



دولة ليبيا
وزارة التربية والتعليم
مركز المناهج التعليمية والبحوث التربوية

تقنية المعلومات دليل المعلم

للسنة الثانية بمرحلة التعليم الثانوي
«للقسمين العلمي والأدبي»
الفصل الدراسي الأول

تأليف

د. عبد المجيد حسين محمد د. عمر مصطفى الصلابي

المراجعة العلمية

د. نصر الدين بشير الزعبي أ. ازدهار الوحيشي الخطابي

المراجعة اللغوية

أ. خليفة مصباح الساروي

1435 - 1436 هـ
2014 - 2015 م

جميع حقوق الطبع والنشر محفوظة
لمركز المناهج التعليمية والبحوث التربوية

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ

مُقَدِّمَةٌ

يقوم هذا الكتاب بإرشاد معلمي مادة تقنية المعلومات لمرحلة التعليم الثانوي - السنة الثانية (جميع الشعب) حول أفضل السبل لشرح وتقييم تعلم الطلاب لجميع الدروس التي وردت بالكتاب الأساسي. لقد تم أيضاً إرشاد المعلم لاستراتيجيات نقل المعلومة إلى الطالب اعتماداً على الأمثلة القياسية وصولاً لتسهيل عملية استيعاب الدروس دون لبس وتقادياً لما خلصت إليه العديد من الأبحاث حول تحديات تدريس مبادئ البرمجة للطلاب. لذلك عند تأليفنا لهذا الكتاب راعينا الاسترشاد بالتوصيات المنهجية لهذه الأبحاث عملاً على تفادي الأخطاء المنهجية الشائعة في تأليف الكتب المنهجية عموماً وكتب برمجة الحاسوب خصوصاً. في هذا السياق لقد اعتمدنا على الاستعانة بالرسوم التوضيحية في شرح مفاهيم البرمجة.

لعل بعض الإخوة المعلمين سيلاحظون عدم تطرق الكتاب إلى بعض الأوامر والتراكيب البرمجية والمكونات التفصيلية لبيئة لغة الفيچوال بيسك. هذا كان أمراً متعمداً وهو أسلوب جديد في تعليم البرمجة للمبتدئين يُدعى أسلوب السقالة*. لقد استرشدنا كذلك بمبدأ السقالة (scaffolding) كأحد أحدث أساليب تدريس لغات برمجة الحاسوب. يعتمد هذا المبدأ على عدم إغراق الطالب بتفاصيل عديدة حول أوامر لغات البرمجة وصيغها المتعددة. بينما يتم الاعتماد على الحد الأدنى من جمل وتراكيب لغة البرمجة والتي تمكن الطالب من استيعاب عملية تصميم وبناء برامج عملية بسيطة. التعليل المنطقي لهذا الأسلوب يتمثل في أن مهارات الأداء اللغوي للإنسان لا تستوجب كونه ملماً بجميع مصطلحات وقواعد التعابير البلاغية للغة ما كالعربية مثلاً. لذلك شكلت مواضيع الكتاب المبادئ الأساسية لبرمجة النظم المحوسبة. التراكيب والصيغ والأوامر البرمجية الأكثر تفصيلاً يتم اكتسابها من خلال مقررات السنوات الدراسية القادمة، بالإضافة إلى ما سيكتسبه الطالب أثناء الممارسة الشخصية.

لقد قُسمت مواضيع الكتب وفقاً لتدرج مدروس واعتمدنا استهلال كل فصل بإيضاح نواتج التعلم متمثلة في المكتسبات مهارية والمعرفية المستهدفة. كل منها يعتبر لبنة لاستيعاب مفاهيم ومهارات أخرى سيتعلمها الطالب في الدروس التالية. كما أن تحديد نواتج التعلم يقوم أيضاً بمساعدة معلم المادة على معرفة ما هو مستهدف تحقيقه ومن ثم يمكنه تحضير الدرس وتحضير وسائل التقويم وفقاً لذلك. لذا يجب على المعلم التركيز على شرح كل درس بما يمكن تحقيق المكتسبات المعرفية لكل درس وتجنب التطرق إلى تفاصيل تتجاوز ذلك. لعل البعض يعتقد أن ثمة تفاصيل أخرى غير مدرجة بالكتاب ويجب التطرق لها، وإجابتنا على ذلك أن منهجية تأليف الكتاب قامت على

❖ تم اشتقاق هذا المفهوم من أسلوب بناء المباني اعتماداً على السقالات المثبتة بحوائط المبنى.

توصيات جمعية علوم الحاسوب العالمية (ACM). إضافة إلى توصيات أخرى بالخصوص ساهم في إصدارها العديد من الأبحاث في مجال تعليم البرمجة للمبتدئين. لذلك نتوخى من إخوتنا المعلمين والمعلمات تحضير كل درس بما يساهم في تحقيق كل عنصر من عناصر نواتج التعلم دون زيادة.

كل فصل تم إرفاقه بحلول التمارين المرفقة بالدروس ويمكن للإخوة المعلمين اعتمادها كما هي لتقييم تحصيل الطلاب أو إجراء أي تعديلات بسيطة عليها لتوفير عدد أكثر من أسئلة التقويم. ونحن نضع هذا الكتاب بين أيدي إخوتنا المعلمين والمعلمات وكذلك مفتشي المادة، فإننا على أتم الاستعداد لاستقبال ملاحظاتهم فيما يتعلق بمحتوى هذا الكتاب وسنعمل على التعامل مع تلك الملاحظات بكل جدية آملين تلافي أي أخطاء في الطبعات القادمة.

والله وليّ التوفيق

المؤلفان

فهرس

الفصل الأول: حل المسائل

9	نواتج التعلم المستهدفة	1.1
9	المسائل	2.1
10	حل المسائل	3.1
10	صياغة خطوات الوصول إلى حل المسألة	4.1
10	دور الحاسوب في حل المسائل	5.1
10	بماذا يتفوق الإنسان على الحاسوب	6.1
11	الخوارزمية	7.1
12	حل تمارين الفصل	8.1

الفصل الثاني: خرائط التدفق

13	نواتج التعلم	1.2
13	أهمية الاستعانة بالخرائط في وصف الأشياء	2.2
13	خرائط التدفق	3.2
14	العمليات المتتابة	4.2
14	عملية اتخاذ قرار من أجل تفرع	5.2
14	العمليات المكررة في خرائط التدفق	6.2
15	حل تمارين الفصل	7.2

الفصل الثالث: أساسيات البرمجة

17	نواتج التعلم	1.3
17	مفهوم برمجة الحاسوب	2.3
18	تعريف لغة البرمجة	3.3
18	البرنامج	4.3
18	العلاقة بين الخوارزمية والبرنامج	5.3
19	لغات البرمجة	6.3
19	تخزين البيانات بذاكرة الحاسوب	7.3

20	مواقع تخزين البيانات	8.3
20	الثوابت	9.3
20	المتغيرات	10.3
20	أنواع المتغيرات	11.3
20	أولوية تنفيذ العمليات الحسابية	12.3
21	استخدام الأقواس في تنفيذ العمليات الحسابية	13.3
21	تخصيص البيانات في المتغيرات بالذاكرة	14.3
21	حل تمارين الفصل	15.3

الفصل الرابع: مدخل إلى البرمجة بلغة البيسك المرئي

23	نواتج التعلم	1.4
23	جملة تخصيص البيانات LET	2.4
24	أخطاء شائعة عند التعامل مع جمل التخصيص	3.4
25	جملة طباعة البيانات PRINT	4.4
25	حل تمارين الفصل	5.4

الفصل الخامس: تشغيل بيئة لغة بييسك المرئي وتنفيذ مثال برمجي

27	نواتج التعلم	1.5
28	أخطاء البرمجة	2.5
28	حل تمارين الفصل	3.5

الفصل السادس: التفاعل مع الحاسوب

31	نواتج التعلم	1.6
31	البرامج الخاصة والبرامج العامة	2.6
32	دالة صندوق الإدخال INPUTBOX	3.6
32	حل تمارين الفصل	4.6

الفصل السابع: أوامر برمجة التحكم المشروط

35	نواتج التعلم	1.7
35	جملة التحكم المشروط IF THEN ELSE	2.7
36	حل تمارين على جملة التحكم المشروط IF THEN ELSE	1.2.7
37	جملة الاختيار SELECT CASE	3.7
37	حل تمارين على جملة الاختيار SELECT CASE	1.3.7
39	جملة التكرار FOR NEXT	4.7

39 FOR NEXT	حل تمارين على جملة التكرار	1.4.7	
41 DO.. WHILE	جملة التكرار		5.7
42 DO.. WHILE	حل تمارين على جملة التكرار	1.5.7	
44 DO.. UNTIL	جملة التكرار		6.7
44 DO.. UNTIL	حل تمارين على جملة التكرار	1.6.7	

حل المسائل

1.1 نواتج التعلم المستهدفت

- لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ الإلمام بأهمية استخدام الحاسوب في حل المسائل.
 - ❖ الإلمام بالأسلوب الصحيح لحل المسائل من خلال فهم المسألة وتحديد متطلبات الحصول على الحل.
 - ❖ إدراك المقصود بالخوارزمية.
 - ❖ استخدام الخوارزمية في صياغة حل المسائل.

2.1 المسائل

الغرض من هذه الفقرة هو اطلاع الطالب على أهمية الحاسوب من خلال مساعدته للإنسان في حل المسائل بأنواعها. من المهم أن يدرك الطالب ما المقصود بالمسائل التي يمكن حلها بواسطة الحاسوب. فليس كل قضية مجهولة تسمى مسألة قابلة للحل بواسطة الحاسوب. انظر مثلاً القضايا التالية لا تعتبر مسائل قابلة للمعالجة بواسطة الحاسوب:

❖ ما هو اسم أطول رجل في العالم؟

❖ أين تقع مدينة أغادير؟

❖ كم هي مساحة ليبيا؟

فالمسائل التي يعالجها الحاسوب تتضمن الحصول على حل (مجهول) لقضية اعتماداً على مجموعة من المعطيات المرفقة بنص المسألة. هذه المسائل قد تُنفذ من خلال خطوة واحدة أو مجموعة من الخطوات التي تُنفذ وفق تسلسل معين. النماذج التالية تمثل أمثلة للمسائل التي يمكن حلها بواسطة الحاسوب:

- ❖ حساب مساحة دائرة نصف قطرها 5 سنتيمترات.

- ❖ تحديد العدد الأكبر ضمن فئة الأعداد {4،12،62،18،34}.
- ❖ طباعة الأعداد الزوجية الموجبة المحصورة بين العدد 14 و 65.

3.1 حل المسائل

هذه الفقرة تُبين ما يجب فعله من خطوات لحل أي مسألة. في كثير من الأحيان قد يفضل الطالب في حل مسألة ما بسبب عدم فهمه أو سوء فهمه للمطلوب إيجادها. لذلك يجب أن يُدرك الطالب كيف يفكك المسألة حيث يقرأ المسألة ويفهمها جيداً لتحديد معطياتها وما يجب الحصول عليه (الحل). المعطيات قد تشمل بيانات معينة (مثل نصف القطر) وأي قوانين يجب تنفيذها (مثل قانون حساب مساحة الدائرة) وهكذا.

4.1 صياغة خطوات الوصول إلى حل المسألة

هذه الفقرة تُبين أهمية صياغة الخطوات اللازمة للوصول إلى حل أي مسألة. وعند الاستعانة بالحاسوب لحل المسائل يجب تلقيه كيفية حل المسألة عبر توضيح خطوات الحل خطوة خطوة وهو سيقوم باتباع نفس الخطوات وصولاً إلى الحل.

5.1 دور الحاسوب في حل المسائل

هذه الفقرة تُبين للطالب الدور المحدود للحاسوب في حل المسائل. في كثير من الأحيان يعتقد بعض الطلاب أنه بإمكان الحاسوب إيجاد الحل الصحيح لأي مسألة. بل أنه بإمكانه تصحيح الأخطاء التي ترد في صياغة الحلول التي يصيغها المبرمج البشري. من المهم أن يدرك الطالب أن المسؤولية تقع على المبرمج البشري في صياغة الخطوات الصحيحة لحل أي مسألة فالحاسوب يقوم بتنفيذ خطوات الحل حرفياً ولو كانت خاطئة.

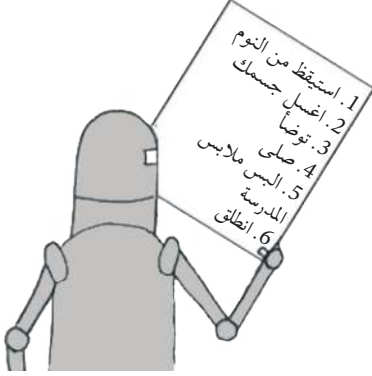
6.1 بماذا يتفوق الإنسان على الحاسوب

في هذه الفقرة يتعلم الطالب أن الحاسوب يتفوق على الإنسان من حيث قدرته على تخزين كميات

هائلة من المعلومات وسرعته الفائقة في استرجاع المعلومات المخزنة به. فذاكرة الحاسوب يمكنها تخزين كميات هائلة من المعلومات ما يعجز الإنسان عن استيعابه.

7.1 الخوارزمية

اشرح للطالب معنى الخوارزمية كمصطلح يعبر عن صياغة حل المسائل على هيئة سلسلة من الخطوات الإجرائية اللازمة لحل مسألة ما. اعتماداً على الشكل التوضيحي (جانباً) بين للطالب أن عملية الذهاب للمدرسة صباحاً يمكن وصفها عبر فقرة متواصلة من الجمل الاعتيادية ولكن يمكن أيضاً تفكيكها إلى مجموعة جمل تمثل سلسلة الأعمال التي يقوم بها الطالب كل صباح قبل الذهاب للمدرسة. أهمية صياغة الخوارزمية على هذا النحو هو معرفة تسلسل أحداث الخوارزمية خطوة خطوة. استعن بالشكل (1-1) والشكل (2-1) لشرح أمثلة أخرى للخوارزمية.



الشكل (1-1): خوارزمية حساب صافي الراتب الشهري لموظف

1. ابدأ
2. اقرأ بيانات راتب الموظف ورمز له بالحرف R
3. اقرأ بيانات عدد أيام الغياب ورمز له بالحرف D
4. احسب خصم الغياب = 5 دينار * D
5. احسب صافي الراتب = راتب الموظف - خصم الغياب
6. اطبع صافي الراتب
7. توقف

الشكل (2-1): خوارزمية حساب المتوسط الحسابي لثلاثة أعداد

1. ابدأ
2. اقرأ بيانات العدد الأول
3. اقرأ بيانات العدد الثاني
4. اقرأ بيانات العدد الثالث
5. احسب حاصل جمع بيانات العدد الأول والثاني والثالث
6. المتوسط الحسابي = حاصل جمع الأعداد على عددها
7. اطبع المتوسط الحسابي
8. توقف

8.1 حل تمارين الفصل

.1

1. ابدأ
2. اقرأ بيانات س
3. اقرأ بيانات ع
4. احسب: $ص = س + ع$
5. اطبع قيمة ص
6. توقف

.2

الإجابة ✓ / X	السؤال
✓	حساب متوسط درجات الحرارة خلال أيام شهر معين
X	شهر ديسمبر هو آخر شهور العام
✓	تحديد ما إذا كان العام 2010 سنة بسيطة أم كبيسة
✓	تحديد الوزن الملائم لشخص ما بمعرفة طولته بالسنتيمترات
✓	تحديد العدد الأكبر ضمن عناصر فئة تتكون من 10 أعداد صحيحة

.3

1. ابدأ
2. اقرأ بيانات عمر الفرد الاول س1
3. اقرأ بيانات عمر الفرد الثاني س2
4. اقرأ بيانات عمر الفرد الثالث س3
5. اقرأ بيانات عمر الفرد الرابع س4
6. احسب: $ص = س1 + س2 + س3 + س4$
7. احسب المتوسط $ع = ص ÷ 4$
8. اطبع قيمة المتوسط ع
9. توقف

2

Flow charts

الفصل الثاني:

خرائط التدفق

1.2 نواتج التعلم

- ❖ لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ معرفة أهمية الاستعانة بالرسوم في وصف حل المسائل.
- ❖ الإلمام بخرائط التدفق والرموز المستخدمة في إعدادها ودلالات كل رمز.
- ❖ الإلمام بطريقة استخدام خرائط التدفق في وصف خطوات الخوارزميات.

2.2 أهمية الاستعانة بالخرائط في وصف الأشياء

فهذه الفقرة هي من أجل أن يستوعب الطالب دور المخططات الرسومية في تسهيل فهم المسائل وطرق حلها. استعن بالأمثلة الواردة في الكتاب كخارطة الكنز أو خارطة طرق المدينة وبين سهولة الاستعانة بالخارطة مقارنة بطرق وصف الحل الأخرى سواء المكتوبة أو المنطوقة. عند ذكر خارطة طرق المدينة أو القرية امنح الطالب مثلاً على كيفية إرشاد شخص غريب للوصول إلى أحد معالم المدينة أو القرية حيث تقع مدرستك. وبين سهولة ذلك مقارنة بالوصف المكتوب أو الشفهي.

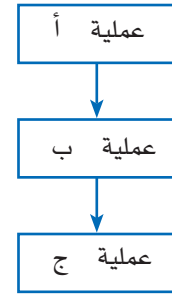
3.2 خرائط التدفق

في هذه الفقرة بين للطالب أن حل المسائل غالباً ما يتخلله أحد العمليات الثلاث: إدخال أو معالجة أو إخراج بالإضافة إلى عمليات تحكم مثل اختبار الشروط أو التكرار. لذلك فحين يتم صياغة حل أي مسألة

على صورة خوارزمية فإن خطوات أي خوارزمية لا تخرج عن نطاق هذه العمليات. خرائط التدفق هي عبارة عن مخططات رسومية تمنح شكلاً هندسياً معيناً لكل عملية من عمليات المعالجة، ومن ثم يمكن التعبير عن خطوات أي خوارزمية بواسطة مخطط رسومي يسمى خارطة تدفق الحل لمسألة ما. اشرح للطلاب أنماط المعالجة التي يمكن أن تتخلل أي خارطة ممثلة في نمط العمليات المتتابعة أو العمليات المكررة أو عمليات التفرع واستعن بالأمثلة الواردة في الكتاب لإيصال المفهوم لمدارك الطلاب.

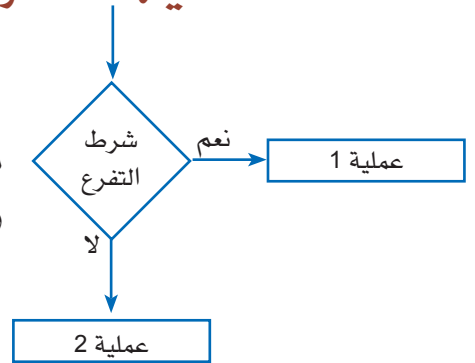
4.2 العمليات المتتابعة

اشرح للطلاب الشكل (جانباً) الذي يبين إنجاز حل مسألة ما على هيئة مجموعة من العمليات المتتالية. وبين أهمية مراعاة تسلسل تنفيذ العمليات من حيث أولوية التنفيذ. الخطأ في أولوية تنفيذ العمليات قد يساهم في الوصول إلى حل خاطئ.



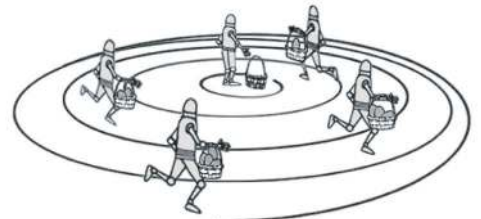
5.2 عملية اتخاذ قرار من أجل تفرع

في هذه الفقرة، أوضح للطلاب أن ثمة نوع من المسائل لا يتخلل حلها مجموعة محددة من العمليات التي تُنفذ بالتتالي. هذه المسائل لها أكثر من حل وذلك وفقاً لتحقق شرط معين أو مجموعة شروط. اشرح للطلاب كيفية تمثيل عملية التفرع المشروط باستخدام الرمز المبين جانباً.



6.2 العمليات المكررة في خرائط التدفق

اشرح للطلاب أن حلول بعض المسائل قد يتخللها تنفيذ عملية معينة أو مجموعة من العمليات بصورة مكررة واعدد معين من المرات. بالطبع يمكن إعادة رسم رمز العملية بعدد مرات تكرارها ضمن نفس خارطة التدفق،

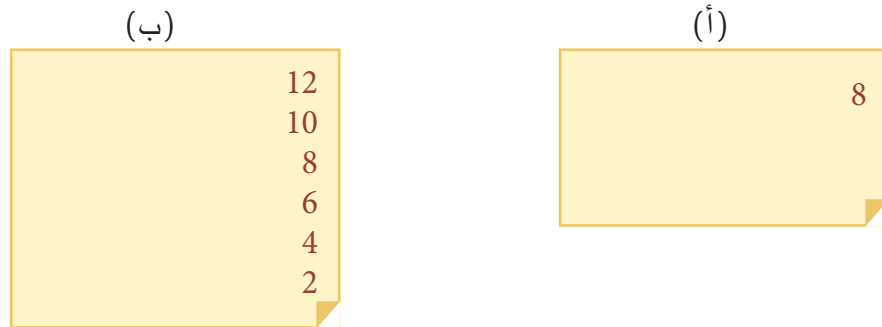


7.2 حل تمارين الفصل

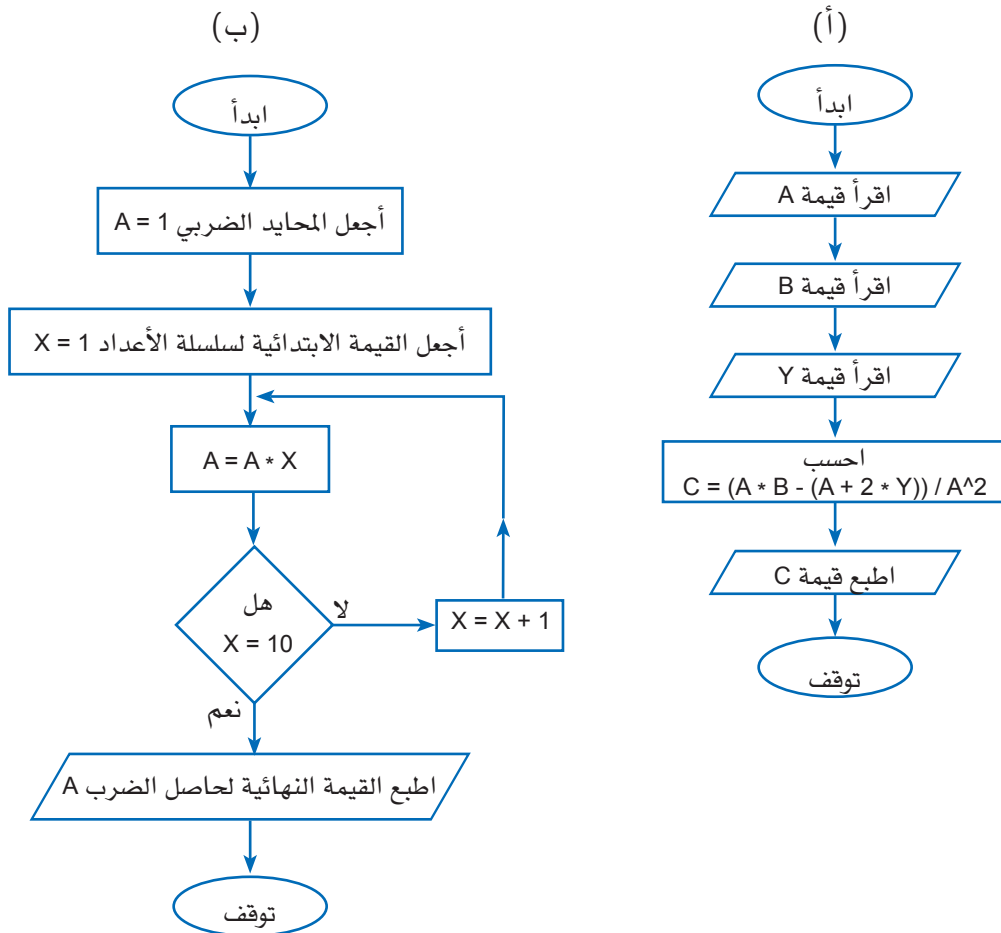
لكن هذا ممكن عندما يكون عدد مرات التكرار صغيراً. كلما أعيد رسم عملية مكررة زاد حجم خارطة التدفق وهو ما يجعلها تشغل صفحات عديدة يصعب تتبعها.

7.2 حل تمارين الفصل الثاني

.1



.2



3

Fundamentals of programming

الفصل الثالث:

أساسيات البرمجة

1.3 نواتج التعلم

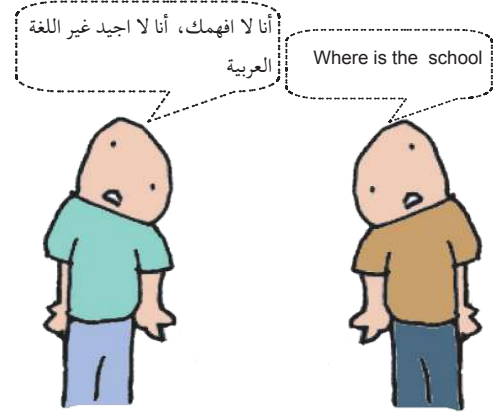
- ❖ لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ تعريف مفهوم برمجة الحاسوب وقدراته الحقيقية في حل المسائل.
- ❖ تسمية أمثلة للغات البرمجة السائدة.
- ❖ تسمية عناصر بناء أي برنامج لحل مسألة ما بواسطة أي لغة برمجة.
- ❖ وصف كيفية تخزين واسترجاع البيانات في ذاكرة الحاسوب.
- ❖ تسمية أنواع البيانات وطريقة تسمية مواقع الذاكرة لتخزين البيانات لمعالجتها لاحقاً.
- ❖ تحويل صيغ التعبيرات الرياضية الاعتيادية إلى صيغ يتعامل بها الحاسوب وتأثير ترتيب الإشارات الحسابية في مخرجاتها.

2.3 مفهوم برمجة الحاسوب

غالباً ما يعاني الطلاب حتى في مرحلة التعليم العالي من مشكلة إدراك ما المقصود بالبرمجة. ومن المدارك الخاطئة لمفهوم البرمجة أن الطالب يعتقد أن حفظ برنامج معين سيمكنه من حل مسائل برمجية شبيهة. من المهم تعليم الطالب أن البرمجة هي عملية بناء تدريجي للحل وليس استرجاع أو تذكر للحل اعتماداً على حلول برمجية سابقة. اشرح للطالب العمليات الأساسية لحل المسائل برمجياً من إدخال أو معالجة أو طباعة أو تكرار أو تنفيذ عملية حسابية، ودعه يعي أن هذه بمثابة الوحدات الأساسية لبناء أي برنامج لحل مسألة معينة.

3.3 تعريف لغة البرمجة

لأن هذا المقرر يتعلق بأحد لغات برمجة الحاسوب ففي هذه الفقرة يُراد إيصال مفهوم اللغة عموماً لمدارك الطالب. فاللغة إنما جُعِلت للتواصل بين أي كائنين. وفي حال غياب لغة مشتركة بين كائنين أو شخصين لن يمكنهما التفاهم. فمن خلال الشكل الكاريكاتوري المصاحب للفقرة لاحظ أن الشخصين لن يستطيع أحدهما خدمة الآخر لإنعدام صيغة تواصل مشترك وهي اللغة. لغات برمجة الحاسوب هي أيضاً مجموعة من الصيغ والتعابير التي طورت لتأمين التواصل مع الحاسوب وجعله يقوم بحل المسائل نيابة عن الإنسان.

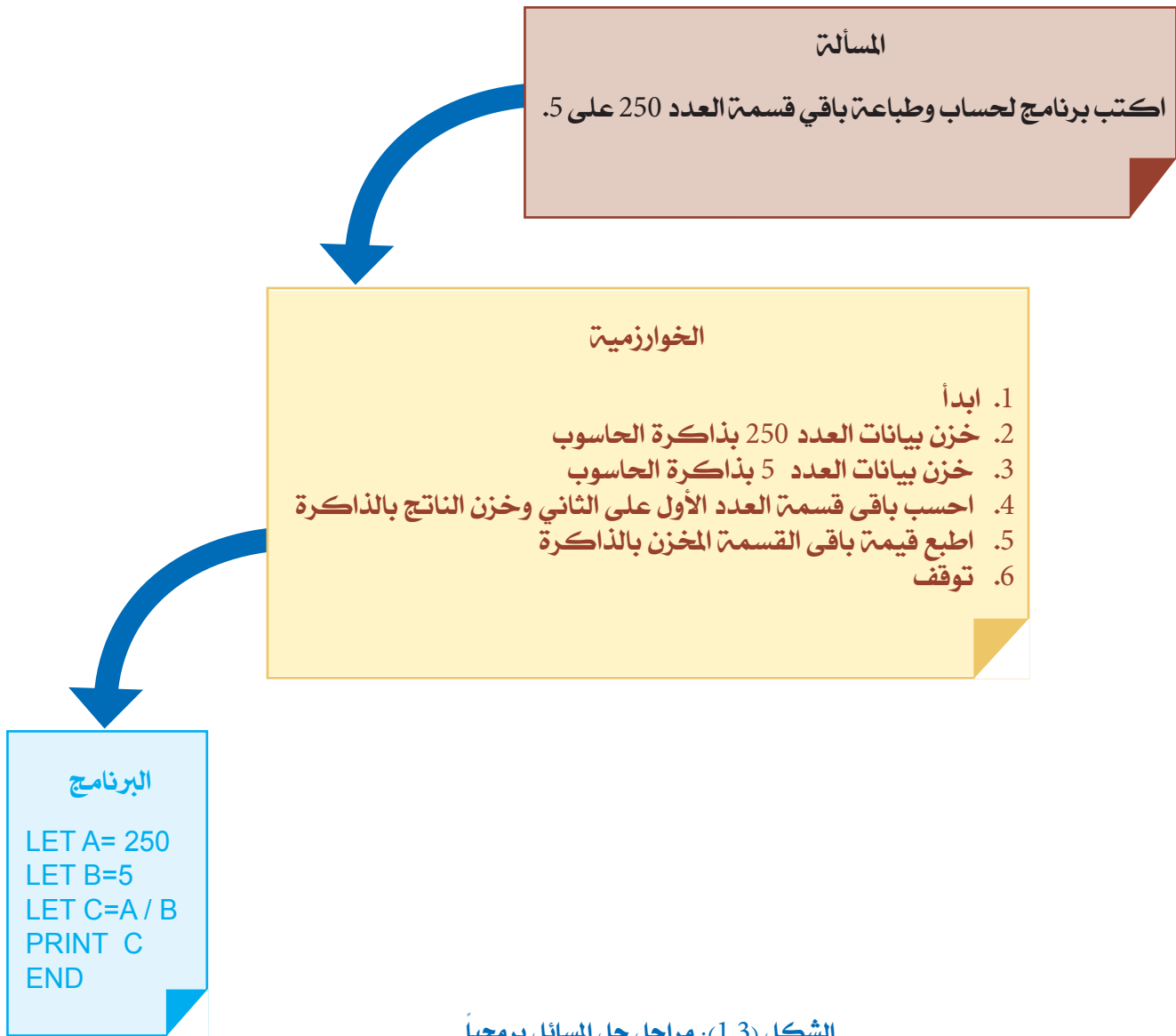


4.3 البرنامج

هذه الفقرة تهدف إلى تعليم الطالب ما المقصود بالبرنامج، وهو وحدة التواصل لتلقين الحاسوب حل أي مسألة. أما من حيث مكوناته فما هو إلا مجموعة من الأوامر التي صيغت بلغة برمجة معينة. ومن المهم ربط مفاهيم فقرات هذا الدرس ببعضها وصولاً لفهم أشمل من قبل الطالب.

5.3 العلاقة بين الخوارزمية والبرنامج

لقد تعرضت في درس سابق لتعريف الخوارزمية وصيغتها وتسلسل تنفيذها. ورغم أن الخوارزمية تعتبر صيغة حل مسألة معينة إلا أن الحاسوب لا يمكنه فهمها. فالخوارزمية قابلة للفهم من قبل الإنسان فقط. لتمكين الحاسوب من فهم تفاصيل أي خوارزمية وإيجاد حل المسألة المعنية فلا بد من ترجمتها إلى صيغة يفهمها الحاسوب. من المعلوم أن الحاسوب لا يفهم إلا صيغ لغات البرمجة، لذلك فترجمة الخوارزمية إلى صيغة يفهمها الحاسوب تسمى برنامج. استعن بالشكل (3-1) لتوضيح مراحل حل المسائل برمجياً من الصيغة العادية للمسألة إلى كتابة الحل على هيئة خوارزمية وصولاً إلى ترجمة الخوارزمية إلى برنامج بلغة برمجة معينة.



الشكل (1.3): مراحل حل المسائل برمجياً

6.3 لغات البرمجة

في هذه الفقرة يُشار إلى تعدد لغات البرمجة وميزة هذا التعدد من حيث الأغراض المختلفة التي يمكن برمجة مسائلها، فبعض لغات البرمجة تعتبر متخصصة في مجال محدد واخرى تعتبر ذات أغراض متعددة.

7.3 تخزين البيانات بذاكرة الحاسوب

في هذه الفقرة يعطى الطالب فكرة عامة عن تنظيم ذاكرة الحاسوب حيث البيانات تخضع إلى صيغ

متعددة عند تخزينها، وذلك بحسب نوع البيانات المراد تخزينها بالذاكرة. استعن بالأشكال الواردة بالفقرة لإيصال المعنى لمدارك الطلاب.

8.3 مواقع تخزين البيانات

الغرض من هذه الفقرة هو شرح الاستغلال الاقتصادي للذاكرة. ويمكنك تشبيه تنظيم محتوى الذاكرة بصناديق البريد ذات الأحجام والأغراض المختلفة. استعن بالأشكال الواردة بالفقرة لإيضاح الاحتياجات المختلفة لكل نوع من البيانات عند تخزينها بالذاكرة.

9.3 الثوابت

اشرح للطلاب أن الثوابت هي أيضاً صيغة لتخزين البيانات بالذاكرة ولكن لا يسمح بتعديلها إثر تخزينها أول مرة بعكس المتغيرات.

10.3 المتغيرات

اشرح مفهوم المتغيرات على أنها أماكن تخزين بيانات بالذاكرة تُمنح اسماً مميزاً ويمكن تخزين البيانات بها مباشرة مع إمكانية تعديل محتواها عبر إجراء أي عملية حسابية أو استبدالها ببيانات أخرى. اشرح للطلاب قواعد إطلاق تسميات للمتغيرات حيث يخضع ذلك لقواعد تم ذكرها ضمن فقرة كتاب الطالب.

11.3 أنواع المتغيرات

هنا يتم الإشارة إلى اختلاف المتغيرات باختلاف البيانات المراد تخزينها بها (رقمية صحيحة، حقيقية، نصية).

12.3 أولوية تنفيذ العمليات الحسابية

هذه الفقرة لشرح أولوية تنفيذ العمليات الحسابية عندما يتخلل أي عبارة حسابية إشارات حسابية مختلفة. راجع مع الطلاب جميع الأمثلة الواردة بكتاب الطالب مع إيضاح الأخطاء الحسابية الممكنة.

حدوثها عند إهمال مسألة أولوية تنفيذ العمليات الحسابية.

13.3 استخدام الأقواس في تنفيذ العمليات الحسابية

هذه الفقرة تشرح استثناء الالتزام بأولوية تنفيذ العمليات الحسابية. فبعض المسائل الرياضية تتطلب تنفيذ عملياتها وفق نسق معين يخالف قواعد أولوية تنفيذ الإشارات ولذلك يمكن اللجوء لاستخدام الأقواس للقيام بذلك. راجع مع الطلاب الأمثلة الواردة بكتاب الطالب مع إيضاح تجنب الالتزام بأولوية تنفيذ العمليات باستخدام الأقواس.

إضافة للمعاملات الحسابية فإن برمجة حل المسائل قد يتخللها بعض المقارنات المنطقية وذلك باستخدام المتباينات ($=$ ، $>$ ، $<$ ، $<=$ ، $>=$ ، $<$ ، $>$). اشرح للطلاب النتيجة المنطقية (صواب) أم (خطأ) المحتملة لكل متباينة مع إيضاح ذلك من خلال الأمثلة المرفقة بالفقرة.

14.3 تخصيص البيانات في المتغيرات بالذاكرة

هذه الفقرة تتعلق باستخدام جمل التخصيص لتسمية مواقع معينة بالذاكرة لتخزين بيانات محددة. من خلال أمثلة خرائط التدفق لعلك لاحظت خطوات إدخال أو تعريف بيانات محددة كمعطيات أو تنفيذ عبارات حسابية معينة. جميع هذه الخطوات تُنفذ بواسطة ما يسمى بجمل التخصيص. كل جملة تخصيص تتعلق بمتغير معين. قد يكون محتوى المتغير عبارة عن قيمة مطلقة أو محتوى متغير آخر أو أنها ناتج حساب عملية حسابية.

15.3 حل تمارين الفصل الثالث

1. اختر الاجابة الصحيحة ضمن الخيارات المدرجة بكل سؤال.

- | | | |
|-----------|-----------|-----------|
| 1.1 - (أ) | 2.1 - (ت) | 3.1 - (أ) |
| 4.1 - (أ) | 5.1 - (ب) | |

.2

اسم المتغير	[✓ / X]	سبب الخطأ أو العلة
S40	✓	
Address_#	X	اسم المتغير تخلله رمز خاص (#)
Z X1	X	اسم المتغير تخلله فراغ
205	X	اسم المتغير لم يبتدئ بحرف
Age%	✓	
2A	X	اسم المتغير ابتداء برقم
Address	✓	
FOR	X	هذه كلمة مجوزة ولها وظيفة محددة ضمن لغة البيسك المرئي

.3

التعبير الرياضي	النتيجة
$Y = 4 + 3 * 5 - 2$	17
$Y = (6 + 3) * 4 - 2$	34
$Y = 4^2 / 4 - 2$	2
$Y = (15 - 3) / (7 - 5)$	6
$Y = 4 + 3 - 3 * 5$	-8
$Y = 40 / 2 / 2^2$	5
$Y = 4^2 / (4 - 2) * 3$	24

4

الفصل الرابع: Introduction to Programing with Visual Basic

مدخل إلى البرمجة بلغة البيسك المرئي

1.4 نواتج التعلم

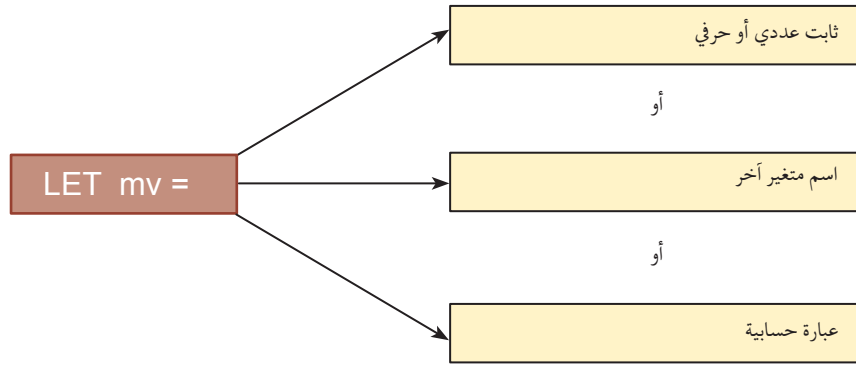
- ❖ لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ الإلمام بصياغة جمل تخصيص مختلف أنواع البيانات بالذاكرة.
- ❖ الإلمام بصياغة تعليمة طباعة البيانات المعالجة وتشكيلها.
- ❖ اكتساب مهارة ترجمة خوارزميات بسيطة إلى برامج بلغة البيسك المرئي.

هذا الدرس يشكل شرحاً لأول وحدات بناء برامج لغة فيجوال بيسك ممثلة بجمل التخصيص (LET) وجمل الطباعة (PRINT). لقد هدفنا بذلك إلى تمكين الطالب من كتابة أول برنامج قصير جداً. وتتمثل البرامج القصيرة في عملية تخزين قيم معينة بذاكرة الحاسوب وطباعة هذه القيم من الذاكرة.

2.4 جملة تخصيص البيانات LET

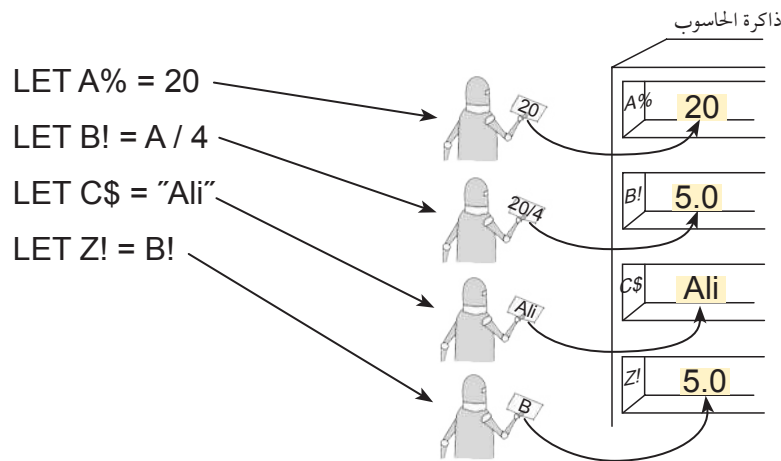
اشرح للطلاب وظيفة هذه الجملة من حيث استخدامها في تسمية متغيرات يتم استغلالها في إدخال معطيات أو إجراء عملية حسابية أو استبدال قيمة متغير بأخرى على النحو المبين بالشكل التالي:

راجع مع الطلاب محتوى الشكل (1-4)، أعطي الطلاب مجموعة أمثلة لجملة (LET) منفردة واستخدم الرموز المختلفة لتمثيل متغيرات الأعداد الصحيحة والحقيقية (العشرية) وكذلك البيانات الحرفية. قم بتبنيه الطلاب إلى أنه في حال عدم استخدام أي من هذه الرموز فالمتغير سيعتبر متغيراً مرناً بصورة تلقائية.



الشكل (1.4): الشكل العام لجمل التخصيص

لاكتمال فهم الطالب لدور جمل التخصيص (LET) استعن بالشكل (2-4) الذي يبين علاقة جمل (LET) وتنظيم البيانات بالذاكرة. ومن الجدير تنبيه الطالب إلى أنه يمكن الاستغناء عن استخدام الكلمة (LET) ضمن جمل تخصيص البيانات والاكتفاء بتسمية المتغير والقيمة المراد تخزينها به على النحو المبين بالأمثلة المعروضة في كتاب الطالب.



الشكل (2.4): مخطط توضيحي يبين شغل الذاكرة ببيانات معينة اثر تنفيذ جمل التخصيص

3.4 أخطاء شائعة عند التعامل مع جمل التخصيص

من خلال الأبحاث المتعددة والتي أجريت حول الصعوبات التي يواجهها الدارسون لمبادئ البرمجة، نوه الباحثون إلى مجموعة من الأخطاء الشائعة في إدراك مفاهيم المتغيرات وكيفية معالجة بياناتها. هذه الفقرة تخللها نماذج للتصورات غير الصحيحة التي تتكون في أذهان الطلاب حول

4.4 جملة طباعة البيانات PRINT

استخدام المتغيرات وكيفية تخزين البيانات خلالها، وكذلك كيفية تبديل محتواها نتيجة لتنفيذ أوامر التخصيص المختلفة. راجع مع الطلاب الأمثلة المتعلقة بما يطرأ في الذاكرة إثر تنفيذ كل نوع من أنواع جملة التخصيص، وذلك من أجل التأكد من استيعاب الطلاب لخصائص جملة التخصيص.

4.4 جملة طباعة البيانات PRINT

اشرح هنا الوظيفة الأساسية لجملة الطباعة (PRINT) وبين الصيغ المختلفة لاستخدام جملة (PRINT) كما هو مبين بالمثال التوضيحي المصاحب للفقرة. إثر ذلك قم بشرح كل صيغة من صيغ استخدام جملة (PRINT) ممثلة في:

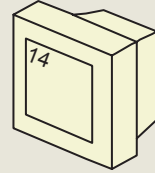
- ❖ طباعة الثوابت العددية والحرفية.
- ❖ طباعة قيمة متغير.
- ❖ طباعة متغيرات السلاسل الحرفية.
- ❖ طباعة حاصل جمع متغيرين حرفيين.
- ❖ طباعة نتيجة تعبير رياضي.

لاحظ الرسم التوضيحي المصاحب لكل صيغة مبيناً للطالب البرنامج الذي سيتم طباعته وما سوف ينتج عن تنفيذه من مخرجات.

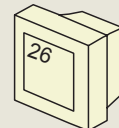
5.4 حل تمارين الفصل الرابع

.1

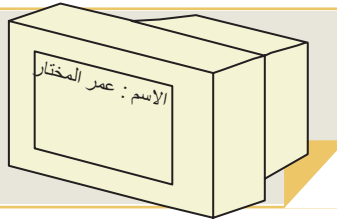
```
LET X = 2
LET Y = 6
LET Z = X+Y*2
PRINT Z
```



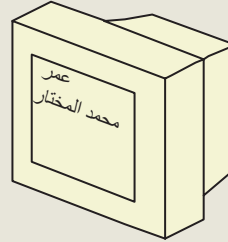
```
LET A = 5
PRINT A^2+1
```



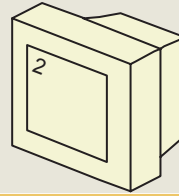
```
LET X$ = "عمر"
LET Y$ = "المختار"
PRINT Y$ + X$ + "الاسم:"
```



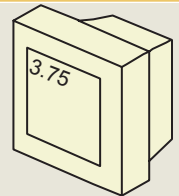
```
LET Z$ = "عمر"
LET X$ = "المختار"
LET R$ = "محمد"
PRINT Z$
PRINT X$ + R$
```



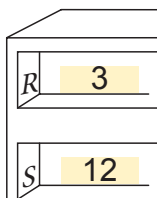
```
LET X% = 5
LET Y% = X% / 2
PRINT Y%
```



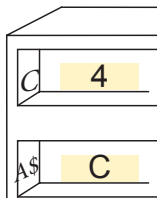
```
LET B = 15
LET C = B / 4
PRINT C
```



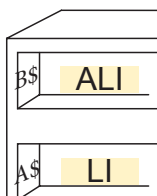
.2



```
LET R = 12
LET S = R
LET R = S/4
```



```
LET C = 6
LET A$ = "C"
LET C = 4
```



```
LET B$ = "A"
LET A$ = "L"
LET A$ = A$ + "I"
LET B$ = B$ + A$
```

5

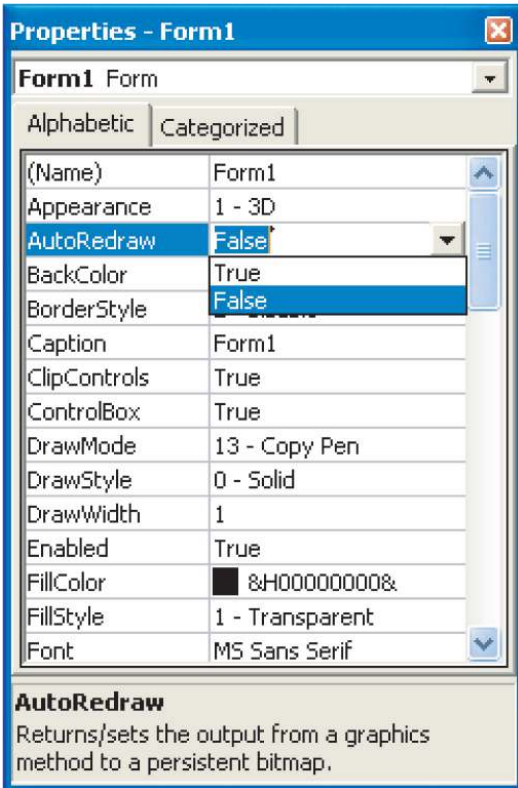
الفصل الخامس: Run Visual Basic and the implementation of an example

تشغيل بيئة لغة بيسك المرئي وتنفيذ مثال برمجي

1.5 نواتج التعلم

لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:

- ❖ تشغيل الحاسوب وتنفيذ برنامج فيجوال بيسك.
- ❖ تمييز مكونات بيئة كتابة برامج لغة بيسك المرئي بما يكفي لكتابة وتنفيذ برنامج بسيط.
- ❖ تحرير وكتابة مثال لبرنامج مبسط والتأكد من صحة الكتابة.
- ❖ كيف تكتب برنامجاً بسيطاً من خلال بيئة فيجوال بيسك وتنفيذه ومشاهدة نتائجه.
- ❖ تمييز أنواع الأخطاء البرمجية والتفريق بينها.



هذا الدرس يقوم بعرض كيفية كتابة وتنفيذ برنامج بلغة فيجوال بيسك وسيتم الاكتفاء بشرح ما يحتاجه الطالب لتنفيذ مثال بسيط. بعض التفاصيل في مكونات بيئة فيجوال بيسك تم تأخير عرضها إلى دروس لاحقة. قم بعرض محتوى الدرس من خلال اطلاع الطلاب على نماذج عملية من خلال معمل الحاسوب. نظراً لأن هذه هي المرة الأولى التي سيتم فيها التعامل مع النواحي العملية للغة فيجوال بيسك فالغرض ليس شرح تفاصيل كل ما ورد بالشاشات التي تظهر في كل خطوة. والغرض من الإيجاز في شرح مكونات الشاشات هو عدم حمل الطالب على حفظ تفاصيل لا حاجة لها في جميع حالات كتابة البرامج. بعض التفاصيل سيتم الطالب بها تدريجياً من خلال التواصل المتكرر مع الحاسوب. لذلك فالشرح سيتناول أي تفاصيل تتعلق فقط بما يحتاجه الطالب لكتابة وتنفيذ برنامج صغير. انظر مثلاً لشاشة خصائص العرض (properties)، ليس من الضرورة تشتيت تركيز الطالب بشرح محتواها الآن فكل ما يحتاجه الطالب خلال هذا الدرس هو خاصية (AutoRedraw).

لاحظ أن قائمة ابدأ قد تختلف في محتوياتها من حاسوب لآخر، فمن المهم أن يدرك الطالب هذه الخاصية لأنه قد يلحظ اختلاف بين محتوى قائمة ابدأ المبينة في الكتاب وقائمة البرامج في حاسوب المعمل أو حاسوبه الشخصي، ما قد يربكه في استيعاب باقي الخطوات.

2.5 أخطاء البرمجة

من الممارسات للمتعلمين المبتدئين للغات البرمجة ارتكابهم للأخطاء سواء عند كتابة أو تحرير البرامج عبر بيئة لغة فيجوال بيسك. من المهم التنويه على نوعي هذه الأخطاء ممثلة في الأخطاء اللغوية أو الأخطاء المنطقية. استعن بأمثلة مما ورد في كتاب الطالب للتفريق بين هذه الأخطاء. استعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحقه بكل مثال ورد بهذا الفصل.

3.5 حل تمارين الفصل الخامس

.1

- 1.1 - (ت) 2.1 - (أ) 3.1 - (ت) 4.1 - (ب)
5.1 - (ث)

.2

(أ) يتمثل الخطأ في استخدام متغيرات حرفية (لاحظ استخدام علامة \$) لتخزين بيانات عددية.

البرنامج الصحيح	البرنامج الخطأ
LET X = 2	LET X\$ = 2
LET Y = 6	LET Y\$ = 6
LET Z = X+Y	LET Z = X\$+Y\$
PRINT Z	PRINT Z

3.5 حل تمارين الفصل

ب) تتمثل الأخطاء في ورود حرف خاص (%) ضمن المتغير A (كحرف أول)، وكذلك خلو السطر الثاني من القوس الأيمن، وكذلك وضع المتغير Y ملتصق مع الجملة PRINT في السطر الأخير.

البرنامج الصحيح	البرنامج الخطأ
LET A% = 4	LET %A = 4
LET Y=4 / (A%-2)	LET Y=4 / (A-2
PRINT Y	PRINTY

3. يتمثل الخطأ المنطقي في عدم تنفيذ أسطر البرنامج وفق التسلسل الصحيح. تسلسل جمل البرنامج يجب أن تكون على النسق التالي:

```
LET A=8
LET B=5
LET C=-3
LET M = A + B + C
PRINT M / 3
```


6

Interaction with the Computer

الفصل السادس:

التفاعل مع الحاسوب

1.6 نواتج التعلم

- ❖ لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ أهمية كتابة برامج عامة تتعامل مع معطيات متعددة دون تعديل البرنامج.
- ❖ معرفة صيغة وطريقة استخدام دالة التحوير مع الحاسوب (InputBox) لإدخال المعطيات من خارج البرنامج.
- ❖ كتابة برامج بسيطة اعتماداً على دالة صندوق المدخلات (InputBox).

2.6 البرامج الخاصة والبرامج العامة

هذه الفقرة تتعلق بتوضيح الفارق بين البرامج الخاصة التي لها مخرجات ثابتة كلما تم تنفيذها، والبرامج العامة التي تسمح باستغلال نفس البرنامج للتعامل مع معطيات مختلفة ومن ثم يمكن الحصول على مخرجات مختلفة. البرامج العامة هي غالباً برامج تفاعلية تسمح للمشغل بتزويد البرنامج بالمعطيات من خارج البرنامج.

```
LET X$ = "الله"  
LET Y$ = "أكبر"  
PRINT Y$ + X$
```



لتمثيل البرامج الخاصة، استرجع مع الطلاب برنامجاً من البرامج التي تم استعراضها في الفصول السابقة (الشكل 4-9 مثلاً). وأوضح للطلاب أن هذا البرنامج سيصدر نفس النتائج في كل مرة تم تنفيذه.

الشكل (9.4)

3.6 الدالة صندوق الإدخال InputBox

اشرح للطلاب مكونات دالة صندوق الإدخال (InputBox) كوسيلة توفرها لغة الفيچوال بيسك لإنجاز البرامج العامة والتي تمنح فرصة التحوار مع الحاسوب وتزويده بالمعطيات من خارج البرنامج، وهنا لا داعي لتخصيص البيانات داخل البرنامج. استعن بالأمثلة الواردة ضمن هذا الفصل لإيصال مفهوم البرامج العامة والخاصة. واستعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتتباع الخطوات المبينة بفقرة **ممارسة** الملحقه بكل مثال ورد بهذا الفصل.

4.6 حل تمارين الفصل السادس

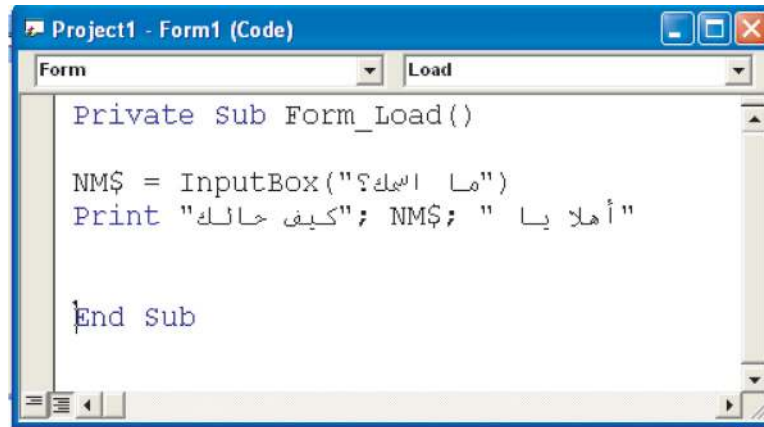
.1

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    S% = InputBox("أدخل سنة ميلادك")
    Age = 2013 - S%
    Print "عمرك: " & Age & " سنة"
End Sub
```

.2

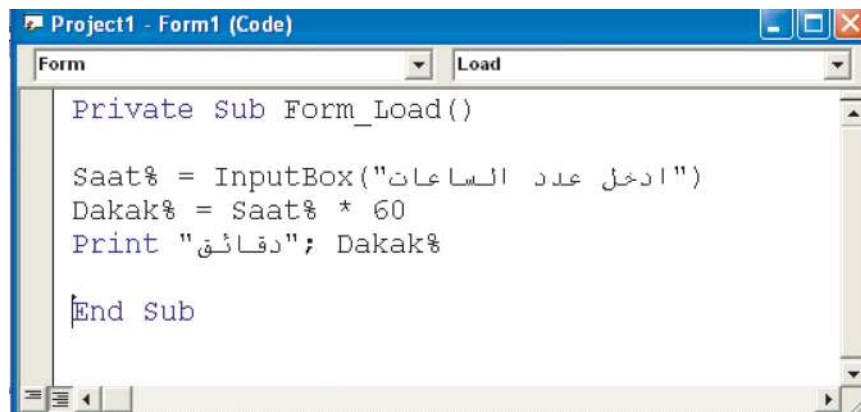
```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    NK = InputBox("ادخل نصف قطر الدائرة")
    TA = 3.14
    msaha = TA * NK ^ 2
    Print msaha & " المساحة="
End Sub
```


.3



```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    NM$ = InputBox("ما اسمك؟")
    Print "أهلا يا " & NM$ & " كيف حالك"
End Sub
```

.4



```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    Saat% = InputBox("ادخل عدد الساعات")
    Dakak% = Saat% * 60
    Print "دقائق"; Dakak%
End Sub
```


7

الفصل السابع: Condition Control programming Orders

أوامر برمجة التحكم المشروط

1.7 نواتج التعلم

- ❖ لقد صمم هذا الفصل لتحقيق مجموعة من نواتج التعلم والمتمثلة في:
- ❖ إدراك الحاجة لاستخدام جمل التحكم عند صياغة حل المسائل.
- ❖ ربط مفهوم رمز التحكم المستخدم في الخوارزمية وجمل التحكم المشروط بلغة فيجوال بيسك.
- ❖ فهم مكونات والصيغ المتعددة لكتابة جمل التحكم المشروط.
- ❖ استخدام جمل التحكم المشروط في كتابة برامج بسيطة تحتوي على تفرعات شرطية.

2.7. جملة التحكم المشروط (IF THEN ELSE)

هذا الدرس يأتي بعد استيعاب الطالب لمفهوم البرنامج الذي يتكون من مجموعة من الأوامر التي يتم تنفيذها بالتتالي. راجع مع الطلاب خرائط التدفق وأشر إلى عملية التفرع كاستثناء لعملية التنفيذ المتتالي لاسطر الأوامر. لترجمة عملية التفرع برمجياً هناك الحاجة لاستخدام جملة التحكم (If-Then-Else) موضوع هذا الدرس. هناك عدة صيغ لكتابة جملة التحكم، راجع مع الطلاب كل صيغة من هذه الصيغ وشرح المثال المصاحب لكل صيغة.

اشرح للطلاب طبيعة بعض شروط التفرع المركبة والتي يتم فيها اختبار أكثر من حالة بنفس الشرط. ثم أوضح لهم كيف يمكن استخدام الروابط المنطقية لصياغة الشروط المركبة مع بيان الفارق بين الرابط (AND) والرابط (OR) من حيث تحديد نوع النتيجة المنطقية للشروط المركبة.

استعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحقة بكل مثال ورد بهذا الفصل.

1.2.7 حل تمارين على جملة التحكم المشروط (IF THEN ELSE)

.1

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    X% = InputBox("أدخل عددا صحيحا")
    If X% > 0 Then
        Print "عدد موجب"; X%
    Else
        Print "عدد سالب"; X%
    End If
End Sub
```

.2

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    X1% = InputBox("أدخل العدد الاول")
    X2% = InputBox("أدخل العدد الثاني")
    If (X1% > X2%) Then
        Print X1%; "العدد الأكبر"
    Else
        Print X2%; "العدد الأكبر"
    End If
End Sub
```

.3

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    A = InputBox("أدخل عدد")
    X = A / 3 + 2
    If (X <= 7) Then
        X = X - 3
    End If
    Print X
End Sub
```

3.7 جملة الاختيار (Select Case)

هذا الدرس يقدم جملة تحكم من نوع آخر مقارنة بجملة (IF THEN) والتي تعتمد على مقارنة شرط له نتيجتان فقط إما صائب أو خاطئ منطقياً. جملة التحكم التي سيتم دراستها في هذا الدرس هي جملة (Select Case) وهي تعتمد على اختبار شرط متعدد القيم الناتجة عن اختباره. راجع مع الطلاب الشكل العام لجملة (Select Case) مبيناً إمكانية استخدام أكثر من جملة (IF THEN) لتحقيق نفس المعالجة.

إثر التأكد من استيعاب الطلاب للصيغة العامة لجملة (Select Case)، اشرح الأمثلة المصاحبة لترسيخ استيعابهم للدرس. واستعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحقه بكل مثال ورد بهذا الفصل.

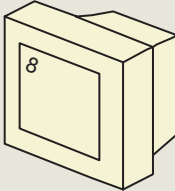
1.3.7 حل تمارين على جملة الاختيار (Select Case)

1. تتبع البرامج التالية ودون المخرجات المتوقعة إثر تنفيذه

```

A = 5
B = A - 6 / 3
Select Case B
Case 1
  Print B - 1
Case 2
  Print B + 1
Case 3
  Print A + B
End Select
  
```

(أ)

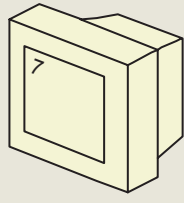


```


B4 = 4 - 2 ^ 2
Select Case B4
    Case 1
        Print B4 - 1
    Case 2
        Print B4 + 1
    Case 3
        Print B4 ^ 2
    Case 4
        Print B4 * 2
    Case Else
        Print B4 + 7
End Select

```

(ب)



.2



```

Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
X%= InputBox("أدخل عددا صحيحا موجبا أقل من خمسة")
Select Case X%
    Case 1
        Print "واحد"
    Case 2
        Print "اثنان"
    Case 3
        Print "ثلاث"
    Case 4
        Print "أربعة"
    Case Else
        Print "رقم خارج الفئة المستهدفة"
End Select
End Sub

```

4.7 جملة التكرار (For Next)

هذا الدرس يتطرق إلى عملية تحكم أخرى من العمليات التي قد يحتاجها المبرمج في صياغة بعض البرامج ذات العلاقة. وعملية التحكم هذه تقوم بتوجيه الحاسوب إلى تكرار سطر أو مجموعة من أسطر أوامر البرمجة. قم بإرشاد الطالب إلى أنه يمكن تكرار أي عملية عبر تكرار كتابة نفس الجملة بحسب المرات المراد الحصول عليها. ثم اطرح تساؤل حول الرغبة في تكرار تنفيذ عملية معينة لمرة عديدة قد تبلغ المئات أو الآلاف. سيلاحظ الطالب أن ذلك سيستغرق إعادة كتابة الأسطر المكررة مما يجعل البرنامج مطولاً جداً. هنا سيدرك الطالب أهمية إنجاز نفس العمليات بصورة مختصرة اعتماداً على جملة التكرار (FOR NEXT).



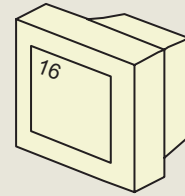
قبل الخوض في صيغة كتابة جملة (FOR NEXT) يجب شرح مفهوم الحلقة التكرارية. استعن بالشكل التوضيحي المبين بكتاب الطالب لشرح طريقة عمل الحلقات التكرارية. فلإنجاز أي عملية تكرار يجب تخصيص عدّاد على هيئة متغير يقوم بحساب عدد مرات التكرار. العدّاد يمثل سلة البرتقال التي تم استخدامها في المثال التوضيحي.

إثر التأكد من استيعاب الطلاب لمفهوم الحلقة التكرارية، انتقل لشرح صيغة كتابة جملة (FOR NEXT) وذلك كما هو مبين بالشكل المصاحب، ثم استعرض مع الطلاب الأمثلة الموضحة بكتاب الطالب. واستعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحقه بكل مثال ورد بهذا الفصل.

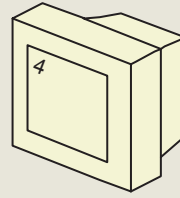
1.4.7 حل تمارين على جملة التكرار (For Next)

.1

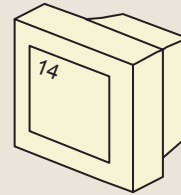
```
R = 0
For X = 1 To 7 Step 2
  R = R + X
Next
Print R
```



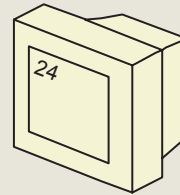
```
R = 0
For X = 1 To 7 Step 2
  If X = 5 Then Exit For
  R = R + X
Next
Print R
```



```
Y = 4
For J = 4 To 1 Step -1
  Y = Y + J
Next
Print Y
```



```
B = 1
For X = 1 To 4
  B = B * X
Next
Print B
```



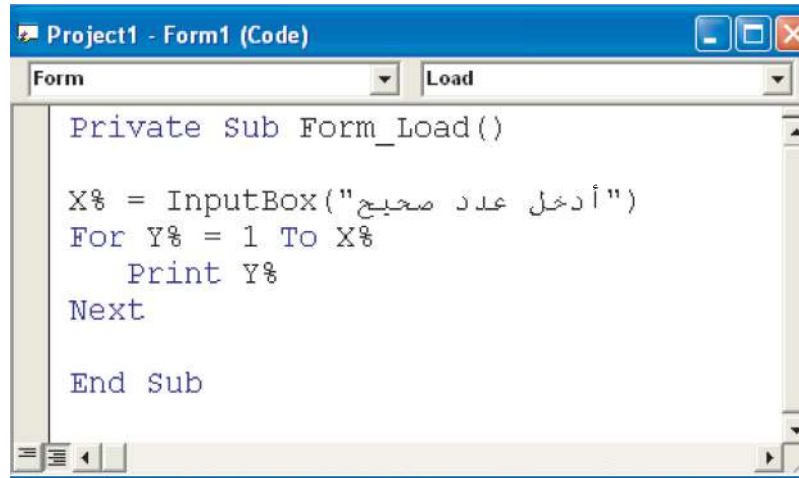
.2

```
FOR K= 2 TO 4
  PRINT "السلام عليكم"
Next
```

(أ)

```
FOR K = 2 TO 10 STEP 2
  PRINT K
Next
```

(ب)



```

Private Sub Form_Load()

    X% = InputBox("أدخل عدد صحيح")
    For Y% = 1 To X%
        Print Y%
    Next

End Sub

```

5.7 جملة التكرار (Do.. While)

هذا الدرس يتعرض إلى جملة (DO WHILE) كجملة أخرى من جمل الحلقات التكرارية. ومن التساؤلات الشائعة التي تبرز في أذهان الطلاب هي لماذا استخدام جملة أخرى لإنجاز الحلقات التكرارية؟ يجب تنبيه الطلاب أنه من الممكن إنجاز بعض الحلقات التكرارية سواء باستخدام جملة (FOR NEXT) أو جملة (DO WHILE). لكن ثمة فارق بسيط يجعل من إحداها هو الأكثر ملاءمة. أوضح للطلاب حقيقة أن الحلقات التكرارية التي تعتمد على جملة (FOR NEXT) تعتبر حلقات يتم تحديد عدد مرات تكرارها سلفاً، لكن هناك بعض المسائل تتطلب إجراء عمليات مكررة دون تحديد عدد مرات التكرار سلفاً، فمنطق المعالجة داخل الحلقة التكرارية هو الذي يحدد عدد مرات تكرارها. في هذه الحالة لا يمكن استخدام جملة (FOR NEXT) وتعتبر جملة (DO WHILE) أكثر ملاءمة في مثل هذه التطبيقات فهي تعتمد على تكرار مجموعة من العمليات طالما تحقق شرط معين يتم تحديده من خلال صيغة هذه الجملة.

استعن بالشكل التوضيحي لشرح الفارق عند استخدام جملة (DO WHILE) مقارنة باستخدام جملة (FOR NEXT) التي تحتاج لتحديد عدد دورات التكرار.

بعد إدراك الطالب لأهمية جملة (DO WHILE) ودواعي استخدامها (مقارنة بجملة (FOR NEXT))، استعن بالشكل المصاحب لإيضاح الصيغة العامة لمكونات جملة (DO WHILE).



يلي ذلك قم باستعراض الأمثلة المرفقة بالدرس لإيضاح منطق المعالجة عند تنفيذ الحلقات التكرارية اعتماداً على جملة (DO WHILE). واستعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحقة بكل مثال ورد بهذا الفصل.

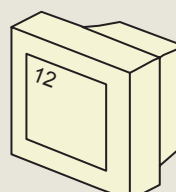
1.5.7 حل تمارين على جملة التكرار (Do.. While)

.1

(أ)

```

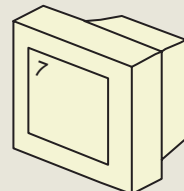
W = 4
Do While W < 12
    W = W + 2
Loop
Print W
    
```



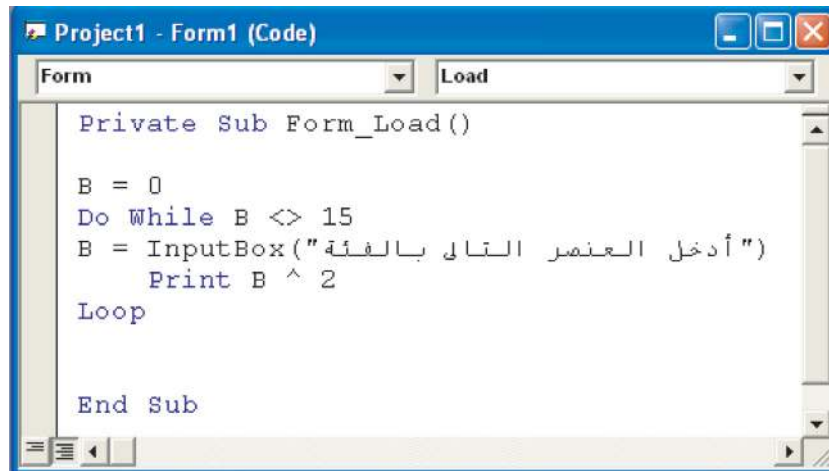
(ب)

```

X = 3
Do While X < 10
    X = X + 2
    If X + 1 > 7 Then Exit Do
Loop
Print X
    
```

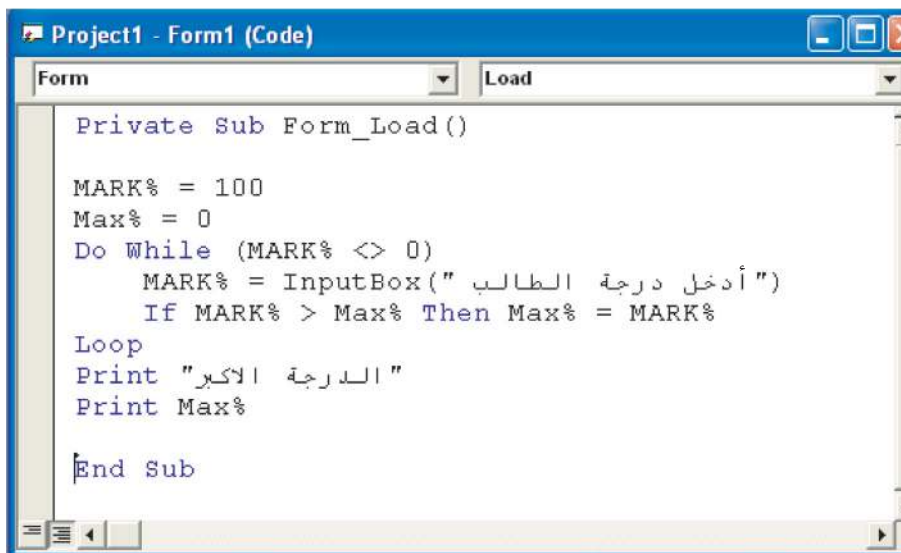


.2



```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    B = 0
    Do While B <> 15
        B = InputBox("أدخل العنصر التالي بالفئة")
        Print B ^ 2
    Loop
End Sub
```

.3



```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
    MARK% = 100
    Max% = 0
    Do While (MARK% <> 0)
        MARK% = InputBox("أدخل درجة الطالب")
        If MARK% > Max% Then Max% = MARK%
    Loop
    Print "الدرجة الاكبر"
    Print Max%
End Sub
```

6.7 جملة التكرار (Do.. Until)

هذا الدرس يتعرض إلى جملة (Do.. Until) كجملة أخرى من جمل الحلقات التكرارية. قبل البدء في شرح صيغة هذه الجملة وعرض أمثلة حولها، يجب تنبيه الطلاب إلى الفارق بين استخدام جملة (Do.. Until) وجملة (DO WHILE) لإنجاز حلقة تكرارية. ثمة فارق بسيط يجعل من إحداها هو الأكثر ملاءمة. أوضح للطلاب أن جملة (Do.. Until) تأخذ صيغة كـ **كرر حتى يتحقق شرط معين**. بمعنى أن عدد الحلقات التكرارية يتحدد بعدم تحقق شرط معين. عكس جملة (DO WHILE) التي تعتمد على اختبار الحالة التي يصير فيها شرط التكرار غير صائب منطقياً وحينها تنتهي حلقات التكرار. بعد الانتهاء من إنجاز دورة معينة يتم اختبار الشرط وتستأنف الدورات التالية طالما لم تتبدل قيمة الشرط من (خاطئ) منطقياً إلى (صائب) منطقياً.

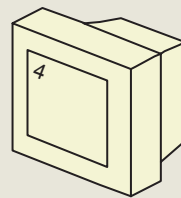
بعد إدراك الطالب لأهمية جملة (Do.. Until) ودواعي استخدامها (مقارنة بجملة DO WHILE)، استعن بالشكل المصاحب لإيضاح الصيغة العامة لمكونات جملة (Do.. Until). يلي ذلك قم باستعراض الأمثلة المرفقة بالدرس لإيضاح منطوق المعالجة عند تنفيذ الحلقات التكرارية اعتماداً على جملة (Do.. Until). أخيراً استعن بمعمل الحاسوب بالمدرسة لإعطاء الفرصة للطلاب كي يمارسوا أمثلة الفصل وذلك باتباع الخطوات المبينة بفقرة **ممارسة** الملحق بكل مثال ورد بهذا الفصل.

1.6.7 حل تمارين على جملة التكرار (Do.. Until)

.1

أ. تتبع البرنامج التالي ودون المخرجات الناتجة عنه

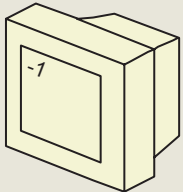
```
A = 12
B = 4
Do Until A <= B
    A = A - 1
Loop
Print A
```



1.6.7 حل تمارين الفصل

ب. حدد ما هي مخرجات البرنامج التالي عندما يتم إدخال القيمة 5 ضمن صندوق المدخلات

```
A = InputBox ("أدخل عددا صحيحا")
Do Until (A < 0)
A = A - 2
Loop
Print A
```



.2

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
N = 1
M = 0
Do Until (M = 1)
M = InputBox ("أدخل العنصر التالي بالفئة")
N = N * M
Loop
Print N
End Sub
```

.3

```
Project1 - Form1 (Code)
Form Load
Private Sub Form_Load()
MARK% = 0
Max% = 0
Do Until (MARK% = 1)
MARK% = InputBox ("أدخل درجة الطالب التالي")
If (MARK% > Max%) Then Max% = MARK%
Loop
Print Max%; "الدرجة الاكبر="
Print
End Sub
```