم بينهم نوع من التفاعل والتواصل والتعزيز الآلى، أي كل طالب يتم تقييمه على حدا من خلال حواره مع جهازه الشخصي، والتغذية الراجعة الآلية المدمجة في تصميم كل نمط من أنماط عرض المشكلات البرمجية.

مستويات التدريب على المشكلة البرمجية :

في إطار كل ما سبق مناقشته من الأدبيات توصل الباحثون إلى أن التدريب على حل المشكلة البرمجية يشتمل على ثلاثة مستويات من التدريب، كالتالي:

١. التدريب على بنية الجمل البرمجية (ضمن التدريب على المفاهيم).
٢. التدريب على بنية المشكلة ، وتركيبها (البناء من خلال المهام والأنشطة في شكل مشكلات).
٣. التدريب على اكتشاف وتصحيح أخطاء المشكلة (التقييم).

ثانياً: حل المشكلات البرمجية وأنماط عرض المشكلات البرمجية للغات البرمجة الحديثة:

مفهوم المشكلة البرمجية :

يعرف كلًا من ديفيد وجيف (David Yong & Jeff Kint) ترجمة (خالد العامري، ٢٠٠١) كما ورد في (أحمد أبو العلا بهنساوى ، وأخرون، ٢٠١٢) المشكلة البرمجية بأنها معوقات وصعوبات تعترض طريقة الفرد أثناء محاولته الوصول إلى حل البرنامج.

وقد عرفها الباحثون أحمد أبو العلا بهنساوى وأخرون (٢٠١٢) بأنها الصعوبات والأخطاء الشائعة والمتكررة والمشكلات التي تواجه الطلاب أثناء قيامهم بإعداد البرنامج أو المشروعات البرمجية باستخدام أى لغة من لغات البرمجة.

ويعرفها الباحثون إجرائيًا بأنها موقف يمثل عقبة في طريق المتعلم لإنشاء برنامج ما باستخدام إحدى لغات البرمجة.

أنماط عرض المشكلات البرمجية:

قامت العديد من الأبحاث لمعرفة: أ) لما يعد تعلم البرمجة تحديًا ؟، ب) كيف يمكن تدريس مهارات البرمجة بفعالية في تطوير البرمجيات؟، وقد أثبت كلًا من: (Ford & Venema,2010; Thomas et al., 2002; Bornat, Dehnadi and Simon, 2008) أن معظم الخريجين لا يستطيعون كتابة كود جيد، وذلك يرجع إلى عدم فهمهم الجيد لمفاهيم البرمجة، ومن هنا قامت الأبحاث بعرض المشكلة البرمجية بأشكال وأنماط مختلفة، كان أهمها:

نمط الألغاز Puzzles في عرض المشكلات البرمجية:

غالبًا ما يبرر المتخصصون في تعليم اللغات استخدام الألعاب في تشكيل الحافز الذي يمكنهم توفيره للطلاب؛ فالألعاب يمكن وصفها وتعريفها فقط بالمتعة، ويتفق الجميع تقريباً على أنه إذا تم التعلم من خلالها فسوف يكون ممتعاً، و الطلاب سيتعلمون أكثر (Alemi, 2010).

فمثلاً إن استخدام ألعاب الكلمات لتعليم المفردات لا يعني تقديم طرقًا ممتعة لتمرير وقت الفصل الدراسي، فترفية الطلاب ليس مسؤولية المعلم (Allen 1983)، ولكن يتحمل المعلمون مسؤولية خلق جو فصلي بناء، مما يشجع على توسيع المفردات، واللعبة المختارة بشكل جيد يمكن أن تساعد الطلاب على اكتساب الكلمات والمفردات بشكل صحيح والإحساس بأن بعض الكلمات مهمة وضرورية لأنه بدون هذه الكلمات، الهدف من اللعبة لا يمكن تحقيقه (Allen 1983).

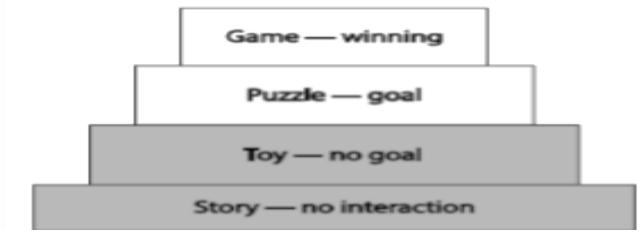
التعلم القائم على الألغاز يتيح لنا تعلم مهارات حل المشكلات بالطرق المختلفة، حيث نتعلم من خلال التجربة (فلا يمكن تعلم مهارات حل المشكلات إلا من خلال المرور بمشكلة) ومنها نتعلم عن طريق التقليد ، لأنه من المفيد تقليد (تطبيق) بعض المبادئ والتقنيات، وقبل كل شيء يتم التعلم من خلال التفكير في التعلم القائم على الألغاز الذي يشجع على التفكير في:

- ما الذي نتعلمه؟
- كيف نتعلمه؟
- كيف نستخدم ما تعلمناه؟

كما تعتبر الألعاب ذات أهمية خاصة للطلاب لتطوير مهارات التعلم التي تسمح لهم بسهولة إنشاء تجريدات مفاهيم أو خوارزميات (Levitin, 2005)، وغالبًا ما تقدم الأنظمة الشبيهة بالألعاب شكلاً من أشكال المشكلة وتتطلب من المستخدمين صياغة حل لحل هذه المشكلة (Dong et al., 2012) في سياق البرمجة ، حيث توجد العديد من أنظمة الألغاز (الأحاجي) التي تعطي المستخدمين هدفًا يتطلب منهم برمجة مسار الكائن من خلال العقبات المتواجدة داخل شبكة ما (Sweller et al., 2011)، كما أنه في كثير من الأحيان تستخدم بيئات برمجة تشبه الألغاز برامج تعليمية كجزء من تجربة المستخدم الأولية (Lee et al., 2014) ويقدم البرنامج التعليمي في شكل ألغاز تتواجد في عدة مستويات من السهل إلى الصعب لتشجيع التعلم ، كما تستخدم بعض الأنظمة الألغازللتدريب على تصحيح الأخطاء ..(Lee et al., 2014) أو المنافسة (Hartness, 2004; Long, 2007).

من خلال دمج الألغاز في الألعاب ، يمكنك إنشاء خيارات مثيرة للاهتمام. وفقا لمصمم اللغز سكوت كيم ، هناك نوعان من السمات الرئيسية التي تميز اللغز: السمة الأولى هي أن الألغاز ممتعة، وهذا يعني أن الألغاز هي شكل من أشكال اللعب، السمة الثانية المميزة هي أن اللغز لديه دائمًا إجابة صحيحة، وهذا يميز الألغاز عن الأشكال الأخرى للعب ، مثل الألعاب أو الدمى (Fullerton, 2008).

الشيء الوحيد الذي تشترك فيه معظم الألغاز هو أنها تتوقف عن أن تكون مرحة بمجرد حلها ، مما يعني أنها ممتعة فقط في المرة الأولى التي تحلها فيها. يختلف هذا النمط من الألعاب عن معظم الألعاب حيث يوجد عادةً نوعًا ما من العناصر الديناميكية في الألعاب التي تمنحهم قيمة إعادة التشغيل وتقدم تحديًا جديدًا، ويمكن تحقيق ذلك من خلال وجود خصم ذكي مثل لعبة الشطرنج ، أو من خلال القدرة على توليد تحديات جديدة للاعب، مما يمنح اللاعب هدفًا دائم التقدم من خلال قائمة النتائج العالية (Schell, 2008). مصمم اللعبة والمؤلف كريس كروفورد يميز أربعة أنواع مختلفة من اللعب ، كل واحدة مبنية فوق الأخرى، وهي: القصص والألعاب والألغاز والدمى، غالبًا ما يكون للألعاب هدف الفوز وتختلف عن الألغاز بمعنى أن الألغاز تدور حول إيجاد حل يمثل الهدف. والدمى يمكن التلاعب بها ، من قبل اللاعب ، ولكن على عكس الألغاز ليس لديهم أهداف ثابتة وأخيرًا لا يمكن التلاعب بالقصص أو تغييرها من قِبل اللاعب بل تشمل اللعب الخيالي (Fullerton, 2008) كما في شكل (٢).



شكل (٢). أنواع **الألعاب** (Fullerton, 2008).

وفقًا لمقتبس من تأليف (John Dewey's 1859-1952) "دع لعبتي تكون تعلمي ، وتعلمي هو لعبتي" ، ويشير إلى أن التعلم واللعب لا ينبغي أن يكونا مختلفين لأن التعلم يجب أن يكون مساويًا بنفس الطريقة التي يستمتع بها الطلاب بالألعاب ومتعتها (Roblyer & Doering, 2013). واتفاقًا على أهمية الألعاب تم وضع الشرط الأمثل للتعلم والذي يتطلب توفير بيئة ممتعة للتعلم من شأنها أن تساعد وتوفر تأثير إيجابي على التعلم وجعله مسليًا للطالب، وكل هذا يسهم في حماس المتعلمين للتعلم، فإذا كان المتعلمين متحمسين فسيكونون ناجحين في التعلم (Petkov & Rogers, 2011)، وعلى الرغم من أن المعلمين ليسوا غرباء على استخدام الألعاب مثل (ألعاب الطاولة ، وألعاب الورق ، وألعاب تمثيل الأدوار) في الفصول الدراسية ، إلا أن الغرض الأساسي من استخدامها هو التعلم لا التحفيز، وعند استخدامها في إطار الفصل الدراسي ، تعمل الألعاب كمساعدات تعليمية للمساعدة في شرح أو تعزيز مفهوم التعلم. في معظم الأوقات ، قد يساعد استخدام الألعاب في التعليم على تبسيط الأمور (Ments, 1999)، وهناك بعض العوامل التي تكمن وراء تحمس الطلاب لنمط العرض **خلال الألغاز Puzzles منها:**

- الألغاز Puzzles تعليمية ، و خاصة في توضيح قواعد حل المشكلات المفيدة (والقوية) بطريقة مسلية للغاية، ومن هنا ترى الباحثة أهميتها في التدريب على بناء القاعدة البرمجية في تعلم اللغات المختلفة.
- الألغاز Puzzles جذابة ومحفزة للتفكير (Michalewicz & Michalewicz , 2008). عندما يقدم المعلمون المفردات للطلاب ، يقومون بتفعيل اهتمام الطلاب وفضولهم.

نمط المصفوفات Matrix في عرض المشكلات البرمجية:

يعتبر نمط المصفوفات Matrix في عرض المشكلات البرمجية من الأنماط شائعة الاستخدام في عرض المشكلة البرمجية، واستشهد به في الأدبيات، وتؤكد العديد من الأدبيات على أنه يقلل الحمل المعرفي الدخيل ويحسن التعلم، والمصفوفة هي عبارة عن "جدول تراكمي ثنائي الأبعاد يسمح بمقارنة الموضوع بسهولة مع فئة واحدة أو أكثر(Kauffman & Kiewra, 2010). كما أنه يُظهر ارتباطات هرمية بالسمات، ويكشف العلاقات التي يسهل استنتاجها وفهمها بسهولة من قبل المتعلم (Robinson & Skinner, 1996). بالإضافة إلى ذلك ، في مراجعة لدراسات المنظم الرسومية لتحديد مبادئ التصميم الفعالة ، يصف فيكري (2002) Vekiri المصفوفة كنوع من أنواع المنظمات البيانية الذي يساعد الطلاب على تحديد العلاقات بين المفاهيم، وهو مفيد أيضًا عندما تستخدم بعد قراءة نص مطولة لدمج مفاهيم جديدة.

بدأ الباحثون في دراسة تأثير المصفوفات على التعلم منذ عدة عقود على سبيل المثال Atkinson et Robinson et al. 1998; (Gerjects & al., 2008; Bera & Robinson, 2004; ; al., 1999; ،Robinson & Skinner 1996; Robinson & Schraw 1994) Robinson & Kiewra 1995;

وفيها قام المتعلمون بدراسة المفاهيم والصفات المتعلقة بالحيوانات عن طريق المصفوفة ، أو الخطوط العريضة ، أو النص الذي قدمه الكمبيوتر.

تم تصميم المصفوفات بطريقة عرضت بها أنواع المفاهيم الفرعية للحيوان (مثل حيتان Baleen) في العمود الموجود في أقصى اليسار ، وتم عرض أسماء فئات السمة في الصف العلوي، وتم عرض السمات المقابلة في الخلايا المتبقية.

ووجد الباحثون أن المتعلمين الذين يستخدمون المصفوفة تمكنوا من الإجابة على أسئلة المقارنة بشكل أسرع من مستخدمي الخطوط العريضة أو النصوص. وقد نسبوا ذلك إلى خصائص المصفوفة، والتي تمكن المتعلمين من استخلاص استنتاجات من المواد المنظمة مكانيًا في الذاكرة العاملة في البحث العالمي (تحديد موقع أكثر من قسمين من المعلومات ثم حساب العلاقة فيما بينها)، علاوة على ذلك يؤكدون أن ميزة المصفوفة هي في فهرستها المكانية للمعلومات، مما يجعل المعلومات تثبت فعليًا لدى المتعلم، وبالتالي فإن استخدام المصفوفة يسهل البحث العالمي بشكل أسرع ، وهذا بدوره يقلل من الحمولة المعرفية الشاملة على الذاكرة العاملة. كما تم الكشف عن نتائج مماثلة في دراسة أجرتها (Gerjects et al. 2008). ويستفيد المشاركون في هذه الدراسة من استخدام أداة المقارنة القائمة على الوسائط الفائقة، والتي مكنتهم من مقارنة الأمثلة مع هياكل مختلفة في شكل مصفوفة، ومع ذلك ، يحذر فيكري (2002) Vekiri أن بعض التصاميم من المصفوفة قد تتداخل مع التعلم. حيث أن الكثير من المعلومات قد تطغى على المتعلمين مما يسبب الحمل الزائد وسوء الأداء.

وفي دراسة اشتملت على مطالبات التفسير الذاتي (Atkinson et al. 2007) كان الأداء أفضل بشكل واضح للمتعلمين الذين تم تزويدهم بتعليمات تستند إلى عرض المشكلات من خلال مثال ولكن بدون مصفوفات من المتعلمين الذين قدموا نفس التعليمات في المصفوفة. واستناداً إلى الحجة القائلة بأن بعض تصميمات المصفوفة قد تتداخل مع التعلم ، فقد اقترح (Robinson & Kiewra, 1995) مراعاة الترتيب المكاني للمعلومات التي تحد من المعالجة الإدراكية للمتعلمين ، مما يؤدي إلى ضعف الأداء. فمع الاستفادة من الترتيب المكاني ومراعاته بشكل أفضل، فإن المصفوفة تنقل العلاقات بشكل أفضل بين المفاهيم ، وبالتالي تقلل الحمل المعرفي الدخيل بينما تتطلب في نفس الوقت اهتمامًا أقل لأن المتعلم يتحرر من عبء المعنى غير المتشابك.

النتائج المختلطة للدراسة التي أجراها (Robinson & Schraw, 1994) تدعم هذه المفارقة ووجد الباحثون أن المصفوفة أدت إلى أداء أعلى من الخطوط العريضة والنص وحده بسبب قدرته على جعل العلاقات بين المفاهيم صريحة بصريا من خلال الترتيب المكاني للكلمات، فاستخدام المصفوفات يكون مفيدًا عندما يتم تصميمه بشكل مناسب (Vekiri, 2002). وقد تم دعم هذه الحجة من خلال اثنين من الدراسات الحديثة ، باستخدام مجموعة متنوعة من السقالات (Gurlitt et al. 2012; Jairam et al. 2012; Kauffman & Kiewra 2010; Suzuki et al. 2008).

و يرى الباحثون أن هذا النمط وفقًا لطريقة تصميمه التى تنص على عرض مشكلة محلولة في البداية يصلح للتدريب على التصحيح وكشف الأخطاء أثناء أداء المهام والأنشطة وذلك أثناء مرحلة بناء المفاهيم، أما أثناء التقييم النهائي فهو لا يصلح للتقييم، وإذا أردنا استخدامه للتقييم لابد من عكس تصميمه أى عرض المشكلة البرمجية المراد تقييمها ثم عرض المشكلة النموذجية المحلولة كتغذية راجعة بعد حل الأولى، وهو

ما يشبه تصميم نمط الإكمال وهو المقترح للقيام بالمستوى لثالث ويمكن استخدام الحوار بأنواع الكثيره الفردي (تفاعل الطالب مع المشكله وجهاه الشخصي بمفرده) والجماعي سواء التعاوني أو التشاركي مع هذا النمط حيث يتم التشاور حول الحل ومن ثم عرض المشكلة المحلولة والتأكد من صحتها.

نمط الإكمال للمشكلة Completion Problem في عرض المشكلات البرمجية:

قدم جيج هاريسون (Gegg-Harrison (1991&1992 اسلوبا يسمح للطلاب بملء الفراغات أو إكمالها في مهمة البرنامج، واقترح فانميرينبور (Van merrienboer (1990& 1992 هذا النمط إنجاز المشكلة كأساس لتعليم البرمجة للمبتدئين.

يستخدم نمط إكمال المشكلة **Completion Problem** برنامجًا جيد التصميم (أي برنامج مصمم من قبل خبير)، وفي هذا البرنامج المصمم جيدًا يُسمح للطلاب بإكماله وتعديله وتوسيعه، وبينما يشارك الطلاب في الإكمال والتعديل والإرشاد ، سيتعلمون مفاهيم ومهارات البرمجة. يمثل نمط إكمال المشكلة طريقة للتعلم من خلال مشكلات محلولة مع تطبيق التعلُّم، والملء الفراغي المقترح من قبل جيج هاريسون هو واحد من طرق تنفيذ التعلم من خلال النمط. وباستثناء ملء الفراغ (أو الإغلاق) ، اقترح نمط إكمال المشكلة كذلك تطبيقين للتعلم والتعديل والإرشاد.

وتعرف مهمة الإكمال بأنها مشكلة يتم فيها تزويد المتعلم بحالة معينة و حل جزئي، وبعد دراسة الحالة والمعلومات المقدمة، فإن عندها يجب على المتعلم إكمال خطوات الحل المتبقية لحل المشكلة Van (Van merrienboer & de Croock) 1990; merrienboer, 1995 ويعد هذا النهج فعال في تعلم البرمجة للمبتدئين لعدة أسباب (Van merrienboer, 1990& 1992):

- تؤدي مهام الإكمال إلى التنشيط، فالمهمة بطبيعتها تقوم على معالجة خطوات الحل المقدمة لأنها تحتوي على معلومات أساسية يحتاج إليها المتعلم قبل أن يتمكن من الاستمرار.
- بالإضافة إلى ذلك ، خطوات الحل المقدمة هي أمثلة على منهجية نظامية صحيحة لحل المشكلة.
- يمكن أن تكون أنماط الممارسة متعددة مراحل لتحفيز دوافع التعلم.
- يمكن أن تقتصر الإجابة الصحيحة على مساحة صغيرة لتبسيط عملية التشخيص المعقدة لبرنامج التتبع.

أشار أندرسون وأخرون (Anderson et al. (1984 إلى أن الطلاب غالبًا ما يستخدمون نماذج جيدة كأساس لحل مشكلات جديدة. بمعنى آخر ، يقوم الطلاب بإعادة كتابة البرنامج النموذجي الأصلي للتكيف مع مهام المشكلات الجديدة. أشار (Adelson , 1981) إلى أنه يجب على الطلاب إنشاء المعرفة لحل المشكلات من خلال دراسة بنية البرنامج ، والاهتمام بتشابه بنى البرامج المعروفة ، وتمييز أقسام البرنامج ، وتطبيق خصائص البرنامج المعروفة في البرنامج المصمم حديثًا. ومن المعروف أنه بالنسبة للمبتدئين ، فإن استخدام المشكلات المحلولة لإجراء أول ثلاث خطوات من التدريب على البرمجة (Deimel & Moffat, 1982) هي استراتيجية جيدة. نظرًا لطبيعة المبتدئين الذين لم يكونوا على دراية بمناهج البرمجة، فإن (Van Merrienboer, 1990 & 1992; Van Merrienboer & Krammer,

(1987 بدأوا نمط الإكمال، واستخدموا له برامج مصممة بشكل جيد كبرامج أساسية يجب إكمالها وتعديلها وتوسيعها.

ويقترح نمط الإكمال أن تكون محتويات التعلم أو مهارات البرمجة موجودة في البرامج المصممة جيدًا، وقد تكون هذه البرامج النموذجية مكتملة أو غير مكتملة، ومن واجب الطلاب تعديل أو تمديد البرنامج النموذجي الأصلي.

أثبت فانميرينبور Van Merrienboer (1990& 1992) أن نمط الإكمال سينتج تعلمًا فعالًا للبرمجة للمبتدئين، وفيما يتعلق بنمط الإكمال فإن محتوى التعلم هو مزيج من برامج النماذج وأنماط العمليات المطابقة، مثل الإغلاق (ملء الفراغات) والتعديل والإضافة، وتسمى هذه الأساليب المختلفة التى يقدم بها النمط عوامل التعلم التي تلد مجموعة متنوعة من الأسئلة للمبتدئين للتعلم، والأساليب الثلاثة هم:

١. اسلوب الإغلاق : يتم ترك عدة أجزاء من البرنامج الكامل ويحتاج الطلاب إلى ملء الإجابات الصحيحة. تمثل الفراغات المفاهيم المهمة التي يريد النظام أن يتعلمها الطلاب .

٢. اسلوب التعديل: يتم إعادة كتابة البرنامج لتحسينه، أو تبسيطه، أو استخدام برنامج آخر لأداء نفس الوظيفة، والغرض من هذا التعديل هو السماح للطلاب بفهم معنى الخوارزمية والدور الذي تلعبه كل عبارة في البرنامج.

٣. اسلوب الإضافة : يتم فيه إضافة معايير جديدة أو أوامر جديدة إلى الأسئلة الأصلية ، ويطلب من الطلاب إعادة كتابة البرنامج القديم عن طريق إضافة أو طرح بعض الخطوات، ويجب على الطلاب إعادة كتابة البرنامج النموذجي لتلبية المتطلبات الجديدة، والغرض من اسلوب الإضافة هو تحفيز قدرة الطلاب على وضع استراتيجيات حل المشكلات.

ويصلح هذا النمط لتدريب الطلاب على بنية المشكلة البرمجية بشكل جيد بسبب تصميه فهو يسمح للطلاب بعرض مشكلة محلولة نموذجية ودراستها وفي المقابل لها لديه مشكلة تنقصها عدة خطوات يحتاج لإكمالها، وهو أفضل من النمط السابق بسبب وضعية المقارنة التى تسمح للطالب بعرض المشكلتين بشكل متوازى، لدراسة بنيتها بشكل جيد والرجوع إليها في حل المشكلة الأخرى.

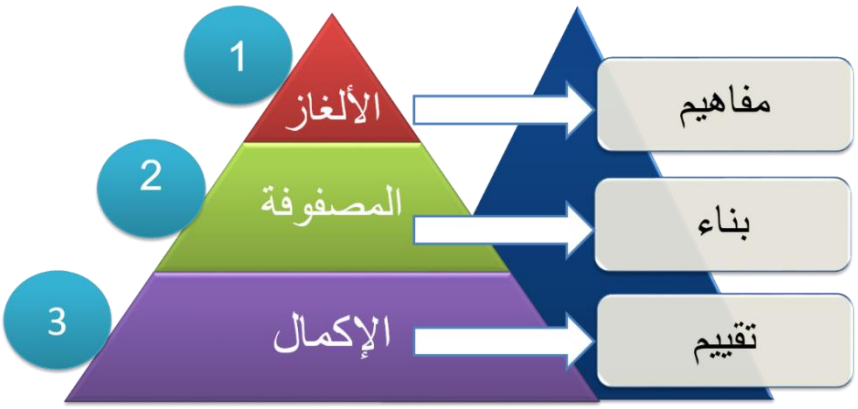
وأخيرًا يرى الباحثون أن هذا النمط لا يصلح للتدريب على تصحيح واكتشاف الأخطاء والتقييم بسبب طريقة تصميمه التى تقتضي بعرض مشكلتين احداهما محلولة بشكل نموذجي والأخرى توازيها مما يساعد الطالب في كشف الحل بسهولة ويصعب تقييم فهمه لها.

أسس تكامل أنماط عرض المشكلات البرمجية في إستراتيجية تكامل أنماط العرض في حل المشكلة البرمجية:

تأسيسا علي ماسبق عرضه من خصائص وإمكانات أنماط عرض المشكلات البرمجية استخلص الباحثون أسس توظيف هذه الأنماط في الإستراتيجية لتعلم البرمجة ووتنمية كفايات حل المشكلة البرمجية كالتالي وكما تتضح في شكل (٣):

- نمط الألغاز هو الأنسب لعرض مشكلات المفاهيم.
- نمط المصفوفة هو الأنسب لعرض مشكلات البناء.
- نمط الإكمال هو الأنسب لعرض مشكلات التقييم مع عكس التصميم.

وهذا ما يوضحه الشكل (٣):



شكل (٣): يوضح التكامل بين أنماط عرض المشكلة البرمجية في تعلم كفايات البرمجة.

المحور الثالث : أسس تطوير الاستراتيجية القائمة على تكامل أنماط عرض المشكلات البرمجية لتنمية كفايات البرمجة (C++) في بيئة التعلم المدمج (الإجابة عن السؤال الأول):

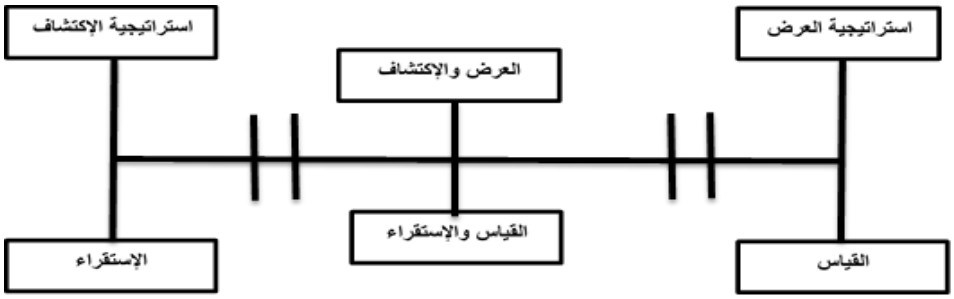
كثير من العلماء والباحثين في مجال التعلم عمل علي وضع مفهوم لاستراتيجية التعلم ، وعلي الرغم من اختلاف تناولهم للمصطلح فإن جميع التعريفات تشير إلي طرق معالجة المتعلم للمعلومات . فقد عرف اينشتاين وماير (١٩٨٦) إستراتيجيات التعلم بأنها السلوكيات والأفكار التي يستخدمها المتعلم في أثناء التعلم والتي تهدف إلى التأثير على عملية الترميز التي يقوم بها المتعلم. وعرفها ماير (١٩٨٨) وبشكل أكثر تحديدًا أنها سلوكيات المتعلم والتي تهدف إلى التأثير على كيفية قيام المتعلم بعمليات معالجة المعلومات.

ولقد عرف عبد العظيم الفرجانى (٢٠٠٢، ص ١٢٧) استراتيجية التعلم (Learning Strategy) على أنها خطة محكمة البناء مرنة اثناء التطبيق، وهي محاولة للاختيار الأمثل لكل عنصر من عناصر العملية التعليمية قبل التنفيذ، وعرض مراحل استراتيجية للتعلم لوالتر ديك (Walter Dick) وهي: الأنشطة قبل التعليمية، تقديم المعلومات، مشاركة الدارس، الاختبارات، المتابعة.

واشار محمد خميس (٢٠٠٣آ، ص ١٠٩) إلى أن " الاستراتيجية بمعناها العام هي خطة منظمة، تتكون من مجموعة محددة من الأنشطة والإجراءات، مرتبة في تسلسل معين، لتحقيق أهداف معينة في فترة زمنية محددة ".وفي ضوء هذا التعريف يرى وجود أنواع عديدة من الاستراتيجيات التعليمية، والتي تكون في مجملها استراتيجية التعليم العامة. ومن أنواع هذه الاستراتيجيات: استراتيجية خاصة بتنظيم المحتوى وتتابع عرضه، واستراتيجيات خاصة بأساليب التعليم / والتعلم، واستراتيجيات التفاعلات التعليمية.

وتتمركز استراتيجيات التعلم حول استراتيجيتين أساسيتين تقعان على خط متصل، في أحد طرفيه توجد استراتيجية العرض (Expository) أوالاستقبال (Reception)، وفي الطرف الآخر استراتيجية الاكتشاف (Discovery) أو التقصى (Inquiry)، وتتدرج المستويات بينهما باختلاف مواقعها على الخط

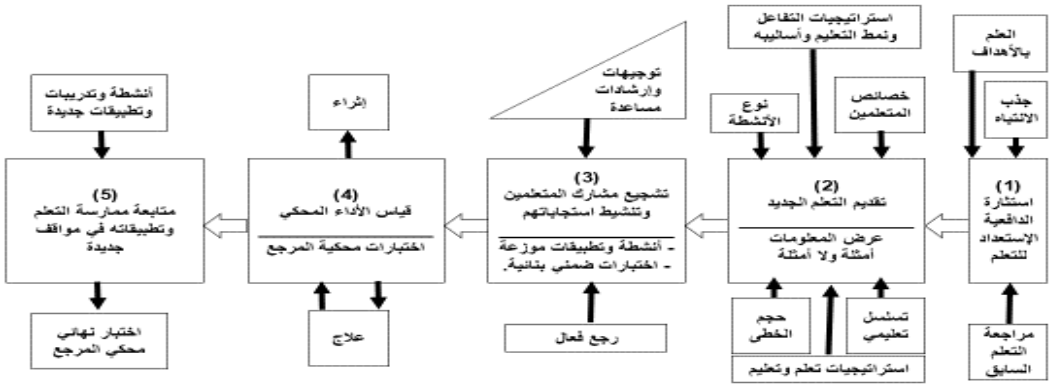
حتى نصل إلى المنتصف فنجد إستراتيجية تجمع بين الاستراتيجيتين، ويرتبط بهاتين الاستراتيجيتين أسلوبان هما: القياس أو الاستنباط (Deduction)، والاستقراء (Induction).



شكل (٤): *يوضح استراتيجيات التعلم على خط متصل.* (محمد عطية خميس، ٢٠٠٣-أ، ص ١٦٣)

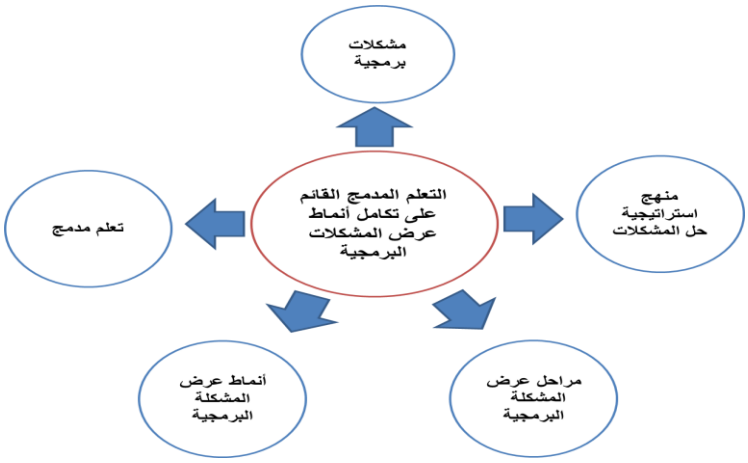
وتتعدد نماذج استراتيجيات التعلم العامة وذلك كما ورد في الشحات سعد محمد عتمان (٢٠١٢) وتناولها محمد خميس (٢٠٠٣-أ، ص ص ١٩٠- ٢٠١، عبداللطيف الجزار، ١٩٩٥، ٢٠٠٢) بالعرض والمناقشة، فمثلًا نموذج "جانيه وبريجز" تضمن تسع خطوات رئيسة هي: جذب الانتباه، تعريف المتعلم بالأهداف التعليمية، استرجاع التعلم السابق، عرض المادة التعليمية المثيرة للتعلم الجديد، توجيه التعلم، تحفيز الأداء، تقديم الرجع، قياس الأداء، تحسين الاحتفاظ بالتعلم وانتقاله إلى مواقف جديدة.

وقد راع الباحثون نموذج جانييه (Gagne' & Briggs, 1979) لخطوات وإجراءات ومراحل استراتيجيات التعلم العامة، إلى جانب أسس ومكونات الاستراتيجية المقترحة التي ستعرض تاليًا قام الباحثون بتصميم استراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج.



شكل (5). *نموذج محمد عطية خميس لإستراتيجية التعلم العامة.*

بعد الإطلاع على الأدبيات التربوية الخاصة بالمشكلات البرمجية توصل الباحثون إلى مكونات الإستراتيجية المقترحة، والتي تتكون من منهج استراتيجية حل المشكلات في البرمجة، مراحل تعلم البرمجة، مشكلات برمجية، أنماط عرض المشكلة البرمجية، تعلم مدمج، كما في شكل (٦).



شكل (٦): *مكونات الاستراتيجية المقترحة.*

وفيما يلي تفصيل لمكونات الإستراتيجية المقترحة من شكل (٦):

- **التعلم المدمج وتصميمه التتابعي Sequential Design of Blended Learning:**

اختار الباحثون التصميم التتابعي للتعلم المدمج لملائمته استراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتعلم البرمجة وكفايات حل المشكلات البرمجية للغات الحديثة، وقد تم عرض ذلك ومبرراته في المحور الأول من هذا الإطار النظري، ويتم استخدام تصميم التعلم المدمج بعد هند الرغيب و عبداللطيف الجزار وأحمد نوبي (Alraghaib, Elgazzar, & Nouby, 2015, p. X) باستراتيجية التتابع التالي: تعليم مباشر (F2F) ، تعليم مباشر (F2F)، تعلم إلكتروني (E-Learning) ، تعلم إلكتروني (E-Learning)، تعلم إلكتروني (E-Learning) ، ويتم توظيف تكامل أنماط عرض المشكلة البرمجية في التعلم الإلكتروني (E-Learning) ، كما في شكل (٧).

- **منهجية أسلوب/استراتيجية حل المشكلات في البرمجة:**

تم اختيار واتباع منهجية أسلوب /استراتيجية المراحل أو الخطوات الثلاث (Dale & Weems, 2008) لحل المشكلات البرمجية:

١. مرحلة حل المشكلة Problem solving phase :

وتتم على عدة خطوات وهي : (أ) التحليل وتحديد المواصفات (analysis and specification): وتعني فهم وتحديد المشكلة وتحديد ما يجب فعله لحل هذه المشكلة. (ب) وضع الحل العام (الخوارزمية) (General Solution): وهنا يتم تحديد الكائنات (الخطوات) وكيفية تفاعلها مع بعضها البعض لحل المشكلة. (ج) التحقق (Verify): وهنا يتم إتباع الخطوات كما تم وصفها في الخوارزمية لحل المشكلة والتحقق من صحتها.

٢. **مرحلة التنفيذ Implementation Phase :**

وتتم على الخطوات التالية: (أ) تنفيذ الحل (البرنامج) (Concert Solution (Program: ويتم هنا ترجمة الخوارزمية (الحل العام) للغة البرنامج. (ب) اختبار البرنامج (Test): ويتم هنا تشغيل البرنامج والتحقق من النتائج، فإذا تواجدت أخطاء، يتم تحليل البرامج والحل العام لتحديد مصدر الأخطاء وتصحيحها.

٣. **مرحلة الصيانة Maintenance Phase:** بمجرد كتابة البرنامج فإنه يدخل في المرحلة التالثة، وهي الصيانة، ويتم فيها: (أ) الاستخدام Use : ويتم فيها استخدام البرنامج وتشكليه. (ب) الصيانة Maintain : ويتم فيها تعديل البرنامج لتلبية المتطلبات المتغيرة أو لتصحيح الأخطاء التى تظهر عند استخدامه.

- **بعض جوانب كفايات البرمجة للغة (C++) المستهدفة:**

قام الباحثون بمراجعة الأدبيات المتصلة بلغة البرمجة الحديثة (C++)، وتوصيف مقرر البرمجة (C++) من وحدة الجودة، قسم تكنولوجيا التعليم والمعلومات (٢٠١٩). وتطبيق منهج تحليل المحتوي كما عرفه عبداللطيف الجزار(٢٠١٧) للمواد التعليمية للمقرر، أمكن الباحثون التوصل إلي هذه القائمة لجوانب كفايات البرمجة للغة (C++) لاستهدافها كأساس للإستراتيجية:

١. الإلمام بمفاهيم البرمجة وعملياتها الأساسية.
٢. الإلمام بمفاهيم لغة البرمجة " C++"، وبنيتها الأساسية.
٣. الإلمام بالمتغيرات Variables في لغة البرمجة " C++".
٤. استخدام المتغيرات في لغة البرمجة " C++".
٥. الإلمام بالمدخلات والمخرجات Input/ Output في لغة البرمجة " C++".
٦. استخدام المخرجات والمخرجات في لغة البرمجة " C++".
٧. الإلمام بالجمل والتعبيرات الحلقية Loop Statements & Expressions في لغة " C++".
٨. استخدام الجمل والتعبيرات الحلقية في لغة البرمجة " C++".
٩. الإلمام بالقرارات Decisions في لغة البرمجة " C++".
١٠. استخدام القرارات في اللغة " C++".

- **أنواع المشكلات البرمجية :**

توجد ثلاثة أنواع من المشكلات غالبا تواجه تعلم البرمجة وحل المشكلات البرمجية، وقد حلل سعيد هاجرويت (Hadjerrouit, 2008) هذه الأنواع للمشكلات البرمجية: **مشكلات قائمة على المفاهيم، ومشكلات قائمة على بناء الأنشطة، ومشكلات خاصة بالتقييم.**

- **أنماط عرض المشكلات البرمجية:**

عرف الباحثون إجرائيًا نمط عرض المشكلة بإنه طريقة/ وسيلة لتعليم وتعلم المشكلة والتدريب عليها حيث يقوم بعرض المشكلة بشكل ما يوضح هدفها ومكوناتها وصولًا لحلها. وعرض الباحثون مسبقًا ثلاثة أنماط تم تتبعها وحصرها استخدمت لعرض المشكلة البرمجية كلًا منها على حدا.

- **مراحل تعلم البرمجة:**

وقد قام الباحثون بعرضها مسبقًا في المحور الأول وكانت المراحل كالتالي:

١. مرحلة تعلم المفاهيم.
٢. مرحلة بناء التعلم.
٣. مرحلة التقييم.

- **أسس النموذج الأولي لتصميم الإستراتيجية تكامل أنماط عرض المشكلات البرمجية لتعلم كفايات البرمجة (C++) ببيئة للتعلم المدمج:**

بعد أن قام الباحثون بعرض أسس ومكونات الاستراتيجية وعرض أنماط المشكلات التي تم التوصل لها، وأنواع هذه المشكلات، وتمرير كل نمط من أنماط العرض على أنواع المشكلات الثلاثة والمفاضلة بينهم لمعرفة ما النمط الذي يلائم كل مشكلة من المشكلات الثلاثة، وتحليل المحتوى ومعرفة الكفايات البرمجية لمقرر " لغات برمجة حديثة (٢) C++" ، أمكن وضع أسس النموذج الأولي (Prototype) لمراحل الاستراتيجية المقترحة لتكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج:

أولًا: مرحلة الإعداد والتهيئة في بيئة التعلم التقليدي:

هذه المرحلة مهمة جدًا لتهيئة الطالبات وتحفيزهم للتعلم، وشرح خطوات الإستراتيجية وبيئة التعلم التي تحتوى على أنماط العرض المستخدمة حيث يتم تدريبهم على استخدام هذه الأنماط والإبحار داخل البيئة وعرض مكوناتها بشكل عام أمام الطالبات، وتبدأ هذه المرحلة بعقد جلسة عامة مع الطالبات لتهيئتهم وإثارة دافعيتهم لبدء التعلم الجديد، ثم تعريفهم بالأهداف التعليمية للمقرر، والمطلوبة منهم في نهاية التعلم، وتجميع بياناتهم المطلوبة لتسجيل دخولهم للبيئة (الاسم، الإيميل، رقم التليفون)، ثم توضيح الإستراتيجية التي سيتم بها تدريس المقرر، وطريقة الدمج بين البيئتين التقليدية والإلكترونية بشكل عام ومعلن للطالبات، ويعقب ذلك تحميل برنامج Dev C++؛ لتنفيذ برامج لغتي C ,C++على أجهزة الطالبات، ثم تدريب الطالبات على تسجيل الدخول إلى الموقع الإلكتروني، والتعامل مع مكونات واجهة التفاعل، وتعليمات السير في الموديولات التعليمية، حتى تألفه الطالبات، ويقبلوا عليه. وبعد ذلك تدريبهم على كيفية التعامل مع أنماط عرض المشكلات المستخدمة في البحث الحالي، والتنقل بينها، ثم إرسال بيانات تسجيل كل طالب على الإيميل الخاص به (Username, Password, Site link) في موعد محدد ومعلوم للطالبات، وبعد تسجيل الطالبات الدخول إلى صفحة الموقع الالكتروني، تقوم كل طالبة بقراءة مقدمة المقرر والتعليمات بدقة، وفهمها جيدًا قبل الدخول للموديول الأول من موديولات المقرر، ثم أداء الطالبات للاختبار القبلي للمقرر، وحل جميع بنود هذا الاختبار، للوقوف على مدى حاجتهم لدراسة المقرر، وذلك من خلال مقارنة درجات كل طالبة بالدرجة المحكية (٩٠%) من درجة هذا الاختبار.

ثانيًا: مرحلة عرض وتكوين المفاهيم البرمجية والقواعد البرمجية في البيئة التقليدية:

وفي هذه الخطوة سيتم شرح الجانب المعرفي للمفاهيم البرمجية وبينة اللغة من الأكواد المكونة لها في بيئة التعلم التقليدية ثم شرح مشكلة محلولة كمثال للطلاب يوضح لهم كيفية بناء المعرفة وربط ما سبق تعلمه من اكواد مع الأكواد الجديدة وترتيبها في ضوء قواعد اللغة بشكل يسمح بحل المشكلة البرمجية ، وتبدأ بقيام المعلم بتوضيح الهدف المراد تدريسه إلى الطالبات، ثم شرح المفهوم البرمجي الذي يتناوله الهدف، وتوضيح كافة جوانبه المعرفية (تعريف، خصائص، تكوين الجملة البرمجية، ...)، وعرض مشكلة محلولة على الطالبات لتعلم بنى وتراكيب حل المشكلة البرمجية، ثم شرح طريقة استخدام برنامج Dev C++ لتنفيذ المشكلات البرمجية من خلاله، ثم فحصها وصيانتها.

ثالثًا: مرحلة التدريب على تكوين المفاهيم والقواعد البرمجية من خلال نمط الألغاز (Puzzle) في بيئة التعلم الإلكترونية:

بعد شرح الجانب المعرفي للمفاهيم البرمجية وبينة اللغة من الأكواد المكونة لها في بيئة التعلم التقليدية، ومن ثم الإنتقال إلى البيئة الالكترونية عقب الإنتهاء من شرح معارف ومفاهيم اللغة للتدريب على بنية الكود بشكل فردي من خلال اللغز المكون من صورة يقوم الطالب بترتيب أجزاء الكود المكونة عليها بالشكل المطلوب لحل المشكلة التي أمامه، ثم ظهور التغذية الراجعة له، وتتم هذه المرحلة بدءًا من توجيه الطالبات إلى بيئة التعلم الالكترونية للتدريب على تكوين المفاهيم التي تم شرحها، وبعض المشكلات التي تنمي تعلم بنى وتراكيب حل المشكلات البرمجية لديهم، ومن ثم تقييم ما تم تعلمه والتدريب عليه، ثم تسجيل الطالبات دخولهم مرة أخرى إلى صفحة الموقع الالكتروني، والدخول إلى الموديول الأول، وقراءة الطالبات لصفحات المقدمة، وتعليمات السير في المودبول، والأهداف التعليمية وعناصر المحتوى، مع ملاحظة أن الموديول سيفتح فقط في حالة أداء الأختبار القبلي، ثم دخول الطالبات إلى القسم الأول من الأقسام الثلاثة، والخاص بالتدريب على تكوين مفاهيم الجمل البرمجية المتاحة لهم في هذا الوقت تزامنًا مع ما تم شرحه في المحاضرة التقليدية من مفاهيم برمجية، وبعدها دخول الطالبات على المفهوم المراد التدرب عليه، ثم عرض المشكلة على الطالبات، ثم تقوم الطالبة بقرائتها جيدًا، بحيث يميز بين المعطى والمطلوب، وتقديم حل المشكلة في شكل لغز Puzzle (صورة) تم تقسيمها إلى عدة أجزاء مرقمة وعلى كل جزء يوجد قطعة من الكود المطلوب لحل المشكلة البرمجية، وقيام الطالبات بترتيب أجزاء الكود من خلال قيامهم بسحب وإلقاء الكود (Drag and Drop) على خلفية الصورة التي تجدها الطالبات أسفل صورة اللغز غير المرتبة، والتي هي نفس الصورة السابقة ولكن خالية من الأكواد وبالتسلسل الصحيح للأرقام، ثم قيام الطالبات بالضغط على Submit and Finish بعد الإنتهاء من الحل، وظهور التغذية الراجعة للطالبات بعد كتابة الترييب الصحيح للمشكلة، والتي تحتوى على الإجابة الصحيحة لحل اللغز، وعدد الحركات التي نجحت في اتمامها، ووضع الكود في المكان الصحيح بالفعل.

رابعًا: مرحلة التدريب على بناء المشكلة وتركيبها الصحيح من خلال نمط المصفوفة (Matrix) في بيئة التعلم الإلكترونية:

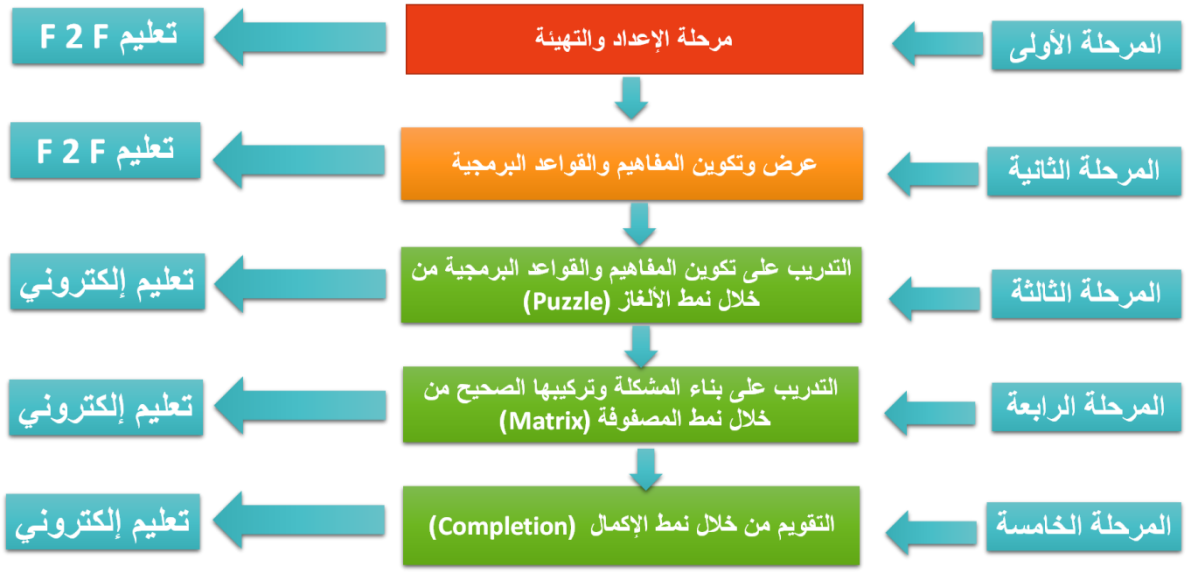
وفي هذه الخطوة وبعد قيام الطالبات بالتدريب على المفاهيم باستخدام نمط الألغاز تنتقل الطالبة إلى القسم الثاني للتدريب على بناء المشكلات من خلال عرض أمثلة نموذجية باستخدام نمط المصفوفة وهي عبارة عن جدول مكون من عمودين، الأول به المثال الذي قام المعلم بعرضه وذلك لمراجعته ودراسته ببطء وتمعن، والعمود الأخر لمشكلة مكافئة للأولي بها خطوة أو خطوتان ناقصة يقوم الطالب بإكمالها من خلال اختيار الإجابة الصحيحة من بين عدة إجابات موجودة أسفل المصفوفة، وذلك للتأكد من فهمه وبنائه الصحيح للمعرفة البرمجية، ثم عرض التغذية الراجعة التي تخبره بمدى صحة حله، وتبدأ هذه المرحلة بدخول الطالبات إلى القسم الثاني من الأقسام الثلاثة، والخاص بالتدريب على بنى وتراكيب حل المشكلة البرمجية وبدائلها المتاحة لهم في هذا الوقت تزامنًا مع ما تم عرضه في المحاضرة التقليدية من مفاهيم برمجية وأساسيات اللغة، ثم دخول الطالبات على انشطة القسم المفتوحة بالترتيب الموجود والمنظم من السهل إلى الصعب، وعقب الدخول تجد الطالبة نفسها أمام مصفوفة Matrix، تم تصميمها على شكل جدول مكون من عمودين، الأول به المثال المحلول الذي قام المعلم بعرضه للمشكلة في البيئة التقليدية وحلها وذلك لمراجعته ودراسته ببطء وتمعن من قبل الطالب، والعمود الأخر به مشكلة مكافئة للمشكلة لأولي وبها خطوة أو خطوتان ناقصة، ثم قيام الطالبات بدراسة المثال الأول والمحلول أمامهم، ثم الإنتقال إلى المشكلة المقابلة

لها في العمود الثاني، وقرائتها جيدًا ثم التمييز بين المعطى والمطلوب، ودراسة خطوات الحل الموجودة، وبعدها قيام الطالبات بإكمال الخطوات الناقصة للمشكلة المعروضة من خلال اختيارها للإجابة الصحيحة من بين عدة إجابات، وذلك للتأكد من فهمها وبنائه الصحيح للمعرفة البرمجية، ثم عرض التغذية الراجعة التي تخبرهم بمدى صحة حلهم، ثم تنفيذ الحل المشكلة الثانية على برنامج Dev C++، ثم صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية، وبعدها الإنتقال للتدريب على مشكلات جديدة من خلال مصفوفات أخرى، وبعد الإنتهاء من التدريب، يقوم الطالب بالإنتقال للقسم الثالث.

خامسًا: مرحلة التقويم من خلال نمط الإكمال (Completion) في بيئة التعلم الالكتروني:

بعد قيام الطالبات بالإنتهاء من التدريب على مرحلتي المفاهيم و وبني المشكلات البرمجية، تقوم الطالبات بالإنتقال للقسم الثالث وذلك للتقويم على ما سبق من خلال نمط الإكمال، وتبدأ هذه المرحلة بقيام الطالبات بالإنتقال إلى القسم الثالث، وفتحه، ثم فتح صفحة التقويم البنائي الأول، وبعد دخول الطالبة إلى الصفحة ، تعرض عليها مشكلة برمجية عليها أن تقرأها بشكل جيد تجد الطالبة خطوات الحل أسفل المشكلة تنقصها نقاط على الطالبة أن تكملها لكي تجتاز هذا التقويم وتنتقل لغيره ثم قيام الطالبة بإكمال الخطوات الناقصة من خلال كتابتها لحل المشكلة مرة أخرى كاملة في محرر نص خاص بذلك، ثم تضغط الطالبة على مفتاح Submit and Finish بعد إنهاء كتابة الحل، ثم عرض التغذية الراجعة على الطالبة التي تخبرها بمدى صحة حلها، تنفيذ حل المشكلة على برنامج Dev C++ ثم صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية، ثم تأخذ الطالبة لقطة شاشة للحل الخاص بكل مشكلة تقويم، وتقوم بإرسالها إلى ايميل المعلم كدليل على قيامه بهذه الخطوة، ثم الإنتقال لتقويم المشكلة الجديدة التي ستفتح عقب حل مشكلة التقويم السابقة، والرجوع مرة أخرى إلى بيئة التعلم التقليدية لدراسة مفاهيم برمجية جديدة حتى ينتهي التعلم المطلوب ويبدأ تعلم جديد، وهكذا إلى نهاية المقرر، وتنتهي هذه المرحلة بانتقال الطالبة لأداء الاختبار البعدي للمقرر عقب الإنتهاء من كل المشكلات المطلوبة للتقويم في المقرر.

ويمكن عرض الشكل المبدئي لأساس النموذج الأولي (prototype) لاستراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة التعلم المدمج في الشكل (٧).



شكل (٧) الشكل المبدئي لأساس النموذج الأولي لاستراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة التعلم المدمج

إجراءات البحث:

قام الباحثون بالإجراءات التالية للتوصل إلى الصورة النهائية لخطوات الإستراتيجية القائمة على تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج :

- تصميم الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج.
- إعداد أداة البحث وهي بطاقة التحكيم علي الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات.
- تطبيق أداة البحث وهي بطاقة التحكيم علي الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات علي المشاركين في البحث .

أولا : تصميم الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج:

قام الباحثون باشتقاق الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج، وذلك في ضوء ما تم عرضه بالإطار النظري للبحث في المحور الثالث خاصة مراحل الإستراتيجية، تم تصميم الصورة المبدئية في (٥) خمسة مراحل رئيسية، (50) خمسون خطوة فرعية كما يعرضها جدول (١)، وبعد مراجعة الباحثون لهذه الصورة وإجراء التعديلات اللازمة، أصبحت الصورة المبدئية جاهزة للتحكيم عليها، ثم قام الباحثون بإعداد أداة البحث للتحكيم علي الصورة المبدئية.

ثانيا :إعداد أداة البحث وهي بطاقة التحكيم علي الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات:

قام الباحثون بتحديد الهدف من بطاقة التحكيم علي الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات وهو التوصل إلي استراتيجية صادقة وصحيحة تكامل أنماط عرض المشكلات، وذلك للتعرف على آراء المحكمين في البطاقة، وإضافة أو حذف أو تعديل ما يرونه مناسبا.ومن ثم تم وضع مراحل وخطوات الإستراتيجية التي تم تصميم صورتها المبدئية في بطاقة (Checklist) في شكل جدول مكون من قائمة بها المراحل الرئيسية، ويندرج تحت كل مرحلة رئيسية الإجراءات الفرعية الخاصة بها. وتتكون استبانة التحكيم من ثلاث خانات وهي: مهم، إلى حد ما، غير مهم ، لبيان مدى أهمية كل من مراحل الإستراتيجية المذكورة بالقائمة وخطواتها الفرعية، كما تم سؤالهم عن إضافة أي مراحل أخرى يرونها واقتراح الخطوات الإجرائية للمرحلة أو المراحل التي يضيفونها، وتم مراجعتها بواسطة ثلاثة من أعضاء هيئة التدريس بقسم تكنولوجيا التعليم والمعلومات بكلية البنات بجامعة عين شمس للتأكد من تحقيقها للغرض الذي وضعت من أجله وهو التحقق من صدق الإستراتيجية.

ثالثا: تطبيق أداة البحث وهي بطاقة التحكيم علي الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات علي المشاركين في البحث:

قام الباحثون بالتواصل مع المحكمين المشاركين في البحث لتسليم الصورة المبدئية لاستراتيجية تكامل أنماط عرض المشكلات البرمجية، ومنهم من طلب ايصال نسخة الصورة المبدئية بشكل ورقي، ومنهم من طلب ارسال النسخة الإلكترونية منها على الإيميل الخاص به، وطلب أحد المحكمين موعدًا مع أحد الباحثين وتحكيم البحث ومناقشته خلال الجلسة، وبعد الإنتهاء من التحكيم قام الباحثون بجمع بطاقات

التحكيم الورقية من المحكمين، واستلام الإلكتروني منها عن طريق الإيميل ثم تم فلترة وتحليل بيانات البطاقات لجمع البيانات المطلوبة وتعديل اللازم وفقًا لأراء السادة المحكمين.

نتائج البحث ومناقشتها والتوصل إلي الشكل النهائي للإستراتيجية

قام الباحثون بتنظيم وتحليل بيانات البطاقات التي تم تحكيمها بعد تطبيقها علي المشاركين في البحث من المحكمين علي استراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج، وذلك للإجابة عن السؤال الفرعي الثاني الذي ينص على كيف يمكن بناء استراتيجية لتكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية تلك الكفايات؟

قام الباحثون بحساب نسب التكرارات الأراء السادة المحكمين على الإجراءات الفرعية، وتم حساب التكرارات على مقياس ثلاثي متدرج (مهم، الى حد ما، غير مهم)، فتم أخذ الإجراءات الهامة والتي حصلت على نسب تصل إلى ٧٠%، وحذف الإجراءات التي حصلت على نسب أقل من ذلك، كما في جدول (١) ، وقد قام الباحثون بحساب نسبة الإتفاق التي أبداها السادة المحكمين على مدى أهمية كل مرحلة من المراحل وفقًا للخطوات والمعادلات التالية:

(١) حساب نسبة الإتفاق أهمية كل إجراء فرعي وفقًا لأراء السادة المحكمين من خلال المعادلة التالية:

$$\frac{\text{عدد المحكين الذين اتفقوا على أهمية كل إجراء}}{\text{عدد المحكين الكلي}}$$

(٢) حساب متوسط نسبة الإتفاق على أهمية المرحلة بأكملها من خلال حساب متوسط نسبة المؤشرات من خلال المعادلة التالية:

$$\frac{\text{مجموع نسب الإجراءات الفرعية}}{\text{عددها}}$$

(٣) حساب متوسط نسبة الإتفاق على الإستراتيجية من خلال المعادلة التالية:

$$\frac{\text{مجموع نسب المراحل}}{\text{عددها}}$$

وقد تم حساب **التكرارات والنسب المئوية للإتفاق لإراء السادة المحكمين كما في جدول (١)**، .كما تم تجميع وتحليل ملاحظات السادة المحكمين، ومتوسط نسبة الأتفاق علي كل مرحلة، ومتوسط نسبة الأتفاق علي الإستراتيجية ككل.

جدول (١) نسبة اتفاق السادة المحكين على أهمية كل مرحلة بإجراءتها:

م	خطوات الإستراتيجية	(التكرار والنسب المئوية)		
		مهم	إلى حد ما	غير مهم

		ت %	ت %	ت %
أولًا:	**مرحلة الإعداد والتهيئة في بيئة التعلم التقليدي**			
١.	عقد جلسة عامة مع الطالبات لتهيئتهم وإثارة دافعيتهم لبدء التعلم الجديد.	٩ (١٠٠%)		
٢.	تعريف الطالبات بالأهداف التعليمية للمقرر، والمطلوبة منهم في نهاية التعلم.	٩ (١٠٠%)		
٣.	تجميع بيانات الطالبات المطلوبة لتسجيل دخولهم للبيئة (الاسم، الإيميل، رقم التليفون).	٩ (١٠٠%)		
٤.	توضيح الإستراتيجية التي سيتم بها تدريس المقرر، وطريقة الدمج بين البيئتين التقليدية والإلكترونية بشكل عام ومعلن للطالبات.	٩ (١٠٠%)		
٥.	تحميل برنامج Dev C++ ؛ لتنفيذ برامج لغتي C, C++ على أجهزة الطالبات.	٩ (١٠٠%)		
٦.	تدريب الطالبات على تسجيل الدخول إلى الموقع الإلكتروني، والتعامل مع مكونات واجهة التفاعل، وتعليمات السير في الموديولات التعليمية، حتى تألفه الطالبات، ويقبلوا عليه.	٩ (١٠٠%)		
٧.	تدريب الطالبات على كيفية التعامل مع أنماط عرض المشكلات المستخدمة في البحث الحالي، والتنقل بينها.	٨ (٨٨,٩%)	١ (١١,١%)	
٨.	إرسال بيانات تسجيل كل طالب على الإيميل الخاص به (Username, Password, Site link) في موعد محدد ومعلوم للطالبات.	٩ (١٠٠%)		
٩.	تسجيل الطالبات الدخول إلى صفحة الموقع الالكتروني.	٩ (١٠٠%)		
١٠.	قراءة مقدمة المقرر والتعليمات بدقة، وفهمها جيدًا قبل الدخول للموديول الأول من موديولات المقرر.	٩ (١٠٠%)		
١١.	أداء الطالبات للاختبار القبلي للمقرر، وحل جميع بنود هذا الاختبار، للوقوف على مدى حاجتهم لدراسة المقرر، وذلك من خلال مقارنة درجات كل طالبة بالدرجة المحكية (٩٠%) من درجة هذا الاختبار.	٩ (١٠٠%)		
١٢.	**أخرى:**			
ثانيًا:	**مرحلة عرض تكوين المفاهيم البرمجية والقواعد البرمجية في البيئة التقليدية:**			
.	قيام المعلم بتوضيح الهدف المراد تدريسه إلى الطالبات.	٩ (١٠٠%)		
.	شرح المفهوم البرمجي الذي يتناوله الهدف، وتوضيح كافة جوانبه المعرفية (تعريف، خصائص، تكوين الجملة البرمجية، ...).	٩ (١٠٠%)		
.	عرض مشكلة محلولة على الطالبات لتعلم بنى وتراكيب حل المشكلة البرمجية.	٩ (١٠٠%)		
.	شرح طريقة استخدام برنامج Dev C++ لتنفيذ المشكلات البرمجية من خلاله، ثم فحصها وصيانتها.	٩ (١٠٠%)		
.	**أخرى:**			
ثالثًا:	**مرحلة التدريب على تكوين المفاهيم والقواعد البرمجية من خلال نمط الألغاز (Puzzle) في بيئة التعلم الإلكترونية:**			
١٨.	توجيه الطالبات إلى بيئة التعلم الالكترونية للتدريب على تكوين المفاهيم التي تم شرحها، وبعض المشكلات التي تنمي تعلم بنى وتراكيب حل المشكلات البرمجية لديهم، ومن ثم تقييم ما تم تعلمه والتدريب عليه.	٧ (٧٧,٨%)	٢ (٢٢,٢%)	
١٩.	تسجيل الطالبات دخولهم مرة أخرى إلى صفحة الموقع الالكتروني.	٩ (١٠٠%)		

٢٠.	الدخول إلى الموديول الأول، وقراءة الطالبات لصفحات المقدمة، وتعليمات السير في الموديول، والأهداف التعليمية وعناصر المحتوى، مع ملاحظة أن الموديول سيفتح فقط في حالة أداء الأختبار القبلي.	٩ (١٠٠%)	
٢١.	دخول الطالبات إلى القسم الأول من الأقسام الثلاثة، والخاص بالتدريب على تكوين مفاهيم الجمل البرمجية المتاحة لهم في هذا الوقت تزامنًا مع ما تم شرحه في المحاضرة التقليدية من مفاهيم برمجية.	٩ (١٠٠%)	
٢٢.	دخول الطالبات على المفهوم المراد التدرب عليه.	٩ (١٠٠%)	
٢٣.	عرض المشكلة على الطالبات، ثم تقوم الطالبة بقرائتها جيدًا، بحيث يميز بين المعطى والمطلوب.	٩ (١٠٠%)	
٢٤.	تقديم حل المشكلة في شكل لغز Puzzle (صورة) تم تقسيمها إلى عدة أجزاء مرقمة وعلى كل جزء يوجد قطعة من الكود المطلوب لحل المشكلة البرمجية.	٩ (١٠٠%)	
٢٥.	قيام الطالبات بترتيب أجزاء الكود من خلال قيامهم بسحب وإلقاء الكود (Drag and Drop) على خلفية الصورة التي تجدها الطالبات أسفل صورة اللغز غير المرتبة، والتي هي نفس الصورة السابقة ولكن خالية من الأكواد وبالتسلسل الصحيح للأرقام.	٩ (١٠٠%)	
٢٦.	قيام الطالبات بالضغط على Submit and Finish بعد الإنتهاء من الحل.	٨ (٨٨,٩%)	١ (١١,١%)
٢٧.	ظهور التغذية الراجعة للطالبات بعد كتابة الترييب الصحيح للمشكلة، والتي تحتوى على الإجابة الصحيحة لحل اللغز، وعدد الحركات التي نجحت في اتمامها، ووضع الكود في المكان الصحيح بالفعل.	٩ (١٠٠%)	
رابعًا:	**مرحلة التدريب على بناء المشكلة وتركيبها الصحيح من خلال نمط المصفوفة (Matrix) في بيئة التعلم الإلكترونية:**		
٢٨.	دخول الطالبات إلى القسم الثاني من الأقسام الثلاثة، والخاص بالتدريب على بنى وتراكيب حل المشكلة البرمجية وبدائلها المتاحة لهم في هذا الوقت تزامنًا مع ما تم عرضه في المحاضرة التقليدية من مفاهيم برمجية وأساسيات اللغة.	٩ (١٠٠%)	
٢٩.	دخول الطالبات على انشطة القسم المفتوحة بالترتيب الموجود والمنظم من السهل إلى الصعب.	٩ (١٠٠%)	
٣٠.	عقب الدخول تجد الطالبة نفسها أمام مصفوفة Matrix، تم تصميمها على شكل جدول مكون من عمودين، الأول به المثال المحلول الذي قام المعلم بعرضه للمشكلة في البيئة التقليدية وحلها وذلك لمراجعته ودراسته ببطء وتمعن من قبل الطالب، والعمود الأخر به مشكلة مكافئة للمشكلة لأولي وبها خطوة أو خطوتان ناقصة.	٩ (١٠٠%)	
٣١.	قيام الطالبات بدراسة المثال الأول والمحلول أمامهم، ثم الإنتقال إلى المشكلة المقابلة لها في العمود الثاني، وقرائتها جيدًا ثم التمييز بين المعطى والمطلوب، ودراسة خطوات الحل الموجودة.	٩ (١٠٠%)	
٣٢.	قيام الطالبات بإكمال الخطوات الناقصة للمشكلة المعروضة من خلال اختيارها للإجابة الصحيحة من بين عدة إجابات، وذلك للتأكد من فهمها وبنائه الصحيح للمعرفة البرمجية.	٩ (١٠٠%)	
٣٣.	قيام الطالبات بالضغط على Submit and Finish بعد الإنتهاء من الحل.	٨ (٨٨,٩%)	١ (١١,١%)
٣٤.	عرض التغذية الراجعة التي تخبرهم بمدى صحة حلهم.	٩ (١٠٠%)	
٣٥.	تنفيذ الحل المشكلة الثانية على برنامج Dev C++.	٩ (١٠٠%)	
٣٦.	صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية.	٨ (٨٨,٩%)	١ (١١,١%)
٣٧.	الإنتقال للتدريب على مشكلات جديدة من خلال مصفوفات أخرى، وبعد الإنتهاء من التدريب، يقوم الطالب بالإنتقال للقسم الثالث.	٩ (١٠٠%)	
٣٨.	أخرى:		

خامسًا:	**مرحلة التقويم من خلال نمط الإكمال (Completion) في بيئة التعلم الالكتروني:**			
٣٩.	قيام الطالبات بالإنتقال إلى القسم الثالث، وفتحه.	٩ (١٠٠%)		
٤٠.	قيام الطالبات بفتح صفحة التقويم الأول.	٩ (١٠٠%)		
٤١.	بعد قيام الطالبات بالدخول إلى الصفحة ، تعرض عليها مشكلة برمجية عليها أن تقرأها بشكل جيد.	٩ (١٠٠%)		
٤٢.	تجد الطالبة خطوات الحل أسفل المشكلة تنقصها نقاط على الطالبة أن تكملها لكي تجتاز هذا التقويم وتنتقل لغيره.	٩ (١٠٠%)		
٤٣.	قيام الطالبة بإكمال الخطوات الناقصة من خلال كتابتها لحل المشكلة مرة أخرى كاملة في محرر نص خاص بذلك.	٩ (١٠٠%)		
٤٤.	تضغط الطالبة على مفتاح Submit and Finish بعد إنهاء كتابة الحل.	٨ (٨٨,٩%)	١ (١١,١%)	
٤٥.	عرض التغذية الراجعة على الطالبة التي تخبرها بمدى صحة حلها.	٩ (١٠٠%)		
٤٦.	تنفيذ حل المشكلة على برنامج Dev C++.	٩ (١٠٠%)		
٤٧.	صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية.	٨ (٨٨,٩%)	١ (١١,١%)	
٤٨.	تأخذ الطالبة لقطة شاشة للحل الخاص بكل مشكلة تقويم.	١ (١١,١%)	٨ (٨٨,٩%)	
٤٩.	تقوم الطالبة بإرسال لقطة الشاشة إلى ايميل المعلم كدليل على قيامه بهذه الخطوة.	١ (١١,١%)	٨ (٨٨,٩%)	
٥٠.	الإنتقال لتقويم المشكلة الجديدة التي ستفتح عقب حل مشكلة التقويم السابقة.	٩ (١٠٠%)		
٥١.	الرجوع مرة أخرى إلى بيئة التعلم التقليدية لدراسة مفاهيم برمجية جديدة حتى ينتهي التعلم المطلوب ويبدأ تعلم جديد، وهكذا إلى نهاية المقرر.	٩ (١٠٠%)		
٥٢.	انتقال الطالبة لأداء الاختبار البعدي للمقرر عقب الإنتهاء من كل المشكلات المطلوبة للتقويم في المقرر.	٩ (١٠٠%)		
٥٣.	أخرى:			

من جدول (١) تم حساب النسبة العامة التي أبداها السادة المحكين لمدى الإتفاق على الإستراتيجية المقترحة بلغت (٩٥,٥٢)، كما بلغت نسب الإتفاق علي المراحل الرئيسية للإستراتيجية كالتالي:

(١) المرحلة الأولى = ٩٩%

(٢) المرحلة الثانية =١٠٠%

(٣) المرحلة الثالثة = ٩٦,٦٧%

(٣) المرحلة الرابعة = ٩٧,٧٨%

(٥) المرحلة الخامسة = ٨٤,١٣%

ملخص ملاحظات وتعليقات المحكمين المشاركين:

١. اتفق سائر المحكمون على أهمية مرحلة الإعداد والتهيئة .

٢. اتفق سائر المحكمون على كافة إجراءات مرحلة الإعداد والتهيئة ، فيما عدا الإجراء رقم (٧) فقد أشار أحد المحكمين إلى دمج هذا الإجراء مع الإجراء رقم (٦) مع تعديل الصياغة ، وجاءت نسبة الإتفاق

على هذه المرحلة وأهميتها (٩٩%).

٣. اتفق سائر المحكمون على أهمية مرحلة تكوين المفاهيم في بيئة التعلم التقليدية

٤. اتفق سائر المحكمون على كافة إجراءات المرحلة، وجاءت نسبة الإتفاق على هذه المرحلة وأهميتها (١٠٠%).

٥. اتفق سائر المحكون على أهمية المرحلة الثالثة والرابعة، وأشار اثنين من المحكمين إلى إمكانية دمج المرحلة الثالثة مع الرابعة أي مرحلة التدريب على تكوين المفاهيم باستخدام نمط الألغاز، والتدريب على بني وتراكيب حل المشكلة باستخدام نمط المصفوفة، تحت بند التدريب على تكوين المفاهيم وبني وتراكيب المشكلة داخل بيئة التعلم الإلكتروني.

٦. أشار المحكمين إلى أهمية إجراءات المرحلة الثالثة والرابعة ، مع التوصية بالتالي:

٧. أشار اثنين من المحكمين بنقل وإعادة ترتيب الإجراء رقم (١٨) إلى المرحلة الثانية.

٨. أشار اثنين من المحكين بدمج الإجراءات وفقًا للتقسيم السابق الخاص بدمج المرحلتين، وذلك لتكون إجراءات المرحلتين في مرحلة واحدة.

٩. أشار واحد من المحكمين إلى إعادة صياغة الإجراءات المرتبطة بإنهاء الحل الخاص بالمشكلة وغلق النمط بحيث يتم حذف كلمة Submit and Finish وجعلها عامة فقد يتغير المفتاح بتغير البيئة والتصميم، وذلك في الإجراءات رقم (٢٦، ٣٣).

١٠. أشار واحد من الحكام إلى دمج الإجراءات (٣٥، ٣٦) في نقطة واحدة.

١١. وجاءت نسبة الإتفاق على المرحلة الثالثة (٩٦,٦٧%)، ونسبة الإتفاق على هذه المرحلة وأهيتها (٩٧,٧٨%).

١٢. اتفق سائر المحكمون على أهمية مرحلة التقويم من خلال نمط الإكمال.

١٣. أشار المحكمين إلى أهمية الإجراءات، مع التوصية بالتالي:

١٤. أشار واحد من المحكمين إلى إعادة صياغة الإجراءات المرتبطة بإنهاء الحل الخاص بالمشكلة وغلق النمط بحيث يتم حذف كلمة Submit and Finish وجعلها عامة فقد يتغير المفتاح بتغير البيئة والتصميم، وذلك في الإجراء رقم (٤٤).

١٥. أشار واحد من الحكام إلى دمج الإجراءات (٤٦، ٤٧) في نقطة واحدة.

١٦. اتفق ثمانية من الحكام عدم أهمية الإجراءات رقم (٤٨، ٤٩) وأسار واحد إلى أهميتهم إلى حد ما.

١٧. وجاءت نسبة أهمية هذه المرحلة (٨٤,١٣%).

الصورة النهائية لاستراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج:

وبعد المراجعة لهذه التعديلات، أصبحت الصورة النهائية جاهزة، كما هو موضح بشكل (٨) الذي يشتمل على الصورة النهائية للإستراتيجية المقترحة حيث تشتمل على (5) مراحل رئيسية، (47) سبعة وأربعون خطوة فرعية، استنادًا علي ما جاء في جدول **(١).**

شكل (٨) الصورة النهائية للإستراتيجية المقترحة

م	مراحل الإستراتيجية وإجراءاتها
أولًا:	**مرحلة الإعداد والتهيئة في بيئة التعلم التقليدي:**
	عقد جلسة عامة مع الطالبات لتهيئتهم وإثارة دافعيتهم لبدء التعلم الجديد. تعريف الطالبات بالأهداف التعليمية للمقرر، والمطلوبة منهم في نهاية التعلم. تجميع بيانات الطالبات المطلوبة لتسجيل دخولهم للبيئة (الاسم، الإيميل، رقم التليفون). توضيح الإستراتيجية التي سيتم بها تدريس المقرر، وطريقة الدمج بين البيئتين التقليدية والإلكترونية بشكل عام ومعلن للطالبات. تحميل برنامج Dev C++ ؛ لتنفيذ برامج لغتي C, C++ على أجهزة الطالبات. تدريب الطالبات على تسجيل الدخول إلى الموقع الإلكتروني، والتعامل مع مكونات واجهة التفاعل، وتعليمات السير في الموديولات التعليمية، حتى تألفه الطالبات، ويقبلوا عليه. تدريب الطالبات على كيفية التعامل مع أنماط عرض المشكلات المستخدمة في البحث الحالي، والتنقل بينها. إرسال بيانات تسجيل كل طالب على الإيميل الخاص به (Username, Password, Site link) في موعد محدد ومعلوم للطالبات. تسجيل الطالبات الدخول إلى صفحة الموقع الالكتروني. قراءة مقدمة المقرر والتعليمات بدقة، وفهمها جيدًا قبل الدخول للموديول الأول من موديولات المقرر. أداء الطالبات للاختبار القبلي للمقرر، وحل جميع بنود هذا الاختبار، للوقوف على مدى حاجتهم لدراسة المقرر، وذلك من خلال مقارنة درجات كل طالبة بالدرجة المحكية (٩٠%) من درجة هذا الاختبار.
ثانيًا:	**مرحلة عرض تكوين المفاهيم البرمجية والقواعد البرمجية في البيئة التقليدية:**
	قيام المعلم بتوضيح الهدف المراد تدريسه إلى الطالبات. شرح المفهوم البرمجي الذي يتناوله الهدف، وتوضيح كافة جوانبه المعرفية (تعريف، خصائص، تكوين الجملة البرمجية، ...). عرض مشكلة محلولة على الطالبات لتعلم بنى وتراكيب حل المشكلة البرمجية. شرح طريقة استخدام برنامج Dev C++ لتنفيذ المشكلات البرمجية من خلاله، ثم فحصها وصيانتها.
ثالثًا:	**مرحلة التدريب على تكوين المفاهيم والقواعد البرمجية من خلال نمط الألغاز (Puzzle) في بيئة التعلم الإلكترونية:**
	توجيه الطالبات إلى بيئة التعلم الالكترونية للتدريب على تكوين المفاهيم التي تم شرحها، وبعض المشكلات التي تنمي تعلم بنى وتراكيب حل المشكلات البرمجية لديهم، ومن ثم تقييم ما تم تعلمه والتدريب عليه. تسجيل الطالبات دخولهم مرة أخرى إلى صفحة الموقع الالكتروني. الدخول إلى الموديول الأول، وقراءة الطالبات لصفحات المقدمة، وتعليمات السير في الموديول، والأهداف التعليمية وعناصر المحتوى، مع ملاحظة أن الموديول سيفتح فقط في حالة أداء الأختبار القبلي. دخول الطالبات إلى القسم الأول من الأقسام الثلاثة، والخاص بالتدريب على تكوين مفاهيم الجمل البرمجية المتاحة لهم في هذا الوقت تزامنًا مع ما تم شرحه في المحاضرة التقليدية من مفاهيم برمجية. دخول الطالبات على المفهوم المراد التدرب عليه. عرض المشكلة على الطالبات، ثم تقوم الطالبة بقرائتها جيدًا، بحيث يميز بين المعطى والمطلوب. تقديم حل المشكلة في شكل لغز Puzzle (صورة) تم تقسيمها إلى عدة أجزاء مرقمة وعلى كل جزء يوجد قطعة من الكود المطلوب لحل المشكلة البرمجية. قيام الطالبات بترتيب أجزاء الكود من خلال قيامهم بسحب وإلقاء الكود (Drag and Drop) على خلفية الصورة التي تجدها الطالبات أسفل صورة اللغز غير المرتبة، والتي هي نفس الصورة السابقة ولكن خالية من الأكواد وبالتسلسل الصحيح للأرقام. قيام الطالبات بالضغط على Submit and Finish بعد الإنتهاء من الحل. ظهور التغذية الراجعة للطالبات بعد كتابة الترييب الصحيح للمشكلة، والتي تحتوى على الإجابة الصحيحة لحل اللغز، وعدد الحركات التي نجحت في اتمامها، ووضع الكود في المكان الصحيح بالفعل.
رابعًا:	**مرحلة التدريب على بناء المشكلة وتركيبها الصحيح من خلال نمط المصفوفة (Matrix) في بيئة التعلم الإلكترونية:**
	دخول الطالبات إلى القسم الثاني من الأقسام الثلاثة، والخاص بالتدريب على بنى وتراكيب حل المشكلة البرمجية وبدائلها المتاحة لهم في هذا الوقت تزامنًا مع ما تم عرضه في المحاضرة التقليدية من مفاهيم برمجية وأساسيات اللغة. دخول الطالبات على انشطة القسم المفتوحة بالترتيب الموجود والمنظم من السهل إلى الصعب. عقب الدخول تجد الطالبة نفسها أمام مصفوفة Matrix، تم تصميمها على شكل جدول مكون من عمودين، الأول به المثال المحلول الذي قام المعلم بعرضه للمشكلة في البيئة التقليدية وحلها وذلك لمراجعته ودراسته ببطء وتمعن من قبل الطالب، والعمود الأخر به مشكلة مكافئة للمشكلة لأولي وبها خطوة أو خطوتان ناقصة.

	قيام الطالبات بدراسة المثال الأول والمحلول أمامهم، ثم الإنتقال إلى المشكلة المقابلة لها في العمود الثاني، وقرائتها جيدًا ثم التمييز بين المعطى والمطلوب، ودراسة خطوات الحل الموجودة. قيام الطالبات بإكمال الخطوات الناقصة للمشكلة المعروضة من خلال اختيارها للإجابة الصحيحة من بين عدة إجابات، وذلك للتأكد من فهمها وبنائه الصحيح للمعرفة البرمجية. قيام الطالبات بالضغط على Submit and Finish بعد الإنتهاء من الحل. عرض التغذية الراجعة التي تخبرهم بمدى صحة حلهم. تنفيذ الحل المشكلة الثانية على برنامج Dev C++. صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية. الإنتقال للتدريب على مشكلات جديدة من خلال مصفوفات أخرى، وبعد الإنتهاء من التدريب، يقوم الطالب بالإنتقال للقسم الثالث.
خامسًا:	**مرحلة التقويم من خلال نمط الإكمال (Completion) في بيئة التعلم الالكتروني:**
	قيام الطالبات بالإنتقال إلى القسم الثالث، وفتحه. قيام الطالبات بفتح صفحة التقويم البنائي الأول. بعد دخول الطالبة إلى الصفحة ، تعرض عليها مشكلة برمجية عليها أن تقرأها بشكل جيد. تجد الطالبة خطوات الحل أسفل المشكلة تنقصها نقاط على الطالبة أن تكملها لكي تجتاز هذا التقويم وتنتقل لغيره. قيام الطالبة بإكمال الخطوات الناقصة من خلال كتابتها لحل المشكلة مرة أخرى كاملة في محرر نص خاص بذلك. تضغط الطالبة على مفتاح Submit and Finish بعد إنهاء كتابة الحل. عرض التغذية الراجعة على الطالبة التي تخبرها بمدى صحة حلها. تنفيذ حل المشكلة على برنامج Dev C++. صيانة البرنامج واستخدامه للتأكد من عدم وجود أخطاء كتابية. الإنتقال لتقويم المشكلة الجديدة التي ستفتح عقب حل مشكلة التقويم السابقة. الرجوع مرة أخرى إلى بيئة التعلم التقليدية لدراســـة مفاهيم برمجية جديدة حتى ينتهي التعلم المطلوب ويبدأ تعلم جديد، وهكذا إلى نهاية المقرر. انتقال الطالبة لأداء الاختبار البعدي للمقرر عقب الإنتهاء من كل المشكلات المطلوبة للتقويم في المقرر.

وبالتالي يكون قد تم التحقق من صحة الفرض البحثي الأول بالإطار النظري في وضع أسس الأستراتيجية، والفرض الثاني بأنه أمكن وضع استراتيجية لتكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج لتنمية الكفايات البرمجية.

خطة تطبيق هذه الإستراتيجية في بيئة التعلم المدمج:

تطبيق الإستراتيجية في بيئة التعلم المدمج بشكل صحيح يعتمد على تطوير بيئة التعلم المدمج بشكل جيد ويتم ذلك من خلال تطبيق نموذج الجزار (Elgazzar, ٢٠١٤) للتصميم التعليمي، فهو مناسب لتطوير بيئات التعلم، ويلائم تطبيق استراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج لتنمية الكفايات البرمجية كالتالي:

أولًا: مرحلة الدراسة والتحليل: وفيها يتم اشتقاق أو تبني معايير التصميم التعليمي، وتحديد خصائص المتعلمين، والحاجات التعليمية، وتحليل المحتوى وتحديد كفايات البرمجة بلغة (C++) ، ومصادر التعلم الالكترونية المتاحة والمعوقات والمحددات.

ثانيًا: مرحلة التصميم: وفيها يتم صياغة الأهداف التعليمية وتحديد عناصر المحتوى التعليمى وتصميم أدوات التقويم، والاختبارات، وتصميم خبرات وانشطة التعلم ، وكذلك اختيار بدائل عناصر الوسائط المتعددة للخبرات والمصادر والانشطة، وتصميم المحتوى اوالسيناريوهات للوسائط التى تم اختيارها، وتصميم اساليب الابحار والتحكم التعليمى وواجهة المتعلم، واستخدام استراتيجية تكامل أنماط عرض المشكلات البرمجية في بيئة التعلم المدمج لتنمية الكفايات البرمجية بمراحلها الخمس بما تتضمنه من التعلم (F2F) والتعلم الإلكتروني، واختيار أو تصميم ادوات التواصل سواء كانت متزامنة او غير متزامنة وتصميم نظم تسجيل المتعلمين وإدراتهم وتجميعهم ونظم الدعم بالبيئة، وكذلك تصميم بيانات ومعلومات

والمخطط الشكلى لعناصر البيئة والابحار، وتصميم المعلومات الاساسية للبيئة مثل العنوان ،وشعار الصفحة.

ثالثًا: مرحله الانتاج والإنشاء: وفيها يتم الحصول على الوسائط والمصادر، وإنتاجها، وترقيمها، وإنتاج معلومات وعناصر المخطط الشكلى لبيئة التعلم الالكترونية، ورفع وتحميل أوعمل الروابط لعناصر البيئة وروابط الويب، وإنشاء (الموديولات) والدروس أادوات التواصل وتسجيل المتعلمين؛ وتشطيب النموذج الأولى للبيئة، وعمل المراجعات استعدادًا للتقويم البنائى .

رابعًا: مرحلة التقويم: وفيها يتم إجراء التقويم البنائى للبيئة على أفراد من المتعلمين، وعمل التحكيم للتاكد من مطابقتها لمعايير التصميم، ومن ثم تطبيق التقويم النهائى والانتهاء من التطوير التعليمى.

خامسًا: مرحلة النشر والاستخدام: ويتم فيها الاستخدام الميدانى وتطبيق بيئة تعلم الالكترونى، وتكون المراقبه فيه مستمرة، لتوفير الدعم والصيانه للبيئة.

توصيات البحث:

في ضوء نتائج البحث، قام الباحثون بعمل قائمة التوصيات التالية:

١. الإستفادة من الإستراتيجية المقترحة في تدريس مقررات لغات البرمجة الحديثة و لغة C++ ، وتطبيقها في مقررات برمجية أخري.
٢. تبني المصممين والمطورين لبيئات التعلم المدمج لتعليم البرمجة وتنمية كفايات حل المشكلات البرمجية لهذه الإستراتيجية لتكامل أنماط العرض بهذه البيئات.
٣. الإستفادة من الخطة المقترحة لتطبيق هذه الإستراتيجية للتطبيق الصحيح لها لتعلم البرمجة وتنمية كفايات حل المشكلة البرمجية.
٤. تطبيق نموذج الجزار (Elgazzar, 2014) لتطوير بيئة التعلم المدمج بشكل عام وتطوير بيئة التعلم المدمج القائمة على تكامل أنماط عرض المشكلات البرمجية لهذا البحث.

مقترحات البحث:

في ضوء نتائج البحث، يقترح الباحثون إجراء المزيد من البحوث والدراسات في الموضوعات البحثية التالية:

١. إجراء بحوث تطويري للكشف عن أثر تطبيق استراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات البرمجة لحل المشكلات البرمجية للغات الحديثة لطلاب تكنولوجيا التعليم.
٢. إجراء بحوث تطويرية للكشف عن أثر تطبيق استراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات المعلوماتية والبرمجة لمقررات أقسام الحاسبات ونظم المعلومات.
٣. إجراء بحث تطويري للكشف عن أثر تطبيق استراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم المدمج لتنمية كفايات المعلوماتية والبرمجة بالتعليم قبل الجامعي.
٤. إجراء بحوث تطويرية للكشف عن أثر تطبيق استراتيجية تكامل أنماط عرض المشكلات البرمجية ببيئة للتعلم الذكية والتكيفية لتنمية كفايات المعلوماتية والبرمجة لمقررات أقسام الحاسبات ونظم المعلومات.

مراجع البحث

أولًا: المراجع العربية:

إبراهيم أحمد السيد عطية (٢٠١٠). أثر التفاعل بين إستراتيجية حل المشكلات مفتوحة النهاية و السعة العقلية على الحلول الإبتكارية لمشكلات البرمجة التعليمية لدى طلاب الدبلوم المهنية. *دراسات تربوية ونفسية: جامعة الزقازيق - كلية التربية* ، مسترجع من http://search.mandumah.com/Record/11118

جمال مصطفي عبدالرحمن الشرقاوي. (٢٠١٢). تصميم إستراتيجية مقترحة لتطوير التعليم المدمج في ضوء الشبكات الاجتماعية لتنمية مهارات تصميم ونشر المقرر الإلكتروني لطلاب كليات التربية. *مجلة كلية التربية*، جامعة المنصورة.

حسن حسين زيتون. (٢٠٠٥): رؤية جديدة في التعليم : التعلم الإلكتروني: المفهوم . القضايا . التطبيق .التقييم، الرياض: الدار الصولتية للتربية.

أحمد أبو العلا بهنساوى، أمل عبدالفتاح سويدان، حسن حسيني جامع، شوقي محمد محمود، منى محمد الجزار. (٢٠١٢): فعالية التدريس الخصوصي بالكمبيوتر في تنمية مهارات حل المشكلات البرمجية لدى طلاب كلية التربية النوعية. *المجلة العربية للتربية العلمية*، ١(١)، ٩٨-١٣٢.

حسن عبد الله النجار. (٢٠٠٩). أثر استراتيجية التعلم التوليفي في تنمية مهارات تصميم مواقع الويب التعليمية لدى طلاب التكنولوجيا بجامعة الأقصى واتجاههم نحوه. *مجلة تكنولوجيا التعليم*، *١٨* (٣)، ١٤٣-١٧٤.

الشحات سعد محمد عتمان.(٢٠١٢): تصميم إستراتيجية عبر الويب لاستخدام المدونات التعليمية فى تنمية مهارات كتابة خطة البحث العلمي في تكنولوجيا التعليم لطلاب الماجستير. *مجلة تكنولوجيا التعليم*، (٢٢)، ٢٢٣- ٢٧٧.

عبد اللطيف بن الصفي الجزار. (١٩٩٥). دراسة استكشافية لاستخدام طالبات كلية التربية وجامعة الإمارات العربية المتحدة لنموذج تطوير المنظومات التعليمية في تكنولوجيا التعليم. *مجلة تكنولوجيا التعليم: سلسلة بحوث ودراسات . المجلد الخامس* . الكتاب الرابع . القاهرة: الجمعية المصرية لتكنولوجيا التعليم .

عبد اللطيف بن الصفي الجزار. (٢٠٠٢). فعالية استخدام التعليم بمساعدة الكمبيوتر متعدد الوسائط في اكتساب بعض مستويات تعلم المفاهيم العلمية وفق نموذج "فراير" لتقوم المفاهيم. *مجلة التربية*، جامعة الأزهر، (١٠٥) ٢٩- ٨٣.

عبد اللطيف الصفي الجزار. (٢٠١٧). منهج تحليل المحتوي في البحوث التطويرية ومنهج البحث التطويري في تكنولوجيا التعليم. *عرض تقديمي PPT ضمن محاضرات مقرر قاعة البحث في تكنولوجيا التعليم، الدبلوم الخاصة (تكنولوجيا تعليم)، كلية البنات بجامعة عين شمس.*

علي محمد علي الزعبي. (٢٠١٤) . أثر إستراتيجية تدريسية قائمة على حل المشكلات فى تنمية مهارات التفكير الإبداعى الرياضى: لدى طلبة معلم صف. المجلة الأردنية في العلوم التربوية: جامعة اليرموك .مسترجع من http://search.mandumah.com/Record/626203

محمد عطية خميس. (٢٠٠٣) . منتوجات تكنولوجيا التعليم. القاهرة، مكتبة دار الكلمة.

محمد عطية خميس. (٢٠٠٣، ب). عمليات تكنولوجيا التعليم. القاهرة، مكتبة دار الكلمة.

وحدة الجودة، قسم تكنولوجيا التعليم والمعلومات .(٢٠٢٠). *توصيف مقرر لغات برمجة حديثة (٢)* (٢٢٣ تك). كلية البنات جامعة عين شمس.

ثانيًا: المراجع الأجنبية:

Adelson, B. (1981). Problem solving and the development of abstract categories in programming languages. *Memory & cognition*, *9*(4), 422-433.

Alajmi, F., & Alkhatib, A. A. (2015). Enhanced Teaching Model (ETM) for Teaching Programming Languages. *International Journal of Computer Applications*, *121*(20).

Alammary, A., Carbone, A., & Sheard, J. (2012, January). Implementation of a smart lab for teachers of novice programmers. In *Proceedings of the Fourteenth Australasian Computing Education Conference-Volume 123* (pp. 121-130).

Alario-Hoyos, C., Kloos, C. D., Estévez-Ayres, I., Fernández-Panadero, C., Blasco, J., Pastrana, S., & Villena-Román, J. (2016). Interactive activities: the key to learning programming with MOOCs. *Proceedings of the European Stakeholder Summit on Experiences and Best Practices in and Around MOOCs, EMOOCS*, *319*.

Albrecht, E., Gumz, F., & Grabowski, J. (2018, June). Experiences in introducing blended learning in an introductory programming course. In *Proceedings of the 3rd European Conference of Software Engineering Education* (pp. 93-101).

Alemi, M. (2010). Educational games as a vehicle to teaching vocabulary. *The Modern Journal of Applied Linguistics*, *2*(6), 425-438.

Alexander S. & Pevzner, P. (2018). Learning Algorithms through Programming and Puzzle Solving.

Allen, V. F. (1983). *Techniques in Teaching Vocabulary*. Oxford University Press, 200 Madison Ave., New York, NY 10016 (ISBN 0-19-434130-5, $4.95)..

Alraghaib, H. K., Elgazzar, Abdellatif E. & Nouby, Ahmed (2015). Sequential Design vs. Integrated Design of Blended Learning of Informatics Subject Matter: Is There Any Effectiveness in Developing Cognitive Achievement and Achievement Motivation among Kuwaiti Female Secondary School Students. *Open Journal of Social Sciences*, *3*(02).

Alturki, R. A. (2016). Measuring and improving student performance in an introductory programming course. *Informatics in Education-An International Journal*, *15*(2), 183- 204.

Ambrósio, A. P., Costa, F. M., Almeida, L., Franco, A., & Macedo, J. (2011, October). Identifying cognitive abilities to improve CS1 outcome. In *2011 Frontiers in Education Conference (FIE)* (pp. F3G-1). IEEE.

Anderson, J. R., Farrell, R., & Sauers, R. (1984). Learning to program in LISP. *Cognitive Science*, *8*(2), 87-129.

Atef, H., & Medhat, M. (2015). Blended learning possibilities in enhancing education, training and development in developing countries: A case study in graphic design courses. *TEM Journal*, *4*(4), 358.

Atkinson, R. K., Levin, J. R., Kiewra, K. A., Meyers, T., Kim, S. I., Atkinson, L. A., & Hwang, Y. (1999). Matrix and mnemonic text-processing adjuncts: Comparing and combining their components. *Journal of Educational Psychology*, *91*(2), 342.–357.

Atkinson, R. K., Merrill-Lusk, M. M., and Bietzel, B. (2007). *Learning from book-based examples: Exploring the impact of combining fading with prompts and matrices*. Paper presented at the biennial meeting of the European Association for Research on Learning and Instruction, Budapest, Hungary.

Baars, M., Visser, S., Van Gog, T., de Bruin, A., & Paas, F. (2013). Completion of partially worked-out examples as a generation strategy for improving monitoring accuracy. *Contemporary Educational Psychology*, *38*(4), 395-406.

Bera, S. J., & Robinson, D. H. (2004). Exploring the Boundary Conditions of the Delay Hypothesis with Adjunct Displays. *Journal of Educational Psychology*, *96*(2), 381–388.

Bornat, R., & Dehnadi, S. (2008, January). Mental models, consistency and programming aptitude. In *Proceedings of the tenth conference on Australasian computing education-Volume 78* (pp. 53-61).

Çakıroğlu, Ü. (2014). Analyzing the effect of learning styles and study habits of distance learners on learning performances: A case of an introductory programming course. *International Review of Research in Open and Distributed Learning*, *15*(4), 161-185.

Caspersen, M. E. (2007). *Educating novices in the skills of programming* (Doctoral dissertation, Department of Computer Science).

Chen, Y. C. (2013). *Learning protein structure with peers in an AR-enhanced learning environment* (Doctoral dissertation).

Dale, N. B., & Weems, C. (2008). *Programming and problem solving with Java*. Jones & Bartlett Learning.

Deimel, L. E., & Moffat, D. V. (1982). A more analytical approach to teaching the introductory programming course. In *Proceedings of the National Educational Computing Conference* (pp. 114-119).

Dong, T., Dontcheva, M., Joseph, D., Karahalios, K., Newman, M., & Ackerman, M. (2012, May). Discovery-based games for learning software. In *Proceedings of the SIGCHI conference on human factors in computing systems* (pp. 2083-2086).

Dunican, E. (2002). Making the analogy: Alternative delivery techniques for first year programming courses.

Ferrari, A., Poggi, A., & Tomaiuolo, M. (2016). Object oriented puzzle programming. *Mondo Digitale*, *15*, 64.

Ford, M., & Venema, S. (2010). Assessing the success of an introductory programming course. *Journal of Information Technology Education: Research*, *9*(1), 133-145.

Fullerton, T. (2008). Game Design Workshop: A Playcentric Approach to Creating Innovative Games.

Gegg-Harrison, T. S. (1991). Learning Prolog in a schema-based environment. *Instructional Science*, *20*(2), 173-192.

Gerjets, P., Scheiter, K., & Schuh, J. (2008). Information comparisons in example-based hypermedia environments: Supporting learners with processing prompts and an interactive comparison tool. *Educational Technology Research and Development*, *56*(1), 73-92.

Gilmore, D. J. (1990). Expert programming knowledge: a strategic approach. In *Psychology of programming* (pp. 223-234). Academic Press.

Gomes, A., & Mendes, A. (2014, October). A teacher's view about introductory programming teaching and learning: Difficulties, strategies and motivations. In *2014 IEEE Frontiers in Education Conference (FIE) Proceedings* (pp. 1-8). IEEE.

Gomes, A., & Mendes, A. J. (2007). Problem Solving in Programming. In *PPIG* (p. 18).

Gomes, A., & Mendes, A. J. (2007, September). Learning to program-difficulties and solutions. In *International Conference on Engineering Education–ICEE* (Vol. 2007).

Griffin, T. D., Wiley, J., & Thiede, K. W. (2008). Individual differences, rereading, and self- explanation: Concurrent processing and cue validity as constraints on metacomprehension accuracy. *Memory & Cognition*, *36*(1), 93-103.

Hadjerrouit, S. (2008). Towards a blended learning model for teaching and learning computer programming: A case study. *Informatics in Education-An International Journal*, *7*(2), 181-210.

Hancock-Niemic, M. A., Lin, L., Atkinson, R. K., Renkl, A., & Wittwer, J. (2016). Example-based learning: exploring the use of matrices and problem variability. *Educational Technology Research and Development*, *64*(1), 115-136.

Hartness, K. (2004). Robocode: using games to teach artificial intelligence. *Journal of Computing Sciences in Colleges*, *19*(4), 287-291.

Hearn, R. A., Demaine, E. D., & Sussman, G. J. (2006). Games, puzzles, and computation.

Howard, E. V. (2002). Can we teach introductory programming as a liberal education course? Yes, we can. In *The Proceedings of ISECON* (Vol. 19).

Ihantola, P., & Karavirta, V. (2011). Two-dimensional parson's puzzles: The concept, tools, and first observations. *Journal of Information Technology Education*, *10*(2), 119-132.

Ilahi, M., Belcadhi, L. C., & Braham, R. (2013, October). Towards a Competence Web-Based Assessment model to support lifelong learning. In *Fourth International Conference on Information and Communication Technology and Accessibility (ICTA)* (pp. 1-6). IEEE.

Ismail, M. N., Ngah, N. A., & Umar, I. N. (2010). Instructional strategy in the teaching of computer programming: a need assessment analyses. *TOJET: The Turkish Online Journal of Educational Technology*, *9*(2).

Ja'ashan, M. M. N. H. (2015). Perceptions and Attitudes towards Blended Learning for English Courses: A Case Study of Students at University of Bisha. *English Language Teaching*, *8*(9), 40-50.

Jairam, D., Kiewra, K. A., Kauffman, D. F., & Zhao, R. (2012). How to study a matrix. *Contemporary Educational Psychology*, *37*(2), 128-135.

Kauffman, D. F., & Kiewra, K. A. (2010). What makes a matrix so effective? An empirical test of the relative benefits of signaling, extraction, and localization. *Instructional Science*, *38*(6), 679-705.

Kiewra, K. A., Dennison, R. S., & Benton, S. T. (1995). *How studying text supplements affects prose processing*. Paper presented at American Educational Research Association, San Francisco, CA (April).

Kori, K., Pedaste, M., Tõnisson, E., Palts, T., Altin, H., Rantsus, R., ... & Rüütmann, T. (2015, March). First-year dropout in ICT studies. In *2015 IEEE Global Engineering Education Conference (EDUCON)* (pp. 437-445). IEEE.

Koriat, A., & Ackerman, R. (2010). Choice latency as a cue for children's subjective confidence in the correctness of their answers. *Developmental science*, *13*(3), 441-453.

Koriat, A., Ma'ayan, H., & Nussinson, R. (2006). The intricate relationships between monitoring and control in metacognition: Lessons for the cause-and-effect relation

between subjective experience and behavior. *Journal of experimental psychology: general*, *135*(1), 36.

Langan-Fox, J., Code, S., & Langfield-Smith, K. (2000). Team mental models: Techniques, methods, and analytic approaches. *Human Factors*, *42*(2), 242-271.

Langan-Fox, J., Waycott, J. L., & Albert, K. (2000). Linear and graphic advance organizers: Properties and processing. *International Journal of Cognitive Ergonomics*, *4*(1), 19- 34.

Lau, W. W., & Yuen, A. H. (2011). Modelling programming performance: Beyond the influence of learner characteristics. *Computers & Education*, *57*(1), 1202-1213.

Lee, M. J., Bahmani, F., Kwan, I., LaFerte, J., Charters, P., Horvath, A., ... & Long, S. (2014, July). Principles of a debugging-first puzzle game for computing education. In *2014 IEEE symposium on visual languages and human-centric computing (VL/HCC)* (pp. 57-64). IEEE.

Lehman, S., Bruning, R., & Horn, C. (1983). A tool for improving critical thinking in web-based instruction: Two experimental studies. *The CLASS project*.

Levitin, A. (2005, February). Analyze that: puzzles and analysis of algorithms. In *Proceedings of the 36th SIGCSE technical symposium on Computer science education* (pp. 171- 175).

Lin, Y. W., Tseng, C. L., & Chiang, P. J. (2016). The effect of blended learning in mathematics course. *EURASIA Journal of Mathematics, Science and Technology Education*, *13*(3), 741-770.

Linden, T., & Lederman, R. (2011). Creating visualizations from multimedia building blocks: A simple approach to teaching programming concepts.

Lister, R., Adams, E. S., Fitzgerald, S., Fone, W., Hamer, J., Lindholm, M., ... & Simon, B. (2004). A multi-national study of reading and tracing skills in novice programmers. *ACM SIGCSE Bulletin*, *36*(4), 119-150.

Long, J. (2007). Just For Fun: using programming games in software programming training and education. *Journal of Information Technology Education: Research*, *6*(1), 279- 290.

McCracken, M., Almstrum, V., Diaz, D., Guzdial, M., Hagan, D., Kolikant, Y. B. D. ... & Wilusz, T. (2001). A multi-national, multi-institutional study of assessment of programming skills of first-year CS students. In *working group reports from ITiCSE on Innovation and technology in computer science education* (pp. 125-180).

Merrick, K. E. (2010). An empirical evaluation of puzzle-based learning as an interest approach for teaching introductory computer science. *IEEE Transactions on Education*, *53*(4), 677-680.

Michalewicz, Z., & Michalewicz, M. (2008). *Puzzle-based learning*. Hybrid Publishers.

Mirriahi, N., Alonzo, D., & Fox, B. (2015). A blended learning framework for curriculum design and professional development. *Research in Learning Technology, 23*.

Mohorovičić, S., & Strčić, V. (2011, January). An overview of computer programming teaching methods. In *XXII Central European Conference on Information and Intelligent Systems* (pp. 1-6).

Ngo-Ye, T., & Park, S. H. (2014). Motivating business major students to learn computer programming–a case study. In *Proceedings of the Southern Association for Information Systems Conference, Macon, GA, USA March 21st–22nd*.

Oliver, R. (1993). Measuring hierarchical levels of programming knowledge. *Journal of Educational Computing Research, 9*(3), 299-312.

Oyelere, S. S., Agbo, F. J., Sanusi, I. T., Yunusa, A. A., & Sunday, K. (2019, July). Impact of Puzzle-Based Learning Technique for Programming Education in Nigeria Context. In *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)* (Vol. 2161, pp. 239- 241). IEEE.

Parsons, D., & Haden, P. (2006, January). Parson's programming puzzles: a fun and effective learning tool for first programming courses. In *Proceedings of the 8th Australasian Conference on Computing Education-Volume 52* (pp. 157-163).

Petkov, M., & Rogers, G. E. (2011). Using Gaming to Motivate Today's Technology-Dependent Students. *Journal of STEM Teacher Education, 48*(1), 7-12.

Rahman, S. S. A. (2012). *Learning programming via worked-examples: The effects of cognitive load and learning styles* (Doctoral dissertation, University of Sussex).

Redford, J. S., Thiede, K. W., Wiley, J., & Griffin, T. D. (2012). Concept mapping improves metacomprehension accuracy among 7th graders. *Learning and Instruction, 22*(4), 262-270.

Robinson, D. H., & Kiewra, K. A. (1995). Visual argument: Graphic organizers are superior to outlines in improving learning from text. *Journal of educational psychology, 87*(3), 455.

Robinson, D. H., & Schraw, G. (1994). Computational efficiency through visual argument: Do graphic organizers communicate relations in text too effectively?. *Contemporary Educational Psychology, 19*(4), 399-415.

Robinson, D. H., & Skinner, C. H. (1996). Why graphic organizers facilitate search processes: Fewer words or computationally efficient indexing?. *Contemporary Educational Psychology, 21*(2), 166-180.

Robinson, D. H., Katayama, A. D., Dubois, N. F., & Devaney, T. (1998). Interactive effects of graphic organizers and delayed review on concept application. *The Journal of Experimental Education*, *67*(1), 17-31.

Roblyer, M. D., & Doering, A. H. (2013). Integrating educational technology into teaching..[Kindle version]. *Retrieved from Amazon. com*.

Rosminah, M. D. S., & Zamzuri, A. (2019). Difficulties in learning programming: Views of students. In *Proc. 1st International Conference on Current Issues in Education* (pp. 74-78).

Rudder, A., Bernard, M., & Mohammed, S. (2007, March). *Teaching programming using visualization*. In Proceedings of the Sixth IASTED International Conference on Web-Based Education (pp. 487-492).

Schell, J. (2008). The art of game design: a book of lenses/by Jesse Schell.

Shih, Y. E. (2007). Setting the new standard with mobile computing in online learning. *The International Review of Research in Open and Distributed Learning*, *8*(2).

Sunday, K., Ocheja, P., Hussain, S., Oyelere, S., Samson, B., & Agbo, F. (2020). Analyzing Student Performance in Programming Education Using Classification Techniques. *International Journal of Emerging Technologies in Learning (iJET)*, *15*(2), 127-144.

Suzuki, A., Sato, T., & Awazu, S. (2008). Graphic display of linguistic information in English as a Foreign Language reading. *TESOL Quarterly*, *42*(4), 591-616.

Sweller, J., Ayres, P., & Kalyuga, S. (2011). Measuring cognitive load. In *Cognitive load theory* (pp. 71-85). Springer, New York, NY.

Tenenberg, J. D., Fincher, S., Blaha, K., Bouvier, D., Chen, T. Y., Chinn, D., ... & Monge, A. (2005). Students Designing Software: a Multi-National, Multi-Institutional Study. *Informatics in Education*, *4*(1), 143-162.

Thiede, K. W., Griffin, T. D., Wiley, J., & Anderson, M. C. (2010). Poor metacomprehension accuracy as a result of inappropriate cue use. *Discourse Processes*, *47*(4), 331-362.

Thomas, L., Ratcliffe, M., Woodbury, J., & Jarman, E. (2002). Learning styles and performance in the introductory programming sequence. *ACM SIGCSE Bulletin*, *34*(1), 33-37.

Topalli, D., & Cagiltay, N. E. (2018). Improving programming skills in engineering education through problem-based game projects with Scratch. *Computers & Education*, *120*, 64-74.

Tucker, C. R. (2012). *Blended learning in grades 4–12: Leveraging the power of technology to create student-centered classrooms*. Corwin Press.

Van Ments, M. (1999). *The effective use of role-play: Practical techniques for improving learning*. Kogan Page Publishers.

Van Merriënboer, J. J. (1990). Strategies for programming instruction in high school: Program completion vs. program generation. *Journal of educational computing research, 6*(3), 265-285.

Van Merrienboer, J. J., Schuurman, J. G., De Croock, M. B. M., & Paas, F. G. W. C. (2002). Redirecting learners' attention during training: Effects on cognitive load, transfer test performance and training efficiency. *Learning and instruction, 12*(1), 11-37.

Vekiri, I. (2002). What is the value of graphical displays in learning? *Educational psychology review*, *14*(3), 261-312.

Designing a strategy to integrate Programming Problems' Presentation Patterns in a Blended learning environment for developing Female Instructional Technology Students' Programming Competencies of Modern Programming Languages

nagwa azzam ahmed,

Abdel-Latif El-Safy Ali El-Gazzar

samah mohamed saber

Abstract

This research aims at designing a new strategy to integrate Programming Problems' Presentation Patterns to develop students' competencies of Modern Programming Languges (C++) in a Blended learning environment. The suggested strategy integrates three presentations patterns of the programming problem: Puzzle pattern, Matrix pattern, and Completion pattern. Authors used the descriptive analytical research method. The literature and related educational technology research have been reviewed and analyzed to formulate the theoretical forundations of the suggested strategy for intended C++ competencies. In light of theese theoretical foundations, Authors developed the initial prototype of the suggested strategy, which was consisted of five stages: preparation and orientation stage (F2F), concept and programming rules presentation and formation stage (F2F), practicing concept formation and programming rules through puzzles pattern stage (eLearning), practicing programming problems formation and structure correction through matrix pattern stage (eLearning), and finally evaluation stage through completion pattern (eLearning). This initial prototype was reviewed and refereed by (9) participant referees from the faculty of Instructional Technology specialization. Suggested modifications were carried out on the initial prototype, which led to the final form of the strategy for integrating patterns of presentation of programming for developing programming competencies, which consists of (5) stages and (47) procedural steps. Then, the refereeing data was analyzed and presented. Research results showed an average agreement (95%) between referees on the overall new strategy. Moreover, the final form of the developed strategy, its implementation plan, figures, references, recommendations, and suggestions of future research are also included in the report.

***Keywords*:** Blended learning, strategy stages, programming problem, integration of presentation patterns, C ++ programming competencies, puzzle presentation pattern, matrix presentation pattern.