

Design Mostafa : 0121389578

المبرمج الصغير

الحاسب الآلي للصف الثالث الإعدادي الفصل الدراسي الأول

أسئلة محلولة ونماذج امتحانات

إعداد:

م. محمد هادي غانم

المحتويات

- أسئلة وتدريبات محلولة على الفصل الأول: مقدمة للبرمجة.
- أسئلة وتدريبات محلولة على الفصل الثاني: خرائط التدفق.
- أسئلة وتدريبات محلولة على الفصل الثالث: الكود الزائف (سودوكود).
- أسئلة وتدريبات محلولة على الفصل الرابع: مقدمة عن البرمجة الشيئية.
- أسئلة وتدريبات محلولة على الفصل الخامس: بيئة التطوير المتكاملة IDE.
- أسئلة وتدريبات محلولة على الفصل السادس: مقدمة إلى الأدوات Controls.
- خمسة نماذج امتحانات محلولة.
- قاموس الكلمات والمصطلحات الإنجليزية والبرمجية.

حاليا بالمكتبات: كتاب الشرح

المبرمج الصغير

الخطوة الأولى: تعلم البرمجة بفيجيوال بيزيك دوت نت

إعداد: م. محمد حمدي غانم



هذا الكتاب يغطي ضمناً منهج الفصل الدراسي الأول من مادة الحاسب الأول للصف الثالث الإعدادي، لكنه لا يقتصر على هذا، فهو موجه إلى المبرمج الصغير من سن ١٤

إلى ١٨ عاماً، الذي يخطو أولى خطواته في عالم البرمجة، ويريد تعلم لغة فيجيوال بيزيك دوت نت VB.NET بأسلوب شيق وجذاب.

• ويبدأ هذا الكتاب بمقدمة تحكي تاريخ لغة البيزيك باختصار.
• ثم يعرض مفاهيم البرمجة الأساسية، كنظم المعلومات ولغة الآلة واللغات عالية المستوى.

• وينتقل الكتاب بعد ذلك إلى تصميم الخوارزميات Algorithms، ورسم خرائط التدفق Flowcharts التي تمثلها، وكتابة الكود الزائف "سودوكود" الذي يعبر عن هذه الخوارزميات، مع شرح واف وأمثلة غزيرة لكل خطوة من هذه الخطوات.
• بعد هذا يعرفك الكتاب على مفاهيم البرمجة الشيئية (الكائنية) OOP، المستخدمة في فيجيوال بيزيك دوت نت.

• ثم يعرفك الكتاب على الواجهة المرئية IDE للفيجيوال بيزيك دوت نت، وأهم نوافذها وأدواتها، وكيفية استخدامها لإنشاء المشاريع وكتابة الكود وتصحيح الأخطاء.
• وأخيراً، يقدم لك هذا الكتاب أهم الأدوات Controls المستخدمة في تصميم وإنشاء البرنامج بفيجيوال بيزيك دوت نت.

وقد حاولنا قدر استطاعتنا تبسيط المادة العلمية لهذا الكتاب، لنقل خبرة البرمجة بأسلوب شيق مبسط إلى المبتدئين الذين يهَوون هذا العلم المثير، المليء بالتحدي والإبداع. فهيا لنخطو معا خطواتنا الأولى على طريق البرمجة.

يطلب الكتاب من مكتبة دار المعرفة:

٤ شارع السرايات – أمام هندسة عين شمس – بالقرب من ميدان عبده باشا –

العباسية – القاهرة.

هاتف: ٢٦٨٤٤٠٤٣

بريد إلكتروني: dar_elmaarefa@yahoo.com

صدر للكاتب عن مكتبة دار المعرفة:

- فيجيوال بيزيك وسي شارب: طريقك المختصر للانتقال من إحدى اللغتين إلى الأخرى.
- من الصفر إلى الاحتراف: فيجيوال بيزيك دوت نت ٢٠٠٨.. لتحميل ثلاثة فصول مجانية من هذا الكتاب، منها فصل عن تقنية دوت نت ٢٠٠٨ الجديدة المسماة بالاستعلام المتكامل مع اللغة Linq، وفصل عن تاريخ تطور الحاسب، وفصل عن العلاقة المثيرة بين نظم التشغيل والحمض النووي الوراثي DNA اضغط هنا: <http://www.elmaktba.com/vb2008.zip>
- من الصفر إلى الاحتراف: برمجة إطار العمل (VB.NET 2008).
- من الصفر إلى الاحتراف: سي شارب ٢٠٠٨.. لتحميل ثلاثة فصول مجانية من هذا الكتاب، اضغط هنا: <http://www.elmaktba.com/csharp2008.zip>
- من الصفر إلى الاحتراف: برمجة إطار العمل (C# 2008).

وترقبوا صدور هذه الكتب في جناح مكتبة دار المعرفة في سراي إيطاليا في معرض الكتاب يناير ٢٠٠٩ بإذن الله:

- من الصفر إلى الاحتراف: برمجة نماذج الويندوز (VB.NET 2008).
- من الصفر إلى الاحتراف: برمجة نماذج الويندوز (C# 2008).
- الفارس الأسود: العدد الأول من سلسلة الخيال العلمي (رفاق الخطر).

أماكن بيع هذه الإصدارات:

- مكتبة دار المعرفة:
- ٤ شارع السرايات – أمام هندسة عين شمس – بالقرب من ميدان عبده باشا – العباسية – القاهرة.

- دار المعارف، دار حراء، دار السحاب، مكتبة شادي، مكتبة النهضة الحديثة:

شارع شريف – وسط البلد – القاهرة.

- مكتبة الأنجلو:

ميدان باب اللوق – القاهرة.

- الجمعية العلمية:

داخل كلية الهندسة – جامعة القاهرة – الجيزة.

- مكتبة منشأة المعارف ومكتبة علاء الدين:

محطة الرمل – الإسكندرية.

- مكتبة العبيكان:

المملكة العربية السعودية.

للاستعلام عن أماكن بيع الكتب في باقي المحافظات المصرية، يمكن الاتصال بمكتبة دار

المعرفة على الهاتف ٢٦٨٤٤٠٤٣ أو عبر البريد الإلكتروني:

dar_elmaarefa@yahoo.com

للمراسلة:

مدونة رفاق الخطر:

<http://refaq.maktoobblog.com>

الفصل الأول

مقدمة للبرمجة

س ١: ما المقصود بنظام المعلومات؟

ج: نظام المعلومات هو أي نظام يجمع بين الأفراد والحاسب الآلي، ويسمح بجمع وتخزين البيانات واستخلاص المعلومات منها.

س ٢: ما الفرق بين البيانات والمعلومات؟

ج: البيانات هي الحقائق المجردة التي يتم جمعها بواسطة نظام المعلومات دون إجراء أية حسابات عليها، مثل الأرقام والحروف والتواريخ والصور والأصوات والفيديو. بينما المعلومات هي النتائج التي يمكن استنتاجها من البيانات بعد إجراء بعض العمليات عليها، كالحسابات الرياضية والمقارنة والترتيب وغير ذلك.. وتأخذ هذه المعلومات أشكالاً متعددة، مثل: التقارير والجدول والقوائم والرسوم البيانية.

س ٣: كيف يمكنك تحويل البيانات إلى معلومات؟

ج: يمكن تحويل البيانات إلى معلومات بإجراء بعض العمليات عليها، كالحسابات الرياضية والمقارنة والترتيب والفهرسة.. إلخ.

س ٤: اذكر الفرق بين اللغات ذات المستوى العالي ولغة الآلة.

ج: لغة الآلة **Machine Language**: هي مجموعة من الأوامر مكتوبة في صورة أرقام، وهي تتعامل مباشرة مع مكونات الحاسوب المادية **Hardware** كالذاكرة ووحدات الإدخال والإخراج، لهذا فهي أصعب في كتابتها وفهمها من لغات البرمجة، وعندما كان المبرمجون يكتبون برامجهم بها قديماً كان يجب أن يكونوا مهندسين ليكونوا على دراية كاملة بتركيب الحاسب.

بينما اللغة مرتفعة المستوى **High-Level Language**: هي لغة تستخدم أوامر مكتوبة بالكلمات الإنجليزية البسيطة، وتعزل المبرمج عن تركيب الجهاز المادي، وهي

بهذا أسهل بكثير من لغة الآلة واللغات منخفضة المستوى، فهي توفر وقت المبرمج وجهده، وتجعل من الممكن لأي شخص أن يصير مبرمجا.

وتستخدم لغة البرمجة عالية المستوى مترجما Compiler أو مفسرا Interpreter لتحويل أوامرها إلى لغة الآلة حتى يتمكن الحاسب من فهمها وتنفيذها.. ومن أمثلة اللغات عالية المستوى: جافا Java وسي شارب C# وفيجيوال بيزيك VB.

س ٥: ما الفرق بين المفسر والمترجم؟

ج: المترجم Compiler: هو برنامج يحول كل الأوامر المكتوبة بلغة البرمجة إلى لغة الآلة، ثم ينفذها بعد ذلك.

المفسر Interpreter: هو برنامج يحول كل أمر في البرنامج على حدة من لغة البرمجة عالية المستوى إلى لغة الآلة.. بمعنى أن المفسر يحول الأمر الأول إلى لغة الآلة ثم ينفذه مباشرة، ثم يحول الأمر الثاني ثم ينفذه.. وهكذا حتى ينتهي البرنامج.

والجدول التالي يقارن بين المترجم والمفسر:

المقارنة	المترجم	المفسر
طريقة الترجمة	تتم ترجمة البرنامج كاملاً.	تتم ترجمة كل أمر في البرنامج بمفرده على حدة.
بدء تشغيل البرنامج	أبطأ.	أسرع.
تنفيذ البرنامج	أسرع.	أبطأ.
فحص الأخطاء	يتم فحص كل الأخطاء قبل الترجمة، وتفشل الترجمة في حالة وجود أي خطأ.	يتم فحص كل أمر أثناء ترجمته وتنفيذه، ولو تم اكتشاف أي خطأ يتوقف المفسر عند السطر الذي به الخطأ ويتيح للمبرمج تصحيحه ثم مواصلة تنفيذ البرنامج.
تصحيح الأخطاء	أصعب.	أسهل.

س٦: ما المقصود بكل من وحدة الإدخال، ووحدة الإخراج؟

وحدة إدخال **Input Unit**: هي جهاز متصل بالحاسب، مهمته إدخال البيانات إلى الحاسب، مثل: لوحة المفاتيح Keyboard، والفأرة Mouse، والمسح الضوئي Scanner، ومحرك الأقراص المرنة Floppy Driver، ومحرك الأقراص الضوئية CD Driver ... إلخ.

بينما وحدة الإخراج **Ouput Unit**: هي جهاز متصل بالحاسب، مهمته إخراج البيانات منه وعرضها للمستخدم، مثل الشاشة والسماعة والطابعة... إلخ.

س٧: ما المقصود بالمتغير **Variable**؟

هو مخزن في ذاكرة الحاسب، يستطيع المبرمج أن يحفظ فيه البيانات ويقرأها منه وقتما يريد.. ويعطي المبرمج اسما للمتغير يدل على وظيفته، كما يحدد نوع البيانات التي ستوضع فيه ليتم حجز المساحة المناسبة لها في الذاكرة.. وقد سمي المتغير بهذا الاسم لأن البيانات المحفوظة فيه لا يُشترط أن تظل ثابتة طوال البرنامج، فالمبرمج يستطيع تغييرها في أي وقت.

س٨: اذكر العمليات الأساسية التي يقوم بها الحاسوب.

ج: يقوم الحاسوب بست عمليات أساسية، وهي:

١. إدخال البيانات.
٢. تخزين البيانات في الذاكرة.
٣. إجراء العمليات الحسابية.
٤. إجراء عمليات المقارنة واتخاذ القرار.
٥. تكرار تنفيذ مجموعة من الأوامر لأي عدد من المرات .
٦. إخراج المعلومات.

س ٩: اختر الإجابة الصحيحة:

- ١- برنامج إدارة الموارد البشرية في شركة الغزل والنسيج هو:
أ. بيان ب. نظام معلومات ج. معلومة
- ٢- عدد موظفي شركة الغزل والنسيج الأكبر من ٥٠ عاما هو:
أ. بيان ب. نظام معلومات ج. معلومة
- ٣- اسم أحد الموظفين في شركة الغزل والنسيج هو:
أ. بيان ب. نظام معلومات ج. معلومة
- ٤- اسم الموظف في شركة الغزل والنسيج، الذي عمل عدد ساعات إضافية أكثر في الشهر الماضي، هو:
أ. بيان ب. نظام معلومات ج. معلومة

ج: الاختيارات الصحيحة هي:

- ١- ب
- ٢- ج
- ٣- أ (لاحظ أن أسماء كل الموظفين محفوظة في نظام المعلومات باعتبارها بيانات أساسية)
- ٤- ج (لاحظ أن معرفة الموظف الذي عمل عدد ساعات إضافية أكثر يحتاج إلى إجراء عمليات جمع لمعرفة عدد ساعات العمل الإضافية لكل موظف في الشهر، ثم إجراء عمليات مقارنة لمعرفة الموظف الذي عمل عدد ساعات إضافية أكثر من الآخرين)

س ١٠: ضع العلامة (U) أمام الجملة الصحيحة، والعلامة (X) أمام الجملة الخاطئة،

مع تصحيح الخطأ:

- ١- تستخدم لغة فيجيوال بيزيك المفسر فقط.
- ٢- نظام المعلومات يخزن كل المعلومات الأساسية التي نريد معرفتها.
- ٣- يسمى الكود المكتوب بلغة البرمجة خوارزمية Algorithm.

- ٤- المترجم ينفذ البرنامج بشكل أسرع من المفسر .
- ٥- يستطيع الحاسب تحليل المعلومات واستنتاج حقائق جديدة منها .
- ٦- الرسم البياني الذي يوضح توزيع أعمار تلاميذ المدرسة هو بيان .
- ٧- فيجيوال بيزيك هي لغة برمجة عالية المستوى .
- ٨- يقرأ الحاسب المعلومات التي يدخلها المستخدم، ويحفظها في الذاكرة، ويجري عليها الحسابات والمقارنات، ثم يقوم بإخراج البيانات إلى المستخدم .

- ج: ١ - (x) والصواب: تستخدم فيجيوال بيزيك كلا من المفسر والمترجم .
- ٢ - (x) والصواب: يخزن نظام المعلومات البيانات الأساسية، ويجري عليها العمليات اللازمة ليعطينا المعلومات التي نريد الحصول عليها .
- ٣ - (x) والصواب: يسمى الكود المكتوب بلغة البرمجة برنامجاً، بينما الخوارزمية Algorithm هي فكرة البرنامج مكتوبة في شكل خطوات منظمة إذا اتبعناها تقود إلى حل المشكلة، ولا تحتوي طريقة كتابتها على أي كود برمجي .

٤ - (ü)

- ٥ - (x) والصواب: ليس للحاسب عقل ولا يمتلك أي ذكاء، لهذا لا يستطيع فهم أية بيانات أو تحليلها.. كل ما يستطيعه الحاسب هو العمليات الست الأساسية: إدخال البيانات وتخزينها في الذاكرة، وإجراء العمليات الحسابية والمقارنات عليها، وتكرار تنفيذ مجموعة من الأوامر لأي عدد من المرات، وإخراج المعلومات .
- ٦ - (x) والصواب: عمر كل تلميذ في المدرسة هو بيان، بينما الرسم البياني الذي يوضح توزيع أعمار تلاميذ المدرسة هو معلومة .

٧ - (ü)

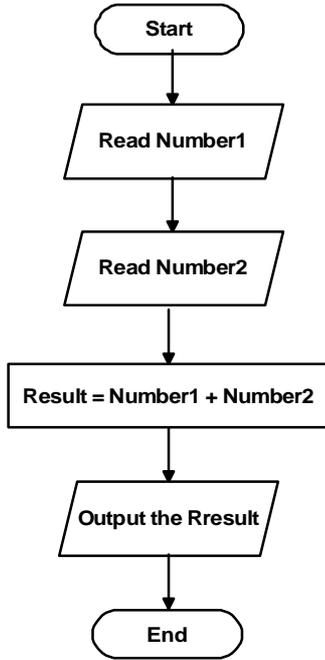
- ٨ - (x) والصواب: يقرأ الحاسب البيانات التي يدخلها المستخدم، ويحفظها في الذاكرة، ويجري عليها الحسابات والمقارنات، ثم يقوم بإخراج المعلومات إلى المستخدم .

الفصل الثاني

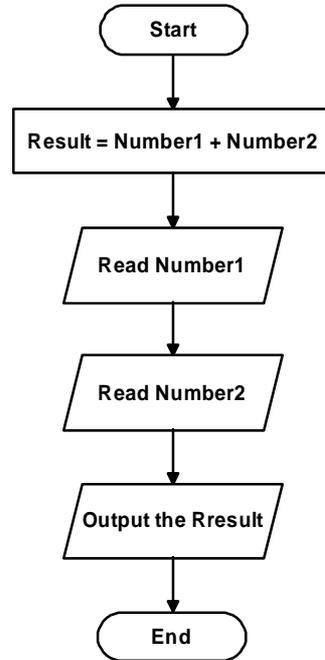
خرائط التدفق Flowcharts

س ١: قامت دنيا برسم خريطة تدفق لعملية جمع عددين وإظهار الناتج كما هو موضح في الشكل (١ - أ).
قم بتصحيح خريطة التدفق.

ج: يجب على دنيا أن تقرأ قيمتي العددين Number1 و Number2 اللتين يقوم المستخدم بإدخالهما أولاً، لكي تستطيع أن تجمعهما بعد ذلك.. لهذا فإن خريطة التدفق الصحيحة ستكون كما في الشكل (١ - ب).



شكل ١ - ب



شكل ١ - أ

س٢: ارسم خريطة تدفق لقراءة عمري هاني وهشام لمعرفة أيهما أكبر سنًا.

ج: سنستخدم في هذا المخطط متغيرين:

- HanyAge: وسنقرأ فيه عمر هاني.

- HeshamAge: وسنقرأ فيه عمر هشام.

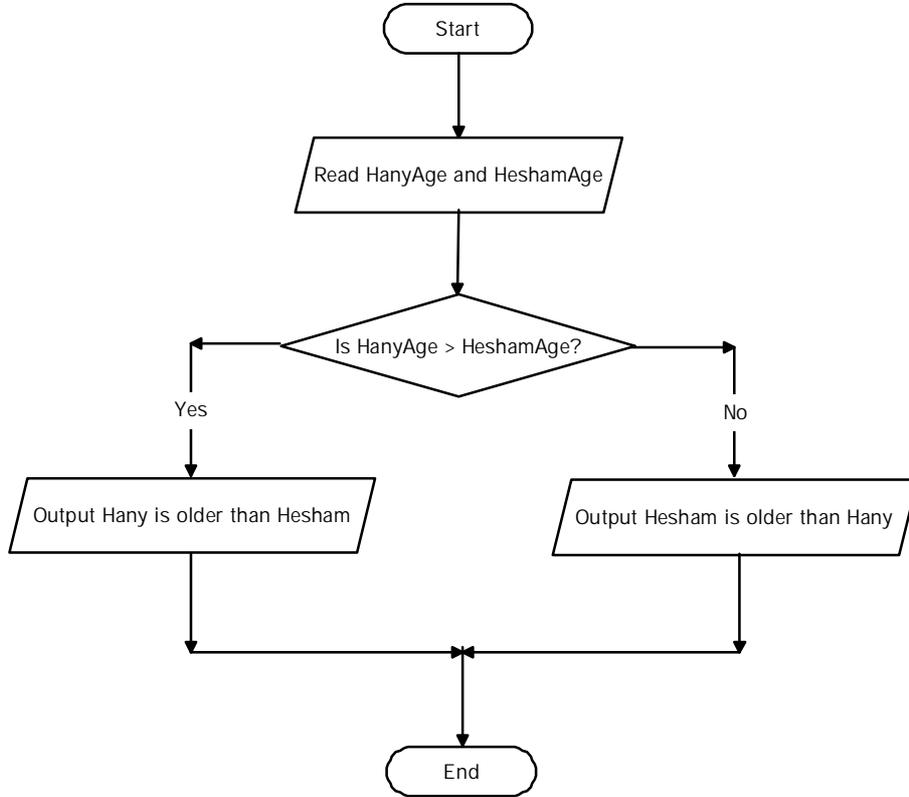
بعد هذا سنقارن قيمتي هذين المتغيرين، فإن كان عمر هاني HanyAge أكبر من عمر هشام HeshamAge نعرض على الشاشة عبارة تخبر المستخدم بذلك:

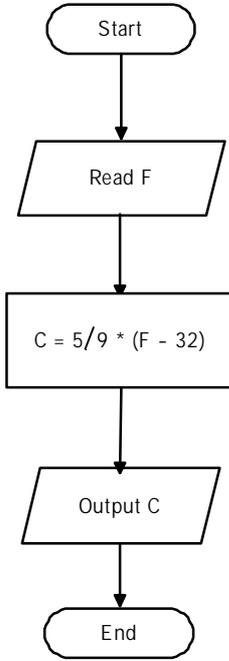
Hany is older than Hesham

وإن كانت نتيجة المقارنة غير صحيحة، نعرض على الشاشة عبارة تخبر المستخدم بأن هشام أكبر من هاني:

Hesham is older than Hany

هذه هي خريطة التدفق:





س٣: ارسم خريطة تدفق لعملية قراءة درجة الحرارة بالفهرنهايت ثم تحويلها إلى ما يقابلها بالدرجات المئوية (C°)، ثم قم بطباعة درجة الحرارة بالفهرنهايت وبالدرجات المئوية، مع العلم بأن العلاقة بينهما كالآتي:

$$C = \frac{5}{9} (F - 32)$$

ج: سنستخدم في هذا البرنامج متغيرين:

F: سنقرأ فيه درجة الحرارة بالفهرنهايت التي يدخلها المستخدم.

C: سنحسب فيه ناتج تحويل درجة الحرارة من مقياس فهرنهايت إلى المقياس المئوي، تبعاً للمعادلة:

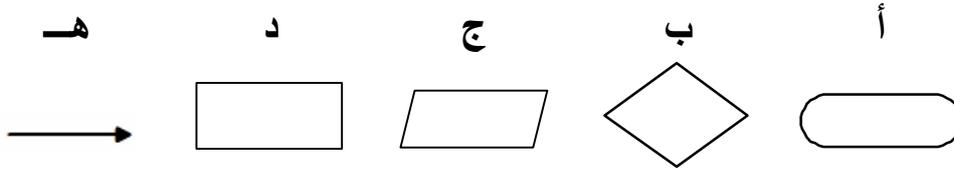
$$C = 5 / 9 * (F - 32)$$

هذه هي خريطة التدفق.

س٤: ارسم الشكل المناسب في الفراغات التالية:

- أ- يستخدم الشكل لبدء وإنهاء خريطة التدفق.
 ب- للمقارنة بين عددين نستخدم الشكل
 ج- يستخدم الشكل للإدخال والإخراج.
 د- للمعالجة نستخدم الشكل
 هـ- يستخدم الشكل لتوصيل شكل بآخر.

ج:



س ٥: ارسم خريطة تدفق لعملية قراءة نصف قطر كره (r)، ثم إيجاد حجم الكره (V)، ومساحة سطحها (A)، علما بأن المعادلات المطلوبة هي:

$$V = \frac{4}{3} \pi r^3$$

$$A = 4 \pi r^2$$

حيث إن:

$$\pi = \text{ط} = \frac{22}{7}$$

مع ملاحظة أنه إذا كان نصف القطر سالبا، يجب عليك طباعة رسالة تحذير ثم إنهاء البرنامج.

ج: سنستخدم في الحل ثلاثة متغيرات:

- r: سنقرأ في هذا المتغير نصف قطر الكرة Radius.. وستكون أول خطوة نقوم بها هي فحص قيمة هذا المتغير، فإن كانت أصغر من الصفر (عددا سالبا)، فسنعرض رسالة للمستخدم نخبره فيها بأن هذه القيمة غير صحيحة:

Radius can't be a negative number.

- A: سنحسب في هذا المتغير مساحة سطح الكرة Area، تبعا للمعادلة:

$$A = 4 * (22 / 7) * r ^ 2$$

حيث يمثل الرمز ^ علامة الأس.. ويمكننا كتابة هذه المعادلة بصيغة أخرى كالتالي:

$$A = 4 * (22 / 7) * r * r$$

حيث إن رفع العدد للأس ٢ يعني ضرب العدد في نفسه مرتين.

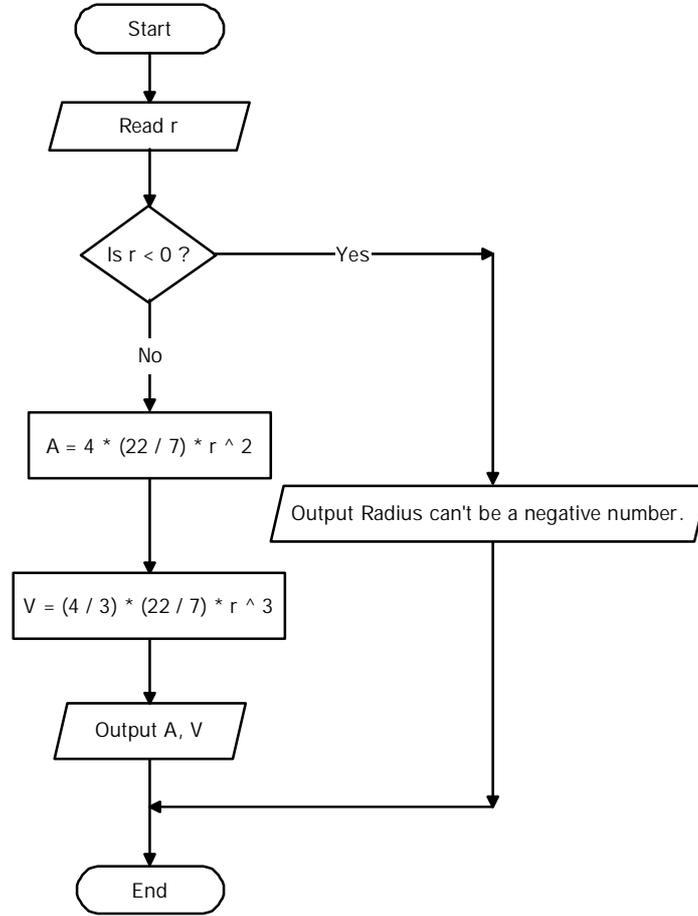
- V: سنحسب في هذا المتغير حجم الكرة Volume، تبعا للمعادلة:

$$V = (4 / 3) * (22 / 7) * r ^ 3$$

ويمكننا كتابة هذه المعادلة بصيغة أخرى كالتالي:

$$V = (4 / 3) * (22 / 7) * r * r * r$$

هذه هي خريطة التدفق.



س٦: ارسم خريطة تدفق لحساب الدالة y بعد قراءة المتغير x ، علماً بأن:

$$y = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$$

و ذلك بعد جمع أول ١٠٠ حد.

ج: الدالة Y التي نتعامل معها في هذه المسألة تمثل متتابعة Series تتكون من عدد لا نهائي من الحدود Terms.. ولحل مثل هذا النوع من المسائل يجب استخدام جملة تكرار Loop، وفي مثالنا هذا سيستمر التكرار إلى مئة لفة لجمع ١٠٠ حد..
وسنستخدم في هذا البرنامج ثلاثة متغيرات أساسية:
- المتغير X : سنقرأ فيه قيمة x التي يدخلها المستخدم.

- المتغير Y: سنحفظ في هذا المتغير قيمة الدالة الناتجة عن جمع حدود المتتابعة، حيث ستبدأ قيمة المتغير بصفر، وفي كل لفة سنجمع عليها قيمة أحد الحدود.

- المتغير C: سيشير هذا المتغير إلى رقم الحد الذي نتعامل معه، علما بأن آخر حد يجب أن نضيفه إلى المجموع Y هو الحد رقم ١٠٠.. لهذا ستبدأ قيمة المتغير C بواحد (رقم أول حد)، ثم تزيد بواحد في كل لفة.. وسنفحص المتغير C مباشرة بعد

زيادة قيمته، فإن كانت قيمته أكبر من ١٠٠

نعرض الناتج للمستخدم وننتهي البرنامج، وإن لم تكن أكبر من ١٠٠ واصلنا جمع حدود المتتابعة.

ونحتاج في هذا النوع من المسائل إلى اكتشاف العلاقة التي تربط كل حد بالحد السابق له، لنستخدم هذه العلاقة في الحصول على قيمة كل حد داخل جملة التكرار Loop.. لاحظ أن كل حد يتكون من ثلاثة أجزاء:

- الجزء الأول هو إشارة الحد Sign.. ونلاحظ في

هذه المسألة أن إشارة الحد الأول موجبة، وإشارة الحد الثاني سالبة، وإشارة الحد الثالث موجبة، وإشارة الحد الرابع سالبة.... وهكذا.. فكيف يا ترى نحصل على هذه الإشارات برمجيا؟

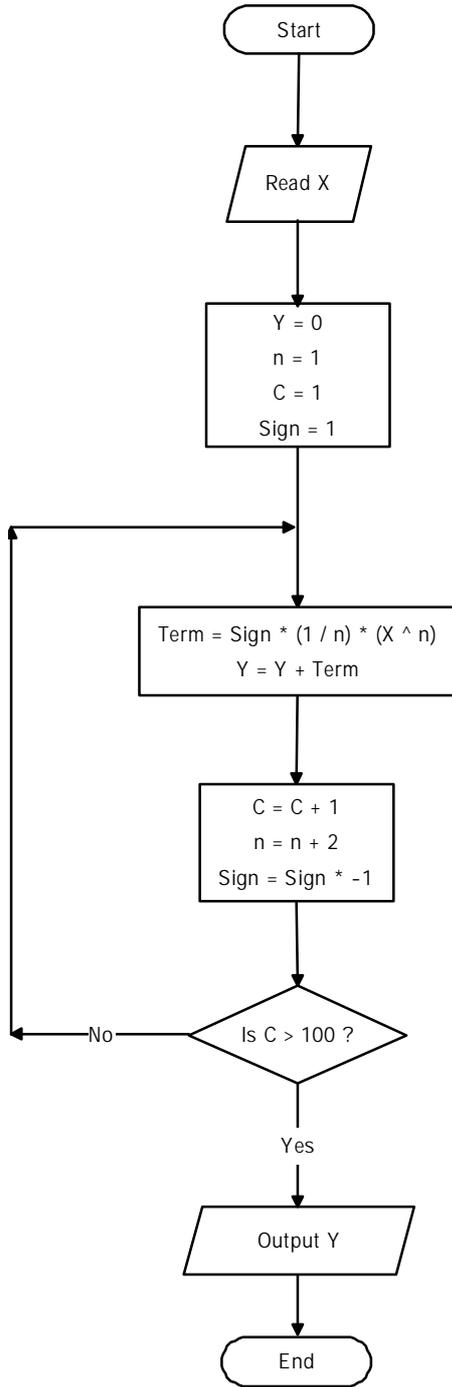
هناك حلول كثيرة لمشكلة الإشارة، أبسطها استخدام متغير اسمه Sign تكون قيمته في البداية ١، ثم نعكس إشارته في كل لفة في التكرار Loop، وذلك بضربه في -١.. كالتالي:

$$\text{Sign} = \text{Sign} * -1$$

تذكر أن:

$$1 - = 1 - \times 1$$

$$1 = 1 - \times 1 -$$



وهكذا ستتغير إشارة المتغير Sign في كل مرة نضربه فيها $x - 1$.

لكن كيف سنستفيد من المتغير Sign الذي يمثل إشارة الحد Term؟

الأمر بسيط جداً، فكل ما سنفعله هو ضرب المتغير Sign في كل حد من حدود المتابعة، وبذلك ستكون إشارة الحد موجبة إذا كانت قيمة المتغير Sign تساوي 1، وستكون إشارة الحد سالبة إذا كانت قيمة المتغير Sign تساوي -1.

- الجزء الثاني هو معامل الحد Factor، وهو العدد الموجب المضروب في المتغير x .. وفي هذه المسألة هذه المعاملات هي بالترتيب (1) و (3/1)، و (5/1) ... وهكذا.. هذا معناه أن هذه المعاملات هي كسور اعتيادية، المقام فيها يبدأ بواحد ثم يزيد المقام بمقدار 2 في كل مرة.. يمكننا إذن أن نستخدم متغيراً اسمه n تبدأ قيمته بواحد، ثم تزيد بمقدار 2 في كل حد جديد، حيث ستكون قيمة معامل الحد هي: $1/n$.

- الجزء الثالث هو الأس Power الخاص بالمتغير x في كل حد.. نلاحظ أن هذه الأسس تبدأ بواحد وتزيد بمقدار 2، أي أنها تساوي المقام.

من كل هذا، سنجد أن قيمة كل حد Term ستساوي:

$$\text{Term} = \text{Sign} * (1 / n) * (X ^ n)$$

هذه هي خريطة التدفق لهذه المسألة.

س7: ارسم خريطة تدفق لحساب الدالة y بعد قراءة المتغير x ، علماً بأن:

$$y = x^2 - \frac{x^4}{3} + \frac{x^6}{5} - \frac{x^8}{7} + \dots$$

و ذلك بعد جمع 30 حداً.

ج: هذه المسألة مشابهة للمسألة السابقة، لكن أس كل حد مختلف.. وبملاحظة بسيطة يمكنك اكتشاف أن أس كل حد يزيد على المقام بواحد، لهذا يمكنك حساب قيمة كل حد تبعاً للمعادلة التالية:

$$\text{Term} = \text{Sign} * (1 / n) * (X ^ (n + 1))$$

سأترك لك رسم خريطة تدفق هذه المسألة كتدريب.. لكن لا تنسَ تغيير الشرط " $C > 100$ " إلى " $C > 30$ " لأن المطلوب في هذه المسألة هو جمع 30 حداً فقط.

س٨: عرف كلا من: الخوارزمية Algorithm وخريطة التدفق Flowchart.

ج: الخوارزمية Algorithm: هي مجموعة من خطوات التفكير، مرتبة ترتيباً منطقياً وواضحا، إذا تتبعناه نصل إلى حل المسألة التي ن فكر فيها.

خريطة التدفق Flowchart: هي تمثيل بياني يعتمد على الرسم لتوضيح ترتيب العمليات اللازمة لحل المسألة.. بطريقة أخرى: خريطة التدفق هي خوارزمية مكتوبة بالصور لا بالكلمات.

س٩: أراد أحد الملوك أن يكافئ حكيما، وترك الملك للحكيم اختيار مكافئته، فطلب منه الحكيم أن يملأ له رقعة الشطرنج بحبوب القمح، بحيث يضع في أول خانة حبة واحدة، وفي الخانة التالية يضع حبتين، وفي التالية يضع أربع حبات، وفي التالية يضع ثمان حبات،... وهكذا، حيث تتم مضاعفة عدد حبات القمح في كل خانة عن الخانة السابقة لها.. وقد وافق الملك على هذا، لكنه فوجئ أن القمح في كل مملكته لا يكفي لملء لوحة الشطرنج بهذه الطريقة!

ارسم خريطة تدفق لحساب عدد حبات القمح الذي يحتاج إليها الملك لمكافأة ذلك الحكيم الماكر!

ج: سنستخدم متغيرا اسمه Y لحساب عدد حبات القمح، حيث ستكون قيمة Y كالتالي:

$$Y = 1 + 2 + 4 + 8 + 16 + \dots\dots$$

كما هو واضح، تمثل Y متتابعة هندسية، كل حد فيها ضعف الحد السابق له.. ونظرا لأن عدد خانات لوحة الشطرنج = ٦٤ خانة، فإن المطلوب هو جمع ٦٤ حدا من حدود هذه المتتابعة.

وهناك طريقتان لحل هذه المتتابعة:

الطريقة الأولى:

هي استخدام المعادلة العامة التي نحسب منها قيمة كل حد.. وبملاحظة بسيطة،

سنكتشف أن هذه المعادلة هي:

$$\text{Term} = 2^{(n - 1)}$$

حيث n هو رقم الحد.. فمثلا، في الحد الأول ستعطينا المعادلة $2^0 = 1$ ، وفي الحد الثاني ستعطينا المعادلة $2^1 = 2$.. وهكذا.

الطريقة الثانية:

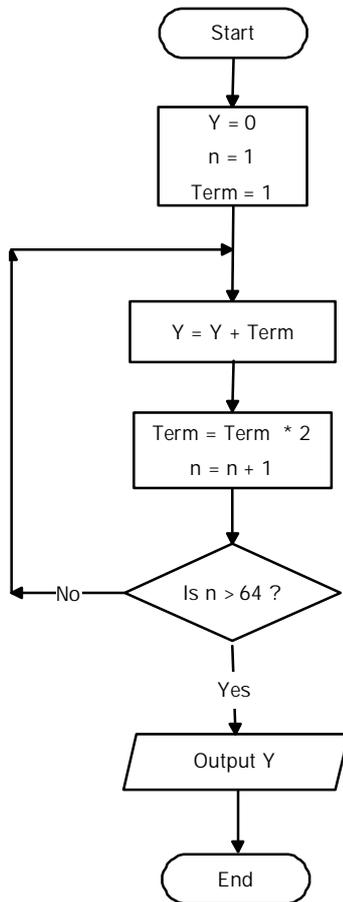
هي ضرب كل حد في 2 للحصول على قيمة الحد التالي له:

$$\text{Term} = \text{Term} * 2$$

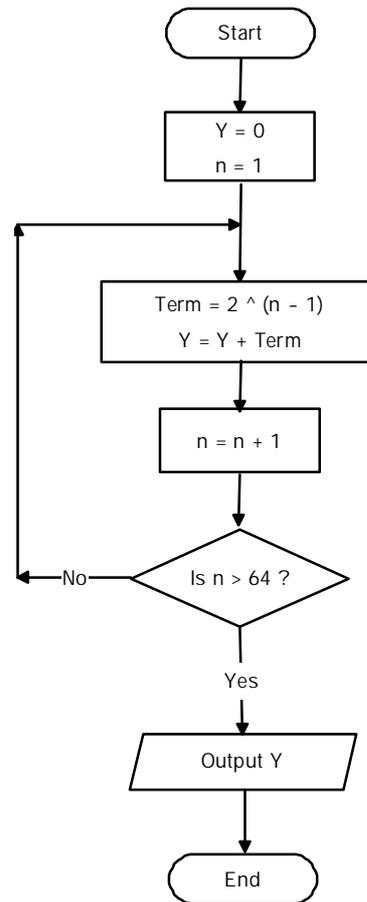
لاحظ أن الحد الأول يجب أن تكون قيمته 1، وهذا لا يتحقق بهذه المعادلة، لهذا سنجعل القيمة المبدئية للمتغير Term تساوي 1 (قيمة الحد الأول) وسنضيفها مباشرة إلى المتغير Y، وفي الخطوة التالية نحسب قيمة الحد التالي بالمعادلة السابقة.

استخدم الطريقة التي تراها أسهل.. هاتان هما خريطتا التدفق لكل منهما:

الطريقة الثانية



الطريقة الأولى



ملحوظة:

لو نفذت هذه الخريطة بكود فيجيوال بيزيك، فسيخبرك الناتج أن عدد حبات القمح المطلوبة هو ١٨٤٤٦٧٤٤٠٧٣٧٠٩٦٠٠٠٠٠٠ مليون حبة قمح، أي ما يزيد عن ١٨ مليون مليون حبة قمح!!.. ولو افترضنا أن وزن حبة القمح الواحدة هو جرام واحد، فهذا يعني أن من حق ذلك الحكيم الحصول على ١٨ مليون مليون طن من القمح!!

كم كان ذلك الملك سانجا، لأنه نظر إلى رقعة الشطرنج فلم يرَ إلا صغر مساحتها، وجعل أن الدوال الأسية دوال شرسة، وقيمها تتضاعف بسرعة مخيفة لتتجاوز أرقامها فلكية خلال عدد صغير من الحدود!

لاحظ أيضا أننا نعرف مضاعفات العدد ٢ بشكل خاص في عالم البرمجة والحاسب، لأنها تتدرج تحت نظام العد الثنائي، والمعروف أن ١٠ ضعفا من مضاعفات الاثنين تعطينا الرقم ١٠٢٤ وهو ما نعرفه في عالم الحاسب باسم الكيلو بايت، و ٢٠ ضعفا من مضاعفات الاثنين تعينا الرقم ١٠٤٨٥٧٦ وهو ما نعرفه في عالم الحاسب باسم الميجا بايت... وهكذا، كلما زاد عدد المضاعفات الثنائية بمقدار ١٠، لنحصل على الجيجا والتتيرا وهلم جرا.. هذا هو السبب الذي يجعل العدد يصل إلى قيمة فلكية عند الخانة رقم ٦٤.. الجديد بالذكر أن لوحة الشطرنج التي تحتوي على ٦٤ خانة، تقابل في متغيرات فيجيوال بيزيك العدد الطويل Long، الذي يتكون من ٤٦ خانة 64 Bits هو أيضا، كما سنرى فيما بعد بإذن الله.

س ١٠: ذات يوم كانت سيدة من قرية (برما) التابعة لمركز طنطا بمحافظة الغربية، تسير وهي تحمل سلة بيض، وفجأة اصطدم بها رجل فسقط البيض وانكسر.. سألها الرجل عن عدد البيض ليدفع لها ثمنه، فقالت له: لا أعرف عدد البيض بالضبط، لكن كل ما أعرفه هو:

- أنني إذا وزعته على سلتين تبقى بيضة زائدة.
- وإذا وزعته علي ثلاث سلال تبقى بيضة زائدة.
- وإذا وزعته علي أربع سلال تبقى بيضة زائدة.
- وإذا وزعته علي خمس سلال تبقى بيضة زائدة.

- وإذا وزعته علي ست سلال تتبقى بيضة زائدة.

- وإذا وزعته علي سبع سلال لا يتبقى أي شيء من البيض خارج السلال.

هل تستطيع رسم مخطط تدفق يحل هذه الحسبة التي أرهقت أهل (برما) برمتهم؟

ج: لم يكن أهل (برما) محظوظين بالتأكد لأنهم ليسوا مبرمجين، فالأمر يغدو في غاية البساطة باستخدام الحاسب!

الفكرة هنا هي أن نستخدم متغيرا اسمه n تبدأ قيمته بواحد، ونستمر في زيادة قيمته بواحد داخل جملة تكرار Loop، ولا نتوقف عن فعل هذا إلا إذا تحققت كل الشروط الستة التالية معا:

$$n \bmod 2 = 1, \quad n \bmod 3 = 1, \quad n \bmod 4 = 1$$

$$n \bmod 5 = 1, \quad n \bmod 6 = 1, \quad n \bmod 7 = 0$$

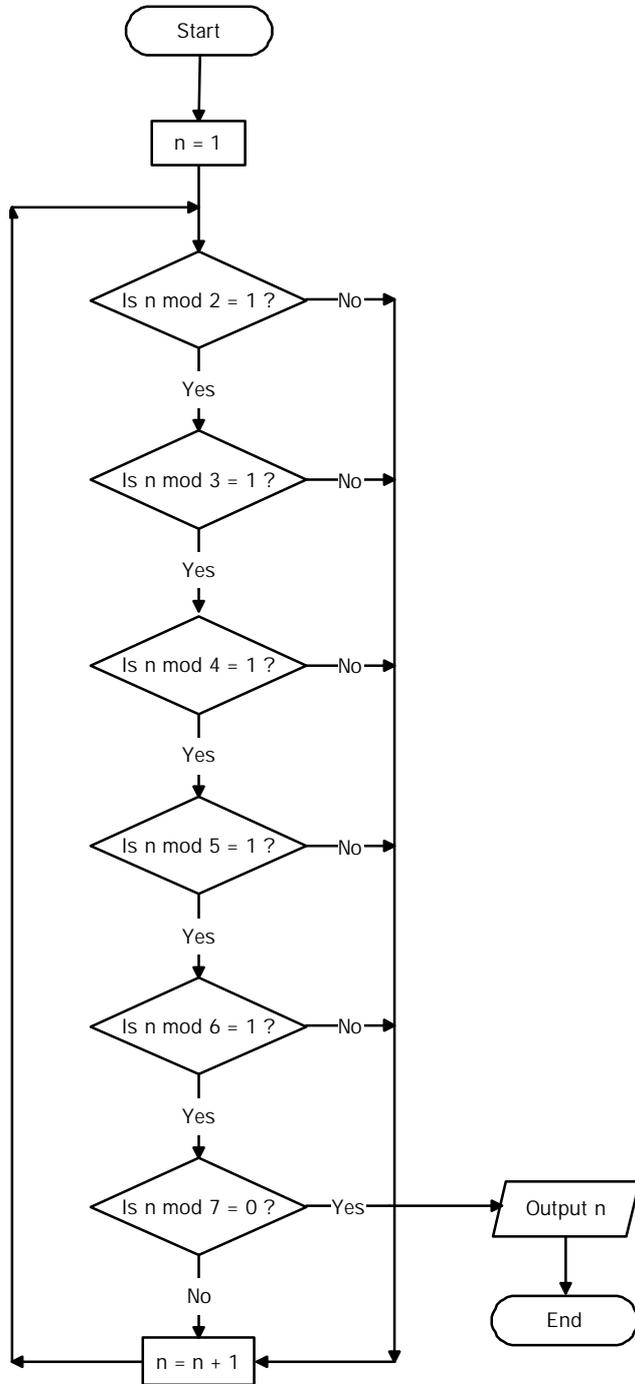
حيث \bmod تعني باقي القسمة.. فمثلا $n \bmod 2$ تعني الباقي من قسمة n على ٢.. وهكذا.

لاحظ أن فشل شرط واحد من هذه الشروط يكفي لمعرفة أن الرقم n لا يصلح لحل حسبة برما، لهذا فلا داعي لفحص باقي الشروط، وعلينا القفز مباشرة إلى السطر الذي نزيد فيه قيمة n بواحد لفحص رقم جديد.

أما إذا تحققت كل الشروط معا، فهذا يعني أن الرقم الموجود في المتغير n في هذه اللحظة هو عدد البيض، وعلينا عرضه على الشاشة وإنهاء البرنامج. والصورة التالية توضح هو مخطط تدفق حسبة (برما).

لاحظ أنك لو نفذت هذا المخطط بكود فيجيوال بيزيك، فستحصل في أقل من ثانية واحدة على ناتج يخبرك أن عدد البيض هو ٣٠١ بيضة.. هل تصدق أن هذه هي "حسبة برما" التي يضرب بها الناس المثل!!؟

وهناك ملحوظة أخرى هامة تتعلق بتحسين أداء البرنامج.. فما فعلناه هو إلقاء العبء كاملا على كاهل الحاسوب، دون أي تدخل من ذكائنا الإنساني، مع أننا بملاحظة بسيطة سنكتشف أن العدد الذي يحل حسبة (برما) لا بد أن يكون من مضاعفات الرقم ٧، لأن قسمة عدد البيض على ٧ هي الحالة الوحيدة التي لا يتبقى فيها باق!

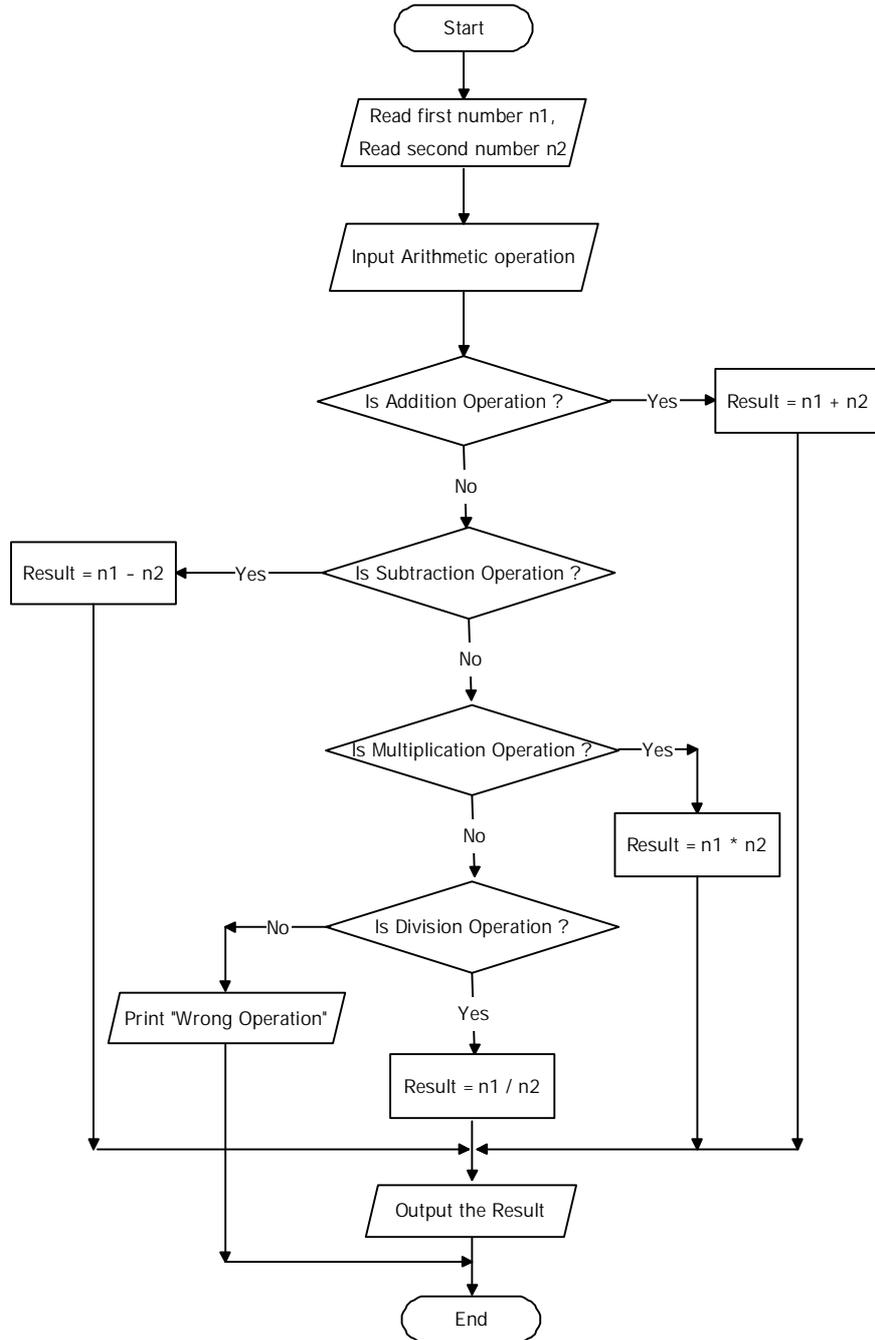


هذه الملحوظة البسيطة يمكن أن تساعدنا في تحسين سرعة البرنامج إلى سبعة أضعاف.. فبدلاً من أن نزيد قيمة المتغير n بواحد، سنجعل قيمته المبدئية ٧ ونزيده في كل مرة بمقدار ٧، ليكون باستمرار أحد مضاعفات العدد ٧.. أليس هذا أذكى ويوفر ٧/٦ من الوقت الذي يستهلكه البرنامج؟ سأترك لك تعديل خريطة التدفق للاستفادة من هذه الفكرة.

الفصل الثالث

الكود الزائف (سودو كود) Pseudo Code

س ١: اكتب الكود الزائف "سودو كود" لخريطة التدفق التالية:



ج: لو تأملت هذه الخريطة قليلا، لفهمت وظيفة هذا البرنامج:

- فهو يقرأ عددين، ويقرأ العملية الحسابية Arithmetic operation التي سيجريها على هذين العددين.
- ثم يفحص إن كانت العملية الحسابية عملية جمع Addition Operation، فإن كانت كذلك قام بجمع العددين وعرض الناتج وإنهاء البرنامج.
- وإن لم تكن عملية جمع، يفحص إن كانت عملية طرح Subtraction Operation، فإن كانت كذلك قام بطرح العددين وعرض الناتج وإنهاء البرنامج.
- وإن لم تكن عملية طرح، يفحص إن كانت عملية ضرب Multiplication Operation، فإن كانت كذلك قام بضرب العددين وعرض الناتج وإنهاء البرنامج.
- وإن لم تكن عملية ضرب، يفحص إن كانت عملية قسمة Division Operation، فإن كانت كذلك قام بقسمة العددين وعرض الناتج وإنهاء البرنامج.
- وإن لم تكن عملية قسمة، يعرض البرنامج رسالة للمستخدم تخبره أن العملية التي أدخلها خاطئة، وينتهي البرنامج.

هذا هو الكود الزائف "سودوكود" الذي يمثل هذه الخريطة:

- 1- Read the first number n1, and the second number n2
- 2- Input the arithmetic operation
- 3- IF the arithmetic operation is addition operation THEN
Result = n1 + n2
Go to 9
END IF
- 4- IF the arithmetic operation is subtraction operation THEN
Result = n1 - n2
Go to 9
END IF
- 5- IF the arithmetic operation is multiplication operation THEN
Result = n1 * n2
Go to 9
END IF
- 6- IF the arithmetic operation is division operation THEN
Result = n1 / n2
Go to 9
END IF

- 7- Print an output line Saying "Wrong operation."
- 8- Go to 10
- 9- Print an output line showing the Result
- 10- Stop processing

ملحوظة:

هناك خطأ محتمل في فكرة هذا البرنامج، وذلك لو أراد المستخدم إجراء عملية قسمة وأدخل قيمة العدد الثاني تساوي صفراً، فالقسمة على صفر غير ممكنة!.. لهذا من الأذكي تصحيح السوكود السابق لإضافة شرط قبل إجراء عملية القسمة، يتأكد أن قيمة المتغير n2 لا تساوي صفراً، كالتالي:

```
IF n2 = 0 THEN
    Print an output line Saying "Can't divide by zero."
    Go to 10
END IF
```

وسأترك لك إعادة رسم خريطة التدفق التي تراعي هذا التصحيح.

حل آخر:

يمكننا إعادة كتابة السوكود بشكل أفضل إذا استخدمنا جملة ELSE.. فبدلاً من كثرة استخدام Go to لتجاوز الشروط التالية والقفز مباشرة إلى سطر عرض الناتج في نهاية البرنامج، يمكننا الاستفادة من أن الكود المكتوب في المقطع ELSE لا يتم تنفيذه إلا إذا كان الشرط الموجود في جملة IF خاطئاً.. هذا هو السوكود الجديد:

س٣: اكتب الكود الزائف "سودوكود" لعملية قراءة درجة الحرارة بالفهرنهايت (F°) ثم تحويلها إلى ما يقابلها بالدرجات المئوية.

ج: لقد رسمنا خريطة التدفق لهذا المثال في إجابة السؤال الثالث في الفصل السابق.. وهذا هو السودوكود المناظر لها:

- 1- Read the Fahrenheit degree, F
- 2- Calculate the centigrade degree, C:
 $C = 5 / 9 * (F - 32)$
- 3- Print an output line showing the centigrade degree, C
- 4- Stop processing

س٤: اكتب الكود الزائف "سودوكود" لعملية قراءة نصف قطر كره (r)، ثم إيجاد حجم الكره (V)، ومساحة سطحها (A).

ج: لقد رسمنا خريطة التدفق لهذا المثال في إجابة السؤال الخامس في الفصل السابق.. وهذا هو السودوكود المناظر لها:

- 1- Read the Radius of the sphere
- 2- IF the value of the Radius is less than zero THEN
Print "Radius can't be a negative number"
Go to 6
END IF
- 3- Calculate the area of the sphere:
 $A = 4 * (22 / 7) * r ^ 2$
- 4- Calculate the Volume of the sphere:
 $V = (4 / 3) * (22 / 7) * r ^ 3$
- 5- Print an output line showing the Area (A) and the Volume (V)
- 6- Stop processing

س٥: اكتب الكود الزائف "سودوكود" لقراءة المتغير x وحساب الدالة y، المعطاة بالصيغة الرياضية:

$$y = x - \frac{x^3}{3} + \frac{x^5}{5} - \frac{x^7}{7} + \dots$$

و ذلك بجمع أول ١٠٠ حد.

ج: لقد رسمنا خريطة التدفق لهذا المثال في إجابة السؤال السادس في الفصل السابق..
وهذا هو السودوكود المناظر لها:

- 1- Read the value of X
 - 2- Initialize the values of the result of function Y, the term counter C, the factor n, and the Sign of the term:
Y = 0
C = 1
n = 1
Sign = 1
 - 3- Calculate the value of the current term:
Term = Sign * (1 / n) * (X ^ n)
 - 4- Add the term to the result of function Y:
Y = Y + Term
 - 5- Increase the term counter by 1, and the term factor by 2, to move to the next term:
C = C + 1
n = n + 2
 - 6- Reverse the sign of the next Term
Sign = Sign * - 1
 - 7- IF the term counter (C) is greater than 100 THEN
Go to 8
Else
Perform steps 3 through 6 until the term counter (c) is greater than 100
END IF
 - 8- Print an output line showing the value of Y
 - 9- Stop processing
-

س٦: قارن بين خريطة التدفق والكود الزائف "سودوكود".

ج: الجدول التالي يوضح أوجه المقارنة بين خريطة التدفق والسودوكود.

وجه المقارنة	خريطة التدفق	السودوكود
التعريف	هي تمثيل بياني يعتمد على الرسم لتوضيح ترتيب العمليات اللازمة لحل المسألة.	هو طريقة كلامية للتعبير عن خطوات حل مسألة معينة باستخدام اللغة الإنجليزية العادية، لكن بدون استخدام كود حقيقي من أكواد لغات البرمجة المعروفة.
الوضوح	أسهل في فهمها بمجرد النظر.. لكن في بعض الأحيان تصير خريطة التدفق معقدة بسبب تداخل الأسهم وكثرة المسارات.	تجب قراءة السودو الكود كاملاً لاستيعابه، لكنه في بعض الأحيان يكون أوضح من خريطة التدفق، لأنه لا يحتوي على أية أسهم أو مسارات متداخلة كما في خرائط التدفق.
سهولة الكتابة	أصعب قليلاً، بسبب الحاجة إلى رسم الرموز والأشكال، وتنظيم الرسوم.	أسهل في كتابته، لأنه مجرد كلام انجليزي عادي بدون قواعد محددة.
الطول	أكثر اختصاراً، لأنها تستخدم الرموز للتعبير.	أكثر إسهاباً وطولاً، بسبب استخدام جمل كاملة للتعبير.
التعديل	أصعب، حتى في وجود برامج رسوم على الحاسب تسهل رسم خريطة التدفق وتعديلها، بسبب الحاجة إلى تنسيق الرموز والأشكال يدوياً.	أسهل بكثير، حيث يمكنك كتابته على أي برنامج محرر نصوص، ولا يحتاج إلى أية تنسيقات معقدة أو مضيعة للوقت والجهد.

<p>١- سهل الفهم، لأنه يستخدم كلمات إنجليزية بسيطة.</p> <p>٢- قريب في صياغته من لغات البرمجة، لهذا يسهل تحويله إلى أية لغة برمجة.</p> <p>٣- لا يحتوي على أشكال ورسوم خاصة، وليست له قواعد صعبة خاصة به، لهذا يكون أسهل وأسرع في كتابته.</p> <p>٤- يسهل تعديله وتطويره.</p>	<p>١- وسيلة اتصال سهلة، لنقل الأفكار البرمجية وشرح حلول المسائل للآخرين.</p> <p>٢- وسيلة فعالة وواضحة لتحليل المسألة واكتشاف الحلول.</p> <p>٣- أداة هامة لتوثيق البرامج وحفظها على الورق.</p> <p>٤- تحقق كفاءة أعلى للمبرمج عند كتابة الكود، حيث تفصل حل المشكلة عن كتابة الكود، مما يبسط كتابة الكود.</p> <p>٥- تسهل تتبع خطوات الحل، مما يساعد على كشف أية أخطاء موجودة فيه.</p> <p>٦- تسهل إصلاح البرنامج.</p>	<p>المزايا</p>
<p>قد يكون طويلا جدا في المسائل المعقدة.</p>	<p>١- إذا كانت المسألة معقدة، يصير المخطط معقدا وتصبح متابعته.</p> <p>٢- قد يحتوي المخطط على الكثير من التفاصيل، مما يجعل تتبع حل بعض المسائل صعبا.</p> <p>٣- يتطلب إجراء تعديلات في الفكرة إعادة رسم المخطط من جديد.</p> <p>٤- صعوبة نسخ المخططات.</p>	<p>العيوب</p>

س٧: اكتب الكود الزائف "سودوكود" لحل حسة برما التي شرحناها في السؤال العاشر في الفصل السابق.

ج: لقد رسمنا خريطة التدفق لهذا المثال في إجابة السؤال العاشر في الفصل السابق.. وهذا هو السودوكود المناظر لها:

1- Initialize the number n:

$n = 1$

2- IF the remainder of dividing n by 2 equals 1 THEN

 IF the remainder of dividing n by 3 equals 1 THEN

 IF the remainder of dividing n by 4 equals 1 THEN

 IF the remainder of dividing n by 5 equals 1 THEN

 IF the remainder of dividing n by 6 equals 1 THEN

 IF the remainder of dividing n by 7 equals 0 THEN

 Print an output line showing the value of n

 Stop processing

 END IF

 END IF

 END IF

 END IF

 END IF

END IF

3- Increase the number by 1:

$n = n + 1$

4- Perform steps 2 through 3 continuously

لاحظ أننا استخدمنا عدة جمل شرط متداخلة Nested IF.. هذا الشكل قد لا يروق لك وقد تظنه مرهقا في كتابته.. لا تقلق، ففي فيجيوال بيزيك هناك طرق لدمج أكثر من شرط في سطر واحد لاختصار كتابة الكود، كما سنتعلم فيما بعد بإذن الله.. وعلى كل حال لو لم تكن تريد استخدام جمل الشرط المتداخلة، فيمكننا إجراء تغيير طفيف في السودوكود، وذلك بعكس الشروط الخمسة الأولى.. فبدلا من يسأل الشرط إن كان باقي القسمة يساوي ١، سنجعل الشرط يسأل إن كان باقي القسمة لا يساوي ١، وبذلك نستطيع استخدام الجملة Go to للقفز مباشرة إلى سطر زيادة قيمة العدد n في نهاية جملة التكرار Loop.. هذا هو السودوكود المعدل، وهو أيضا يستخدم طريقة أكثر كفاءة في الحل، وذلك بجعل القيمة المبدئية للعدد n تساوي ٧ وزيادة ٧ على قيمة n في كل لفة:

- 1- Initialize the number n to 7:
 $n = 7$
- 2- IF the remainder of dividing n by 2 doesn't equal 1 THEN
 Go to 8
 END IF
- 3- IF the remainder of dividing n by 3 doesn't equal 1 THEN
 Go to 8
 END IF
- 4- IF the remainder of dividing n by 4 doesn't equal 1 THEN
 Go to 8
 END IF
- 5- IF the remainder of dividing n by 5 doesn't equal 1 THEN
 Go to 8
 END IF
- 6- IF the remainder of dividing n by 6 doesn't equal 1 THEN
 Go to 8
 END IF
- 7- IF the remainder of dividing n by 7 equals 0 THEN
 Go to 10
 END IF
- 8- Increase the number by 7:
 $n = n + 7$
- 9- Perform steps 2 through 8 continuously
- 10- Print an output line showing the value of n
- 11- Stop processing

وأنت حر في استخدام الطريقة التي تتاسبك أكثر، وتراها أكثر سهولة.

س٨: اكتب الكود الزائف "سودوكود" لحل مسألة ملء رقعة الشطرنج بحبوب القمح التي شرحناها في السؤال التاسع في الفصل السابق.

ج: لقد رسمنا خريطة التدفق لهذا المثال في إجابة السؤال التاسع في الفصل السابق.. وهذا هو السودوكود المناظر لطريقة الحل الأولى:

Initialize the values of the result Y and the term number n:

$$Y = 0$$

$$n = 1$$

1- Calculate the value of the nth term:

$$\text{Term} = 2^{(n - 1)}$$

2- Add the term to the result Y:

$$Y = Y + \text{Term}$$

3- Increase the term number by 1:

$$n = n + 1$$

4- IF the term number (n) is greater than 64 THEN

Go to 6

Else

Perform steps 2 through 4 until the term number (n) is greater than 100

END IF

5- Print an output line showing the value of Y

6- Stop processing

وسأترك لك كتابة السودوكود الخاص بالطريقة الثانية على سبيل التدريب.

الفصل الرابع

مقدمة عن البرمجة الشيئية (الكائنية)

Object Oriented Programming (OOP)

س ١: ضع العلامة (√) أمام العبارة الصحيحة والعلامة (×) أمام العبارة الخاطئة، مع تصحيح الخطأ:

- ١ - يتم إنشاء التصنيف (Class) من الكائن (Object).
- ٢ - يعتبر جهاز الحاسب كائنا (Object) مكونا من عدة كائنات.
- ٣ - يعتبر لون القلم إحدى وظائف القلم (Methods).
- ٤ - تشغيل جهاز الكاسيت يعتبر حدثا (Event).
- ٥ - تحدد خصائص الكائن (Properties) شكل وسمه الكائن.
- ٦ - يتميز الكائن بأن له خصائص (Properties) ووظائف (Methods) وأحداثا (Events).
- ٧ - يستخدم التغليف (Encapsulation) لإخفاء بيانات الفئة (Class).
- ٨ - عندما ترث فئة (Class) فئة أخرى فإنها ترث الخصائص فقط.
- ٩ - من أهداف التغليف (Encapsulation) حماية بيانات التصنيف (Class).
- ١٠ - تعد البرمجة الشيئية OOP برمجة تقليدية، تتم في خطوات محددة متتالية.
- ١١ - الكائن Object هو متغير يتم تعريفه من الفئة Class.
- ١٢ - يمكن تعريف كائن واحد فقط من كل التصنيف Class.
- ١٣ - استخدام الكائنات يوفر الوقت والجهد ويضمن أقصى استفادة من الكود.
- ١٤ - معالج الحدث Event Handler هو جزء من البرنامج يتم تنفيذه عند انطلاق حدث معين.
- ١٥ - الفئة Class هي نوع الكائن Object.
- ١٦ - الكائنات التي لها نفس النوع تمتلك نفس الخصائص، لهذا تتشابه تماما في شكلها ووظيفتها.
- ١٧ - الحدث هو تنبيه يطلقه الكائن لإخبار المبرمج بأن فعلا معيناً قد وقع.

١٨ - تسمح فيجوال بيزيك دوت نت للفئة بالوراثة من أكثر من فئة أخرى بطريقة مباشرة.

١٩ - يمكن أن يكون هناك أكثر من تصنيف أب Parent Class لنفس التصنيف الابن Child Class.

٢٠ - يمكن أن يكون هناك أكثر من تصنيف ابن Child Class لنفس التصنيف الأب Parent Class.

٢١ - يتيح لك التغليف Encapsulation إعادة استخدام ما تم تصميمه من فئات، والتعديل فيها حسب الحاجة، بدلا من إعادة كتابة كل الكود من جديد.

٢٢ - التغليف يجعل استخدام مشغل البرنامج للكائن أسهل.

ج: ١ - (×) الصواب: يتم إنشاء الكائن (Object) من التصنيف (Class).
٢ - (√)

٣ - (×) الصواب: يعتبر لون القلم إحدى خصائص القلم (Properties).

٤ - (×) الصواب: تشغيل جهاز الكاسيت يعتبر وسيلة (Method).
٥ - (√)

٦ - (√)

٧ - (√)

٨ - (×) الصواب: عندما ترث فئة (Class) فئة أخرى فإنها ترث الخصائص والوسائل والأحداث.

٩ - (√)

١٠ - (×) الصواب: تعد البرمجة الشيئية برمجة حديثة، تعتمد على الكائنات وكتابة الكود في معالجات الأحداث للتفاعل مع تصرفات مستخدم البرنامج.

١١ - (√)

١٢ - (×) الصواب: يمكن تعريف عدد لا حصر له من الكائنات من كل فئة.

١٣ - (√)

١٤ - (✓)

١٥ - (✓)

١٦ - (×) الصواب: الكائنات التي لها نفس النوع تمتلك نفس الخصائص، لكنها قد تختلف في شكلها وسلوكها بسبب اختلاف قيم هذه الخصائص في كل كائن منها عن الآخر.

١٧ - (✓)

١٨ - (×) الصواب: تسمح فيجيوال بيزيك لكل فئة بالوراثة من فئة واحدة فقط بطريقة مباشرة.. يسمى هذا بالوراثة المنفردة Single Inheritance، وهو يهدف إلى تقليل التعقيد ومنع أية أخطاء محتملة بسبب تداخل عمليات الوراثة.

١٩ - (✓).. تنويه: يحدث هذا في فيجيوال بيزيك بطريقة غير مباشرة.. افترض أن لدينا ثلاثة تصنيفات أسماؤها Class1 و Class2 و Class3.. لو ورت التصنيف Class2 التصنيف Class1، وورث التصنيف Class3 التصنيف Class2، ففي هذه الحالة يعتبر التصنيف Class3 ابنا لكل من Class2 (بطريقة مباشرة) و Class1 (بطريقة غير مباشرة).

٢٠ - (✓)

٢١ - (×) الصواب: تتيح لك الوراثة Inheritance إعادة استخدام ما تم تصميمه من فئات، والتعديل فيها حسب الحاجة، بدلا من إعادة كتابة كل الكود من جديد، أما التغليف فيهدف إلى حماية بيانات الفئة، وتسهيل استخدام الكائن.

٢٢ - (✓) تنويه: التغليف يخفي عن المستخدم التفاصيل التي لا تهمه، مما يجعل استخدامه للكائن أسهل.

س٢: أكمل مكان النقط مستخدماً الكلمات الآتية لجعل الجملة صحيحة:

Object – Event – Property – Method – Class – Encapsulation – Inheritance

- ١ - يعتبر الضغط بمؤشر الفأرة داخل النافذة
- ٢ - يعتبر ارتفاع النافذة
- ٣ - الضغط على مفتاح من لوحة المفاتيح داخل النافذة يعتبر
- ٤ - الخصائص والوظائف التي يأخذها الابن من أبيه تُعتبر
- ٥ - الذهاب إلى المدرسة يعتبر
- ٦ - تُعتبر كمية المياه الموجودة في بحيرة
- ٧ - الرسم التخطيطي لجهاز التسجيل الموجود بكتيب التشغيل يعتبر
- ٨- يعتبر إخفاء للبيانات داخل التصنيف (Class).
- ٩- ضغط زر تشغيل التسجيل يعتبر بينما بدء التسجيل في العمل يعتبر
- ١٠ - أوجه الشبه بين نافذة مربع الرسالة Message Box Window، ونافذة دخول البرنامج Login Window سببه

ج:

- | | | | |
|------------|------------------|-----------|-----|
| ١ - Event | ٢ - Property | ٣ - Event | ٤ - |
| ٥ - Method | ٦ - Property | ٧ - Class | ٨ - |
| ٩ - Method | ١٠ - Inheritance | | |
- Inheritance
- Encapsulation
- Event

ملحوظة بخصوص النقطة رقم ٥ :

الذهاب إلى المدرسة يعتبر

بعض الأفعال تسبب ارتباكاً في تصنيفها كوسيلة Method أو حدث Event، وفي البرمجة، ما يميز بين الوسيلة والحدث هو تصميم المبرمج للفئة، فالحقيقة أن معظم الوسائل تؤدي إلى انطلاق أحداث في الفئة.. فمثلاً يؤدي استدعاء الوسيلة Close الخاصة بالنموذج إلى انطلاق الحدثين FormClosing و FormClosed على التوالي، وبالتالي لو قلت لك: إغلاق النموذج يعتبر فإنك تستطيع أن تجيب بأنه وسيلة أو أنه حدث!!.. كذلك فإن ذهاب التلميذ إلى المدرسة هو وسيلة للتعلم، أو حدث قام به التلميذ، ولن يميز بينهما في البرمجة إلا الطريقة التي ستصمم بها فئة التلميذ Student Class، فلو وضعت بها دالة Function اسمها GotoSchool فسيكون الذهاب إلى المدرسة وسيلة Method، أما إذا وضعت بها حدثاً اسمه GoingToSchool فسيكون الذهاب إلى المدرسة حدثاً Event!.. لمثل هذا من الأفضل تجنب مثل هذه النوعية المربكة من الأمثلة العامة، فالأكواد البرمجية أكثر دقة ووضوحاً منها.. لكن عموماً، دعنا نتفق على اعتبار الوظائف التي يقوم بها الكائن ذاتياً (مثل تشغيل التسجيل أو ذهاب الطالب إلى المدرسة) وسائل Methods، بينما سنعتبر الوظائف التي يؤثر بها مستخدم خارجي على الكائن (كتحريك مؤشر الفأرة أو ضغط أزرار لوحة المفاتيح) أحداثاً Events.

س٣: عرف كلا من المصطلحات التالية:

Object - Event - Property - Method - Class - Object - Encapsulation - Inheritance.

ج: البرمجة الشيئية (Object Oriented Programming (OOP):

نوع من البرمجة الحديثة، يستخدم الكائنات Objects لأداء وظائف البرنامج المختلفة.. ولا يتبع تنفيذ البرنامج التسلسل المنطقي، حيث يتم تقسيم البرنامج إلى أجزاء صغيرة تسمى معالجات الأحداث Event Handlers، يتم تنفيذ كل منها كاستجابة لحدث معين من الأحداث التي يقوم بها المستخدم.

الخاصية Property:

صفة من صفات الكائن تحدد شكله وسمته.. ولكل كائن مجموعة من الخصائص.

الوسيلة Method:

هي سلوك معين أو وظيفة معينة يقوم بها الكائن.

الحدث Event:

هو فعل يقع على الكائن ويستجيب له.

الفئة أو التصنيف Class:

مقطع من الكود يستخدم لتعريف نوع جديد مستقل بذاته، يحتوي على خصائص ووسائل وأحداث، ويمكن تعريف أي عدد من الكائنات منه.

الكائن Object:

الكائن بشكل عام، هو شيء له وجود، ويمتلك مجموعة من الخصائص Properties، والوسائل Methods، والأحداث Events، ويمكن أن يتكون من مجموعة من الكائنات الأصغر، أو يكون جزءا من كائن أكبر. والكائن في البرمجة هو متغير Variable تم تعريفه من الفئة التي تمثل نوعه.. أو هو نسخة Instance تم إنشاؤها من الفئة.

التغليف Encapsulation:

هو إخفاء البيانات داخل الكائن، بحيث لا يتم الوصول إليها إلا بصلاحيات معينة.

الوراثة Inheritance:

يقصد بها امتلاك فئة مشتقة Derived Class كل خصائص ووسائل وأحداث الفئة الأساسية Base Class، بالإضافة إلى امتلاك الفئة المشتقة بعض العناصر الإضافية الخاصة بها. وفائدة الوراثة هي إعادة استخدام ما تم تصميمه من فئات، والتعديل فيها حسب الحاجة، بدلا من إعادة كتابة كل الكود من جديد.

الفصل الخامس

بيئة التطوير المتكاملة IDE

س ١: ما معنى المصطلح "IDE"؟

ج: الحروف IDE هي اختصار التعبير Integrated Development Environment بمعنى "بيئة التطوير المتكاملة"، وهي البرنامج الذي نستخدمه للتعامل مع مشاريع فيجيوال بيزيك دوت نت، والذي يقدم للمبرمج مجموعة كبيرة من الأدوات لتصميم النماذج وما عليها من أدوات تحكم Controls، وكتابة كود البرنامج وتجربته.

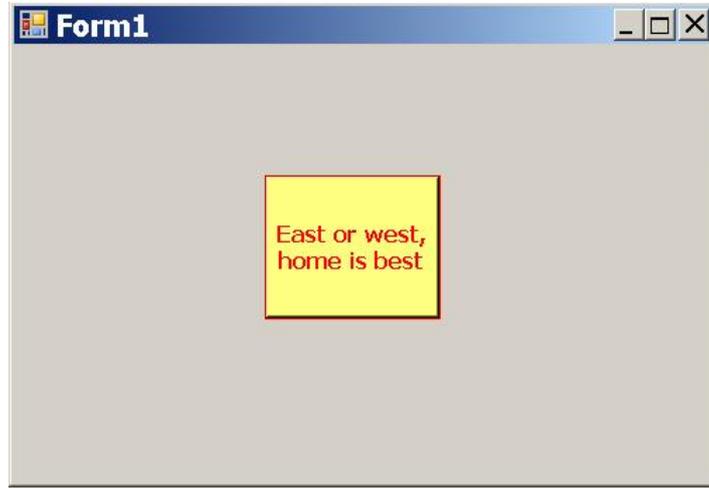
س ٢: اذكر استخدامات نافذة الخصائص **properties window** وصندوق الأدوات **.toolbox**.

ج: تستخدم نافذة الخصائص **Properties window** لعرض خصائص النموذج والأدوات الموضوعه عليه وخصائص ملفات المشروع، وبهذا يستطيع المبرمج تغيير قيم هذه الخصائص بشكل مرئي سهل وسريع أثناء تصميم البرنامج. ويستخدم صندوق الأدوات **Toolbar** لعرض كل الأدوات Controls التي يمكن للمبرمج رسمها على النموذج واستخدامها في برنامجها، وذلك بمجرد سحبها من صندوق الأدوات وإلقائها على النموذج، وهي مبنية تحت شرائط Tabs على حسب نوعها.

س ٣: قم ببناء واجهة للمستخدم **User Interface** بها زر واحد فقط اسمه **BtnTest**، وعند النقر على الزر يجب أن يصير لون خلفيته أصفر، ولون الكتابة عليه أحمر، ويظهر عليه النص:

East or west, home is best

كما هو مبين في الصورة.



ج: اتبع الخطوات التالية:

- اضغط القائمة الرئيسية "ملف" File أعلى نافذة بيئة التطوير IDE، واضغط الأمر "مشروع جديد" New Project.. سيؤدي هذا إلى عرض مربع حوار "مشروع جديد" New Project.
- من القائمة الموجودة على اليسار اختر Visual Basic من لغات البرمجة المتاحة في دوت نت.
- من القائمة الموجودة على اليمين اختر النوع "تطبيق ويندوز" Windows Application من بين أنواع المشاريع التي يمكن إنشاؤها بفيجيوال بيزيك.
- في خانة اسم المشروع Name أسفل النافذة، اكتب اسم هذا المشروع، وليكن MyButton.
- اضغط زر موافق OK، لإغلاق مربع الحوار وإنشاء المشروع.
- من القائمة File اضغط Save All، وفي نافذة الحفظ حدد المسار الذي تريد حفظ البرنامج عليه، واضغط الزر OK.
- من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.
- اسحب الزر بالفأرة لوضعه في منتصف النموذج.

- اسحب أطراف الزر بالفأرة ليصير عرضه وارتفاعه كما هو موضح في الصورة السابقة.
- حدد الزر واضغط F4 لعرض نافذة الخصائص.. غير اسم الزر Name إلى BtnTest.
- انقر الزر مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnTest_Click.. في هذا المعالج اكتب الكود التالي:

BtnTest.Text = "East or west, home is best"

BtnTest.BackColor = Color.Yellow

BtnTest.ForeColor = Color.Red

حيث إن:

- الخاصية Text هي المسئولة عن قراءة أو تغيير النص المعروض على الزر.
- الخاصية BackColor هي المسئولة عن قراءة أو تغيير لون خلفية الزر.
- الخاصية ForeColor هي المسئولة عن قراءة أو تغيير لون النص المكتوب على الزر.
- Color.Yellow كائن يمثل اللون الأصفر.
- Color.Red كائن يمثل اللون الأحمر.

س ٤: عرف كلا من:

**Form Designer - Code Editor Error List - Object Browser –
Help Window - IntelliSense - SnapLines**

ج: مصمم النماذج **Form Designer**: هو جزء من بيئة التطوير المتكاملة IDE، يساعد المبرمج في تصميم النموذج Form والأدوات الموجودة عليه باستخدام نافذة الخصائص وصندوق الأدوات.

محرر الكود Code Editor: هو جزء من بيئة التطوير المتكاملة IDE، يساعد المبرمج في كتابة الأوامر وتصحيح أخطاء صياغة الكود Syntax Errors.

قائمة الأخطاء Error List: هي نافذة تظهر في أسفل بيئة التطوير IDE، تعرض أخطاء الصياغة Syntax Errors الموجودة في الكود، كما تعرض بعض التحذيرات Warnings للمبرمج.

متصفح الكائنات Object Browser: هو جزء من بيئة التطوير المتكاملة IDE، يعرض الفئات Classes المستخدمة في البرنامج.

نافذة الاستعلام والمساعدة Help: هي نافذة تحتوي على كل المعلومات اللازمة التي تشرح مكونات بيئة التطوير IDE وأوامر فيجيوال بيزيك وكيفية استخدامها.. ويمكن عرضها بضغط الزر F1.

الاستشعار الذكي IntelliSense: هي تقنية تساعد المبرمج أثناء كتابة الكود في إكمال أسماء المتغيرات وأسماء الأدوات تلقائياً، وعرض ملخص عن وظيفة كل منها وكيفية استخدام الخصائص والوسائل.

خطوط المحاذاة SnapLines: هي خطوط تظهر على النموذج عند تحريك أية أداة، تصل بين حافتها وحافة الأداة المحاذاية لها رأساً أو أفقياً، وبهذا تسهل على المبرمج تنسيق الأدوات والحصول على أجمل تصميم.

الفصل السادس

مقدمة إلى الأدوات Controls

س ١: أ- هل كتابة النص "btnCalculate" في الخاصية Text الخاصة بزر تعتبر فكرة جيدة؟.. اشرح السبب.

ب - هل هناك أي معنى لجعل قيمة الخاصية Text الخاصة بالأداة Label سلسلة حروف فارغة (نصا فارغا) "empty string"؟.. اشرح السبب.

ج - اذكر مميزات استخدام الدالة MsgBox عن الأداة Label.

ج: أ- لا تعتبر كتابة النص "btnCalculate" في الخاصية Text الخاصة بالزر Button فكرة جيدة، لأن هذا النص سيظهر على الزر وسيراه المستخدم، وهو ليس مألوفا كعنوان بالنسبة للمستخدم، لهذا من الأفضل أن نكتب في الخاصية Text نصا واضحا ومألوفا يشرح وظيفة الزر، مثل Calculate.. أما النص "btnCalculate" فيصلح لنكتبه في الخاصية Name ليصير هو اسم الزر ونستخدمه في كتابة الكود، لأن البادئة btn ستلفت نظرنا إلى أن هذا الاسم خاص بزر، والمقطع Calculate يوضح وظيفة الزر وأنه يستخدم في إجراء عملية حسابية.

ب- في بعض الحالات نحتاج إلى جعل قيمة الخاصية Text الخاصة باللافتة Label نصا فارغا "empty string"، حتى تظهر اللافتة فارغة وليس فيها أي نص.. يحدث هذا عندما نستخدم اللافتة لعرض بعض نتائج البرنامج، ففي هذه الحالة يجب أن تكون اللافتة فارغة في البداية، ولا تعرض الناتج إلا عند ضغط الزر الذي يؤدي إلى عرض النتيجة.. وتسمى اللافتة في هذه الحالة لافتة متغيرة "Dynamic Label".. أما اللافتات التي تعرض نصا ثابتا لا يتغير طيلة البرنامج ليشرح وظيفة أداة أخرى، فتسمى لافتة ثابتة "Static Label"، ويجب وضع عنوانها في الخاصية Text أثناء تصميم البرنامج.

ج- يفضل استخدام الدالة MsgBox بدلاً من اللافتة Label إذا كان من الضروري أن نجبر مشغل البرنامج على قراءة الرسالة.. مثلاً: عند حدوث خطأ في البرنامج، فإن المبرمج يحتاج أن يخبر المشغل User بحدوث هذا الخطأ، ولا يدعه يستمر في التشغيل حتى يتأكد من أنه قرأ الرسالة وأخذ علماً بالمشكلة التي حدثت.. في هذه الحالة إذا تم عرض رسالة الخطأ في اللافتة Label، فإن المستخدم له الحرية في قراءة أو عدم قراءة الرسالة.. أما عند استخدام الدالة MsgBox فإنها تعرض الرسالة في نافذة خاصة على الشاشة، تجبر البرنامج على التوقف عن التشغيل أثناء عرضها، ولا يعود إلى حالة التشغيل إلى أن يستجيب المستخدم للمطلوب منه في الرسالة، بضغط الزر المناسب على مربع الرسالة ليتم إغلاقه.

س ٢: قم بإنشاء واجهة مستخدم بها زران "2 Buttons" ولافتة ثابتة "Static Label" ولافتة متغيرة "Dynamic Label" تعرض قيمه ابتدائية عبارة عن سلسلة حروف فارغة (نص فارغ) "Empty String".. والجدول التالي يظهر الأدوات المطلوب وجودها على النموذج و الخواص المناظرة لها.

Control	Property	Value
First Button	Name Text	btnYes Yes
Second Button	Name Text	btnNo No
Static Label	Text	Are you happy?
Dynamic Label	Name Text	lblReply ""

عند الضغط على الزر "Yes" تصير خاصية النص "Text" للمبين "lblReply" مساوية للآتي:

"There is no rose without a thorn."

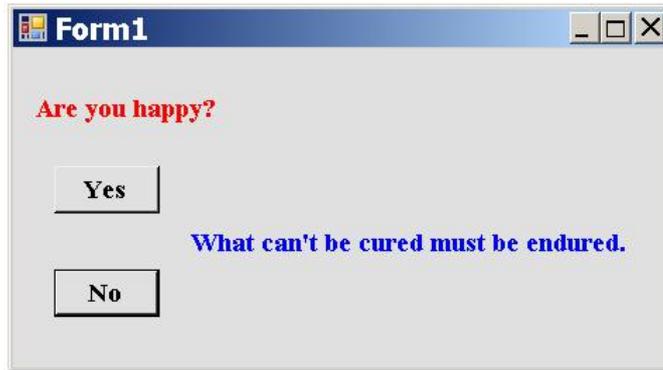
بمعنى: لا توجد وردة بلا شوك.

وعند الضغط على الزر "No" تصير خاصية النص "Text" للمبين "lblReply" مساوية لـ "لا":

"What can't be cured must be endured."

بمعنى: ما لا يمكن علاجه يجب احتماله.

ويجب أن يبدو النموذج عند الضغط على الزر "No" كما هو موضح في الصورة.



ج: اتبع الخطوات التالية:

1. اضغط القائمة الرئيسية "ملف" File أعلى نافذة بيئة التطوير IDE، واضغط الأمر "مشروع جديد" New Project.. سيؤدي هذا إلى عرض مربع حوار "مشروع جديد" New Project.
2. من القائمة الموجودة على اليسار اختر Visual Basic من لغات البرمجة المتاحة في دوت نت.
3. من القائمة الموجودة على اليمين اختر النوع "تطبيق ويندوز" Windows Application من بين أنواع المشاريع التي يمكن إنشاؤها بـ فيجيوال بيزيك.
4. في خانة اسم المشروع Name أسفل النافذة، اكتب اسم هذا المشروع، وليكن Question.
5. اضغط زر موافق OK، لإغلاق مربع الحوار وإنشاء المشروع.
6. من القائمة File اضغط Save All، وفي نافذة الحفظ حدد المسار الذي تريد حفظ البرنامج عليه، واضغط الزر OK.

٧. من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.. كرر ذلك مرة أخرى لإضافة زر آخر.

٨. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة اللافتة Label لإضافتها إلى النموذج.. كرر ذلك مرة أخرى لإضافة لافتة أخرى.

٩. حدد الزر الأول بضغطة مرة واحدة بزر الفأرة واضغط F4 لعرض نافذة الخصائص.. استخدم الخاصية Name لتغيير اسم الزر إلى BtnYes، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى Yes.

١٠. حدد الزر الثاني وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسم الزر إلى BtnNo، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى No.

١١. حدد اللافتة الأولى وفي نافذة الخصائص استخدم الخاصية Text لتغيير النص الذي تعرضه اللافتة إلى:

Are you happy?

واستخدم الخاصية ForeColor لتغيير لون النص المكتوب في هذه اللافتة إلى اللون الأحمر.

لاحظ أننا لا نحتاج إلى تغيير اسم هذه اللافتة، لأنها لافتة ثابتة القيمة Static Label ولن نستخدم اسمها في الكود.

١٢. حدد اللافتة الثانية وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسم هذه اللافتة إلى lblReply، واستخدم الخاصية Text لحذف النص المعروض على اللافتة.. كل ما عليك هو حذف النص الموجود في خانة القيمة المجاورة لهذه الخاصية وتركها فارغة.. لاحظ أنك لو كتبت "" فستظهر علامتا التنصيص على اللافتة، فعلامتا التنصيص لا تستخدمان في نافذة الخصائص، بل تستخدمان في الكود فقط للتفريق بين النصوص العادية وأسماء المتغيرات.

استخدم أيضا الخاصية ForeColor لتغيير لون النص المكتوب في هذه اللافتة إلى اللون الأزرق.

١٣. انقر الزر Yes مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnYes_Click.. في هذا المعالج اكتب الكود التالي:

lblReply.Text = "There is no rose without a thorn."

١٤. عد إلى مصمم النموذج وانقر الزر No مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnNo_Click.. في هذا المعالج اكتب الكود التالي:

lblReply.Text = "What can't be cured must be endured."

س٣: قم بإنشاء مشروع فيجوال بيزيك دوت نت يحتوى على نموذج ومربع نص TextBox ولافتة Label.. اكتب الكود الذي يجعل ضغط الزر يؤدي إلى نسخ محتويات مربع النص وعرضها في اللافتة.

ج: كرر الخطوات من ١ إلى ٦ الموضحة في حل السؤال السابق، مع جعل اسم المشروع CopyText.. بعد هذا اتبع الخطوات التالية:

٧. من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.

٨. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة اللافتة Label لإضافتها إلى النموذج.

٩. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة مربع النص TextBox لإضافته إلى النموذج.

١٠. حدد الزر بضغطه مرة واحدة بزر الفأرة واضغط F4 لعرض نافذة الخصائص.. استخدم الخاصية Name لتغيير اسم الزر إلى BtnCopy، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى Copy.

١١. حدد اللافتة وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسم اللافتة إلى LblCopy، واستخدم الخاصية Text لحذف النص المعروض على اللافتة.. كل

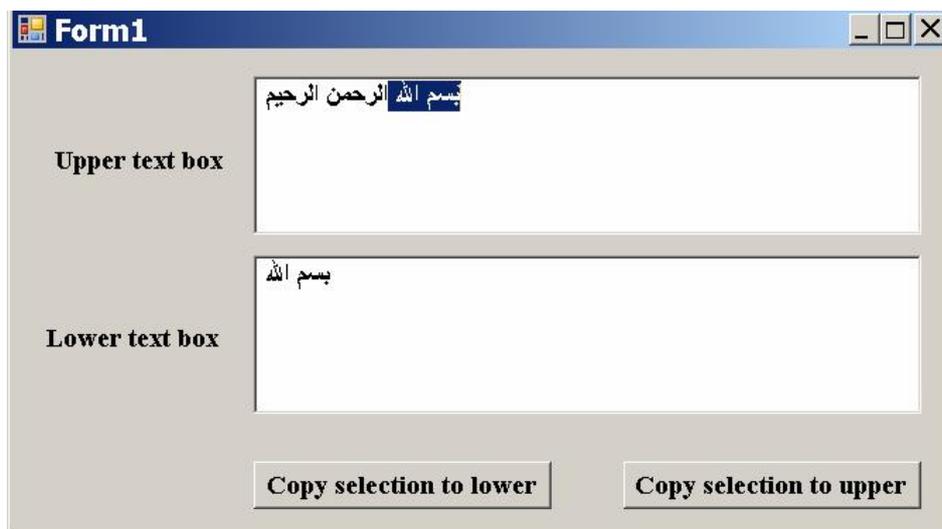
ما عليك هو حذف النص الموجود في خانة القيمة المجاورة لهذه الخاصية وتركها فارغة.

١٢. حدد مربع النص، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtOriginal.

١٣. انقر الزر مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnCopy_Click.. في هذا المعالج اكتب الكود التالي:

LblCopy.Text = TxtOriginal.Text

س٤: اكتب برنامج فيجوال بيزيك يحتوى على مربعي كتابة متعددي الأسطر (Multiline)، على أن يميز الأول بالاسم Upper، أما الآخر فيميز بالاسم Lower.. بالإضافة إلى ذلك يستخدم البرنامج زرین كما هو موضح بالصورة.



عند تشغيل البرنامج اكتب أي عدد من الجمل في مربع الكتابة العلوي، ثم قم باختيار أي جزء من النص المكتوب ثم اضغط على الزر "Copy" فيتم نسخ الجزء الذي تم اختياره في مربع الكتابة السفلي. وبالمثل يمكنك نسخ أي جزء من مربع الكتابة السفلي إلى مربع الكتابة العلوي، وذلك بالضغط على الزر الآخر، ويجب أن تأخذ في

الاعتبار أن الجزء المنسوخ يكتب حيث يوجد مؤشر الكتابة (Cursor) في مربع النص وليس في آخر النص.

ج: كرر الخطوات من ١ إلى ٦ الموضحة في حل السؤال الثاني، مع جعل اسم المشروع TwoTextBoxes.. بعد هذا اتبع الخطوات التالية:

٧. من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.. كرر هذا مرة أخرى لإضافة زر آخر.

٨. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة اللافتة Label لإضافتها إلى النموذج.. كرر هذا مرة أخرى لإضافة لافتة أخرى.

٩. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة مربع النص TextBox لإضافته إلى النموذج.. كرر هذا مرة أخرى لإضافة مربع نص آخر.

١٠. حدد الزر الأول بضغطة مرة واحدة بزر الفأرة واضغط F4 لعرض نافذة الخصائص.. استخدم الخاصية Name لتغيير اسم الزر إلى BtnLower، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى Copy selection to lower.

١١. حدد الزر الثاني وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى BtnUpper، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى Copy selection to upper.

١٢. حدد مربع النص الأول، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtUpper، وغير قيمة الخاصية MultiLine إلى True ليصير مربع النص متعدد الأسطر.

١٣. حدد مربع النص الثاني، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtLower، وغير قيمة الخاصية MultiLine إلى True ليصير مربع النص متعدد الأسطر.

١٤. حدد اللافتة الأولى وغير موضعها على النموذج لتحاذي مربع النص TxtUpper، وفي نافذة الخصائص استخدم الخاصية Text لتغيير النص المعروض على اللافتة إلى Upper text box.

١٥. حدد اللافتة الثانية وغير موضعها على النموذج لتحاذي مربع النص TxtLower، وفي نافذة الخصائص استخدم الخاصية Text لتغيير النص المعروض على اللافتة إلى Lower text box.

١٦. انقر الزر BtnLower مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnLower_Click.. في هذا المعالج اكتب الكود التالي:

TxtLower.SelectedText = TxtUpper.SelectedText

هذا الكود يستخدم الخاصية SelectedText ليقراً النص المحدد في مربع النص TxtUpper، ويضعه في الخاصية SelectedText الخاصة بمربع النص TxtLower.. في هذه الحالة لو كان هناك نص محدد في مربع النص TxtLower فسيتم حذفه وكتابة النص المنسوخ بدلاً منه.. أما لو لم يكن هناك أي نص محدد في مربع النص TxtLower فستتم كتابة النص المنسوخ في موضع مؤشر الفأرة Cursor.

١٧. انقر الزر BtnUpper مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnUpper_Click.. في هذا المعالج اكتب الكود التالي:

TxtUpper.SelectedText = TxtLower.SelectedText

ملحوظة:

عند ضغط الزر لن ترى النص المحدد في أي من مربعي النص، بسبب انتقال مؤشر الكتابة إلى الزر.. لتغيير هذا السلوك يمكنك استخدام الخاصية "إخفاء التحديد" HideSelection الخاصة بمربع النص، حيث ستجد قيمتها المبدئية True مما يعمل على إخفاء التحديد بعد مغادرة المؤشر لمربع النص.. كل ما عليك هو تغيير قيمة هذه الخاصية إلى True ليظل التحديد ظاهراً دائماً.

س ٥: ضع العلامة (√) أمام العبارة الصحيحة والعلامة (×) أمام العبارة الخاطئة، مع تصحيح الخطأ:

- ١- يمكن أن يحتوي اسم الأداة على مسافة أو فاصلة أو نقطة.
- ٢- يعتبر عدم وضع البادئة Btn في بداية اسم الزر خطأ في البرنامج.
- ٣- يمكن أن توجد أكثر من أداة بنفس الاسم لكن من أنواع مختلفة.
- ٤- يمكن تنسيق الأدوات على النموذج باستخدام القائمة Format.
- ٥- إذا جعلت للخاصية Visible القيمة False في أية أداة فستصير غير فعالة ولا يمكن استخدامها.
- ٦- يمكنك استخدام الخاصية BackGroundImageLayout لوضع صورة في خلفية الأداة.
- ٧- يمكن تغيير خط الأداة باستخدام الخاصية Font.
- ٨- ينطلق الحدث KeyPress عند ضغط زر الفأرة فوق الأداة.
- ٩- لكي يكون هناك تأثير للحدث، لا بد أن يكتب المبرمج معالج الحدث Event Handler.
- ١٠- يبدأ معالج الحدث بالكلمة Function وينتهي بالكلمة End Function.
- ١١- ينطلق الحدث Enter عند ضغط زر الإدخال Enter من لوحة المفاتيح.
- ١٢- ينطلق الحدث MouseMove عندما يتحرك مؤشر الفأرة فوق الأداة.
- ١٣- ينطلق الحدث Click عند ضغط المستخدم زر الفأرة فوق الأداة.
- ١٤- يمكن عرض اللافتة بإطار مجسم، باستخدام الخاصية BorderStyle ووضع القيمة FixedSingle فيها.
- ١٥- عند كتابة العلامة & في النص الموضوع في الخاصية Text الخاصة باللافتة، فإن الحرف التالي لها يصبح حرف اختصار للافتة.
- ١٦- عند ضغط Alt مع زر الاختصار من لوحة المفاتيح، ينتقل مؤشر الكتابة إلى اللافتة التي تستخدم حرف الاختصار المضغوط.

- ١٧- يمكن أن يعرض مربع النص أكثر من سطر، وذلك باستخدام الخاصية
.Lines
- ١٨- إذا جعلت للخاصية WordWrap القيمة True فإن النص المعروض في
اللائحة ينقسم على أكثر من سطر إذا تجاوز طول النص عرض اللائحة.
- ١٩- يظهر المنزلق الرأسى Vertical ScrollBar في مربع النص، فقط إذا كان
مربع النص متعدد الأسطر (MultiLine = True).
- ٢٠- لكي يظهر المنزلق الأفقى Horizontal ScrollBar في مربع النص، يكفي
أن تكون للخاصية ScrollBars القيمة Horizontal أو Both.
- ٢١- يمكنك معرفة عدد الحروف المكتوبة في مربع النص باستخدام الخاصية
.TextLength
- ٢٢- إذا وضعت صفراً في الخاصية MaxLength فلن يستطيع المستخدم الكتابة
في مربع النص.
- ٢٣- عند تغيير قيمة الخاصية CharacterCasing إلى Lower فإن كل الحروف
الموجودة في مربع النص تتحول إلى حروف صغيرة Small، لكن لو كتب
المستخدم أية حروف جديدة في مربع النص بعد هذا فإنها تظهر بحالتها
الطبيعية كما كتبها المستخدم.
- ٢٤- تستخدم الخاصية PasswordChar لإخفاء حروف كلمة المرور التي يكتبها
المستخدم في مربع النص.
- ٢٥- عند وضع القيمة ١ في الخاصية SelectionStart، والقيمة ١ في الخاصية
SelectionLength، فإن أول حرف في مربع النص يتم تحديده.
- ٢٦- تمحو الوسيلة Clear كل النص الموجود في مربع النص.
- ٢٧- تضيف الوسيلة AppendText النص المرسل إليها، في موضع مؤشر الكتابة
Cursor في مربع النص.
- ٢٨- تقوم الوسيلة Paste بنسخ النص من اللائحة إلى مربع النص.

- ٢٩- ينطلق الحدث TextChanged عندما يكتب المستخدم شيئاً من لوحة المفاتيح وقبل حدوث التغيير في مربع النص.
- ٣٠- لتحويل حروف النص المكتوب في مربع النص إلى حروف كبيرة Capital، يمكن استخدام الوسيلة ToUpper الخاصة بمربع النص.
- ٣١- يمكن وضع الأداة MsgBox على النموذج لعرض مربع الرسالة للمستخدم.
- ٣٢- يمكن أن يزيح المستخدم مربع الرسالة جانبا، ويواصل التعامل مع البرنامج بدون إغلاق مربع الرسالة.
- ٣٣- يقوم معامل التوجيه Prompt Argument في الدالة MsgBox باستقبال نص الرسالة التي سيتم عرضها للمستخدم.
- ٣٤- في الدالة MsgBox، يتحكم المعامل Title Argument في عدد الأزرار التي تظهر على الرسالة.
- ٣٥- للدالة MsgBox معامل واحد إجباري ومعاملان اختياريان.

ج:

- ١- (×) الصواب: يمكن أن يحتوي اسم الأداة على شرطة منخفضة _ لكن لا يمكن أن يحتوي على أية علامات ترقيم أو علامات حسابية أو مسافات.
- ٢- (×) الصواب: وضع البادئة Btn في بداية اسم الزر ليس إلزاميا ولا يسبب خطأ في البرنامج، لكنه يسهل فهم الكود بعد ذلك، لهذا هو أمر مرغوب فيه بشدة.
- ٣- (×) الصواب: لا يمكن وجود أكثر من أداة بنفس الاسم حتى لو كانت من أنواع مختلفة، فلا يمكن مثلا أن يوجد في البرنامج مربع نص اسمه Calculate وزر اسمه Calculate أيضا، لأن هذا سيمنع فيجيوال بيزيك من معرفة الأداة التي يشير إليها الاسم Calculate في الكود.. هنا تظهر فائدة وضع البادئة التي تختصر نوع الأداة في بداية اسمها، مثل BtnCalculate و TxtCalculate.

٤- (√)

٥- (x) الصواب: إذا جعلت للخاصية Visible القيمة False في أية أداة فستختفي الأداة من على النموذج.. أما إذا جعلت للخاصية Enabled القيمة False فستصير الأداة غير فعالة ولا يمكن استخدامها.

٦- (x) الصواب: يمكنك استخدام الخاصية BackGroundImage لوضع صورة في خلفية الأداة، ويمكنك استخدام الخاصية BackGroundImageLayout لتنسيق شكل الصورة على خلفية الأداة.

٧- (√)

٨- (x) الصواب: ينطلق الحدث KeyPress عند ضغط زر من لوحة المفاتيح فوق الأداة، بينما ينطلق الحدث MouseDown عند ضغط زر الفأرة فوق الأداة.

٩- (√)

١٠- (x) الصواب: يبدأ معالج الحدث بالكلمة Sub وينتهي بالكلمة End Sub.

١١- (x) الصواب: ينطلق الحدث Enter عند انتقال مؤشر الكتابة Focus إلى الأداة لتصير هي الأداة الفعالة، بينما ينطلق الحدث KeyPress عند ضغط زر الإدخال Enter من لوحة المفاتيح.

١٢- (√)

١٣- (√)

١٤- (x) الصواب: يمكن عرض اللافتة بإطار مجسم، باستخدام الخاصية BorderStyle ووضع القيمة Fixed3D فيها.

١٥- (√)

١٦- (x) الصواب: لا تستطيع اللافتة استقبال مؤشر الكتابة Focus، لهذا عند ضغط Alt مع زر الاختصار من لوحة المفاتيح يتم انتقال مؤشر الكتابة إلى الأداة التالية للافتة على النموذج.

١٧- (x) الصواب: يمكن أن يعرض مربع النص أكثر من سطر، وذلك بوضع القيمة True في الخاصية MultiLine.

- ١٨- (×) الصواب: إذا جعلت للخاصية WordWrap القيمة True فإن النص المعروف في مربع النص ينقسم على أكثر من سطر إذا تجاوز طول النص عرض مربع النص.. ويمكن الحصول على نفس الأداء في اللافتة بوضع القيمة False في الخاصية AutoSize الخاصة باللافتة.
- ١٩- (×) الصواب: لكي يظهر المنزلق الرأسي Vertical ScrollBar في مربع النص، يجب أن تكون للخاصية ScrollBars القيمة Vertical أو Both، كما يجب أيضا أن يكون مربع النص متعدد الأسطر (MultiLine = True).
- ٢٠- (×) الصواب: لكي يظهر المنزلق الأفقي Horizontal ScrollBar في مربع النص، يجب أن تكون للخاصية ScrollBars القيمة Horizontal أو Both، ويجب أن يكون مربع النص متعدد الأسطر (MultiLine = True)، ويجب كذلك أن تكون لخاصية التفاف السطر WordWrap القيمة False.
- ٢١- (√)
- ٢٢- (×) الصواب: إذا وضعت صفرا في الخاصية MaxLength فلن يكون لها أي تأثير، وسيتمكن المستخدم من كتابة أي عدد من الحروف في مربع النص بدون قيود.
- ٢٣- (×) الصواب: عند تغيير قيمة الخاصية CharacterCasing إلى Lower فإن كل الحروف الموجودة في مربع النص تتحول إلى حروف صغيرة Small، وكذلك سيتم تحويل كل الحروف الجديدة التي يكتبها المستخدم بعد هذا في مربع النص إلى حروف صغيرة.
- ٢٤- (√)
- ٢٥- (×) الصواب: عند وضع القيمة ١ في الخاصية SelectionStart، والقيمة ١ في الخاصية SelectionLength، فإن ثاني حرف في مربع النص يتم تحديده.. ولتحديد أول حرف في مربع النص، يجب وضع القيمة ٠ في الخاصية SelectionStart، والقيمة ١ في الخاصية SelectionLength..

السبب في هذا أن أول حرف في مربع النص يأخذ الترقيم صفر في ترتيب الحروف!

٢٦- (✓)

٢٧- (×) الصواب: تضيف الوسيلة AppendText النص المرسل إليها، بعد آخر حرف مكتوب في مربع النص.

٢٨- (×) الصواب: تقوم الوسيلة Paste بنسخ النص من لوحة قصاصات الويندوز Clipboard ولصقه في مربع النص.

٢٩- (×) الصواب: ينطلق الحدث KeyPress عندما يكتب المستخدم شيئاً من لوحة المفاتيح وقبل حدوث التغيير في مربع النص، بينما ينطلق الحدث TextChanged بعد كتابة الحروف وحدث التغيير في مربع النص.

٣٠- (×) الصواب: لتحويل حروف النص المكتوب في مربع النص إلى حروف كبيرة Capital، يمكن استخدام الخاصية CharacterCasing الخاصة بمربع النص.. لكن الوسيلة ToUpper خاصة بفئة النصوص String Class، ولاستخدامها مع مربع النص يجب تطبيقها على الخاصية Text مثلًا كالتالي:

TextBox1.Text = TextBox1.Text.ToUpper()

٣١- (×) الصواب: MsgBox دالة وليست أداة ولا يتم وضعها على النموذج، بل يتم استدعاء هذه الدالة من الكود مباشرة لعرض مربع الرسالة للمستخدم.

٣٢- (×) الصواب: لا يمكن أن يواصل المستخدم التعامل مع البرنامج إلا بعد إغلاق مربع الرسالة.

٣٣- (✓)

٣٤- (×) الصواب: في الدالة MsgBox، يتحكم المعامل الثاني Buttons في عدد الأزرار التي تظهر على الرسالة، بينما يتحكم المعامل الثالث Title في عنوان مربع الرسالة.

٣٥- (✓)

س٦: اذكر سبب الخطأ في الأكواد التالية، وقم بتصحيحها:

- 1- TextBox1.Text = Hello Ahmad
- 2- Label1.WordWrap = True
- 3- TextBox1.AppendText = "New text"
- 4- TextBox1 = TextBox2.Text
- 5- Button1.Enabled = True
- 6- TextBox1.Visible = "Yes"
- 7- TextBox1.Color = Red
- 8- Dim S As String, TextBox1.Text = S
- 9- TextBox1.Text
= "My name is Mohammad"
- 10- TextBox1.AutoSize = True
- 11- Label1.BorderStyle = None
- 12- TextBox1.TextAlign = ContentAlignment.BottomCenter
- 13- Label1.MultiLine = True
- 14- TextBox1.ScrollBars(ScrollBars.Both)
- 15- If TextBox1.Text = ""
 TextBox1.Text = "Hello"
- 16- TextBox1.TextLength = 3
- 17- TextBox1.MaxLength = -1
- 18- TextBox1.AppendText (""
- 19- TextBox1.Undo(3)
- 20- TextBox1.TextChanged = True
- 21- MsgBox "Wrong"
- 22- MsgBox(, MsgBoxStyle.YesNo, "Error")
- 23- MsgBox("This is wrong", "Error")
- 24- MsgBox("wrong", MsgBoxStyle.YesNo, MsgBoxStyle.Critical,
 "")

ج: ١- يجب وضع النص بين علامتي تنصيص:

TextBox1.Text = "Hello Ahmad"

٢- ليس للافتة خاصية اسمها WordWrap، فهي تابعة لمربع النص:

TextBox1.WordWrap = True

٣- AppendText وسيلة وليست خاصية، ويجب إرسال النص إليها بوضعه بين قوسين بعد اسم الوسيلة:

```
TextBox1.AppendText ("New text")
```

٤- لا يمكن وضع النص في كائن مربع النص TextBox1 نفسه مباشرة، لكن يمكن استخدام الخاصية Text الخاصة بمربع النص TextBox1 بدلا من ذلك:

```
TextBox1.Text = TextBox2.Text
```

٥- يتم استخدام الخاصية بوضع نقطة بينها وبين اسم الأداة، وليس قوسين:

```
Button1.Enabled = True
```

٦- الخاصية Visible خاصة منطقية تقبل قيمتين اثنتين فحسب: إما True وإما False، ولا تقبل نصوصا:

```
TextBox1.Visible = True
```

٧- لا يحتوي مربع النص على خاصية اسمها Color.. يمكن استخدام الخاصية ForeColor بدلا من هذا.. كما أن الألوان يجب أن تكون منسوبة إلى كائن الألوان Color:

```
TextBox1.ForeColor = Color.Red
```

٨- لا يمكنك كتابة أكثر من أمر في نفس السطر وفصلها بفاصلة.. إما أن تكتب كل أمر في سطر مستقل:

```
Dim S As String
```

```
TextBox1.Text = S
```

وإما أن تفصل بين الأمرين بنقطتين قائمتين:

```
Dim S As String : TextBox1.Text = S
```

٩- لا يمكن فصل أمر واحد على أكثر من سطرين.. إما أن تكتب الأمر كله في سطر واحد:

```
TextBox1.Text = "My name is Mohammad"
```

وإما أن تضع علامة تكملة السطر _ في نهاية السطر الأول:

```
TextBox1.Text _
```

```
= "My name is Mohammad"
```

١٠- لا يمتلك مربع النص خاصية اسمها AutoSize، فهي تابعة للافتة:

```
Label1.AutoSize = True
```

١١ - يجب نسبة القيمة None إلى الكائن BorderStyle الذي يحتوي على كل القيم المناسبة للخاصية:

```
Label1.BorderStyle = BorderStyle.None
```

١٢ - الخاصية TextAlign في مربع النص تأخذ قيمة من قيم الكائن HorizontalAlignment، بينما تأخذ الخاصية TextAlign في اللافتة قيمة من قيم الكائن ContentAlignment:

```
TextBox1.TextAlign = HorizontalAlignment.Center
```

١٣ - لا تمتلك اللافتة الخاصية MultiLine، فهي خاصة بمربع النص:

```
TextBox1.MultiLine = True
```

١٤ - ScrollBars هي خاصية وليست وسيلة، ويجب وضع القيمة فيها باستخدام العلامة = :

```
TextBox1.ScrollBars = ScrollBars.Both
```

١٥ - جملة If ينقصها Then في نهاية السطر الأول، و End If بعد السطر الثاني:

```
If TextBox1.Text = "" Then  
    TextBox1.Text = "Hello"  
End If
```

١٦ - الخاصية TextLength للقراءة فقط ReadOnly، لهذا يمكن استخدامها لمعرفة عدد الحروف المكتوبة في مربع النص، لكن لا يمكن استخدامها لتغيير عدد هذه الحروف:

```
Dim I As Integer = TextBox1.TextLength
```

١٧ - لا يمكن وضع قيمة سالبة في الخاصية MaxLength، لكن يمكن وضع أية قيمة موجبة مثل ١٠:

```
TextBox1.MaxLength = 10
```

١٨ - هناك قوس ناقص في نهاية السطر:

```
TextBox1.AppendText ("")
```

١٩ - الوسيلة Undo لا تستقبل أية معاملات:

```
TextBox1.Undo( )
```

٢٠ - TextChanged هو حدث ولا يمكن استخدامه بهذه الطريقة، بل يجب إنشاء معالج له وكتابة الكود فيه.. أما الخاصية التي يمكن التعامل معها للإشارة إلى أن النص قد تغير، فهي الخاصية Modified:

TextBox1.Modified = True

٢١ - يجب وضع معامل الدالة بين قوسين:

MsgBox ("Wrong")

٢٢ - المعامل الأول للدالة MsgBox إجباري ولا يمكن تركه فارغا.. إذا كنت لا تريد إرساله فأرسل نصا فارغا إلى الدالة:

MsgBox("", MsgBoxStyle.YesNo, "Error")

٢٣ - المعامل الثاني لا يمكن أن يكون نصا.. يجب أن تضع فاصلة إضافية لتخبر فيجيوال ببيزيك أن المعامل الاختياري الثاني مفقود، وإلا فإنها ستظن أنك ترسل النص "Error" إلى المعامل الثاني:

MsgBox("This is wrong", , "Error")

٢٤ - لعرض الأزرار والأيقونات معا على مربع الرسالة استخدم علامة الجمع + بينهما.. أما استخدام الفاصلة فسيجعل فيجيوال ببيزيك تظن أنك ترسل معاملا زائدا إلى الدالة:

MsgBox("wrong", MsgBoxStyle.YesNo + MsgBoxStyle.Critical, "")

س٧: اختر الإجابة الصحيحة:

١- تستخدم لتغيير اسم الأداة.

أ- Text ب- TextChanged ج- Name

٢- تستخدم لتفعيل الأداة.

أ- Enabled ب- Visible ج- Show

٣- تستخدم لتغيير خط الكتابة في الأداة.

أ- ForeColor ب- Font ج- Text

٤- تستخدم لوضع صورة على الأداة.

- أ- BackGroundImage - ب- BackGroundImageLayout - ج- ImageAlign
 ٥- تستخدم لتنسيق تخطيط الصورة على الأداة.
- أ- BackGroundImage - ب- BackGroundImageLayout - ج- ImageAlign
 ٦- تستخدم لتغيير محاذاة الصورة على الزر واللافتة.
- أ- Image - ب- BackGroundImageLayout - ج- ImageAlign
 ٧- يمكن مط الصورة لتملأ كل مساحة الأداة باستخدام الخاصية.....
- أ- BackGroundImage - ب- BackGroundImageLayout - ج- ImageAlign
 ٨- تستخدم لتغيير شكل مؤشر الفأرة فوق الأداة.
- أ- MouseMove - ب- Focus - ج- Cursor
 ٩- ينطلق الحدث عند ضغط زر الفأرة فوق الأداة.
- أ- MouseMove - ب- Click - ج- KeyPress
 ١٠- ينطلق الحدث عند مغادرة مؤشر الكتابة Focus للأداة.
- أ- MouseMove - ب- Focus - ج- Leave
 ١١- تستخدم لجعل مساحة الأداة تلقائيا مناسبة للنص المعروض عليها:
- أ- WordWrap - ب- AutoSize - ج- Size
 ١٢- تستخدم لتغيير شكل إطار اللافتة.
- أ- BorderStyle - ب- AutoSize - ج- Size
 ١٣- تستخدم لتغيير محاذاة النص.
- أ- Text - ب- ImageAlign - ج- TextAlign
 ١٤- تستخدم لتقسيم السطر الطويل في مربع النص على سطرين.
- أ- MultiLine - ب- WordWrap - ج- AutoSize
 ١٥- تستخدم للتحكم في نوع المنزلق (شريط اللف) المعروض في مربع النص.
- أ- Height - ب- Vertical - ج- ScrollBars

- ١٦- تستخدم لمعرفة طول النص الموجود في مربع النص.
- أ- `TextLength` ب- `SelectionLength` ج- `MaxLength`
- ١٧- تستخدم لتغيير طريقة كتابة الحروف في مربع النص.
- أ- `TextChanged` ب- `ToUpper` ج- `CharacterCasing`
- ١٨- ينطلق الحدث بعد كتابة حرف في مربع النص.
- أ- `Modified` ب- `TextChanged` ج- `KeyPress`
- ١٩- تستخدم للكتابة في موضع مؤشر الفأرة في مربع النص.
- أ- `Text` ب- `AppendText` ج- `SelectedText`
- ٢٠- لا يمكن استخدام لحذف النص المحدد فقط في مربع النص.
- أ- `Clear` ب- `Cut` ج- `SelectedText`

ج:

١- ج	٢- أ	٣- ب	٤- أ
٥- ب	٦- ج	٧- ب	٨- ج
٩- ب	١٠- ج	١١- ب	١٢- أ
١٣- ج	١٤- ب	١٥- ج	١٦- أ
١٧- ج	١٨- ب	١٩- ج	٢٠- أ

ملحوظة حول الإجابة رقم ٢٠:

يمكن استخدام الوسيلة `Cut` لحذف النص المحدد في مربع النص، لأنها تقطع هذا النص وتضعه في لوحة القصاصات `Clipboard`.

ويمكن استخدام الخاصية `SelectedText` لمحو النص المحدد في مربع النص، لأنها تكتب فوق النص المحدد حالياً، ولو وضعنا فيها نصاً فارغاً فستمحو النص المحدد:

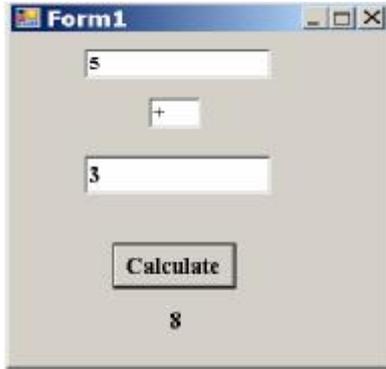
```
TextBox1.SelectedText = ""
```

لكن لا يمكن استخدام الوسيلة `Clear` لمحو النص المحدد فقط، لأنها تمحو كل النص الموجود في مربع النص، سواء كان محدداً أم لا.

س٨: اكتب برنامج فيجيوال بيزيك لتنفيذ خريطة التدفق والكود الزائف "سودودكود" الموجودين في السؤال الأول بالفصل الثالث، مع الأخذ في الاعتبار أن قراءة المدخلات ستتم من خلال ثلاثة مربعات نص، بينما إخراج الناتج سيتم باستخدام لافتة Label، وسيتم حساب الناتج عند الضغط على زر.

ج: كرر الخطوات من ١ إلى ٦ الموضحة في حل السؤال الثاني، مع جعل اسم المشروع ArithmeticOperations.. بعد هذا اتبع الخطوات التالية:
٧. من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.

٨. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة اللافتة Label لإضافتها إلى النموذج.



٩. في صندوق الأدوات انقر مرتين بالفأرة على أيقونة مربع النص TextBox لإضافته إلى النموذج.. كرر هذا مرتين آخرين لإضافة مربعي نص آخرين.. نسق الأدوات على النموذج لتبدو كما في الصورة.

١٠. حدد الزر بضغطه مرة واحدة بزر الفأرة واضغط

F4 لعرض نافذة الخصائص.. استخدم الخاصية Name لتغيير اسم الزر إلى BtnCalculate، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى Calculate.

١١. حدد مربع النص الأول، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtN1، وضع في الخاصية MaxLength القيمة ٦ لمنع المستخدم من كتابة أي عدد أطول من ٦ أرقام.

١٢. حدد مربع النص الثاني، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtOperator، وضع في الخاصية MaxLength القيمة ١ لنجبر المستخدم على كتابة حرف واحد فقط، لأن مربع النص هذا يخص العلامة الحسابية.

١٣. حدد مربع النص الثالث، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمه إلى TxtN2، وضع في الخاصية MaxLength القيمة ٦ لنمنع المستخدم من كتابة أي عدد أطول من ٦ أرقام.

١٤. حدد اللافتة، وفي نافذة الخصائص استخدم الخاصية Name لتغيير اسمها إلى LblResult، واستخدم الخاصية Text لحذف النص المعروف على اللافتة واتركها فارغة.

١٥. انقر الزر BtnCalculate مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnCalculate_Click.. سيكون هذا المعالج كالتالي:

```
Private Sub BtnCalculate_Click(sender As Object, _
    e As EventArgs) Handles BtnCalculate.Click
    Dim N1 As Integer = TxtN1.Text
    Dim N2 As Integer = TxtN2.Text
    Dim Op As String = TxtOperator.Text
    Dim Result As Integer = 0
    If Op = "+" Then
        Result = N1 + N2
        GoTo 9
    End If

    If Op = "-" Then
        Result = N1 - N2
        GoTo 9
    End If

    If Op = "*" Then
        Result = N1 * N2
        GoTo 9
    End If
```

```

If Op = "/" Then
  If N2 = 0 Then
    LblResult.Text = "Can't divide by zero."
    GoTo 10
  End If
  Result = N1 / N2
  GoTo 9
End If
LblResult.Text = "Wrong operation."
GoTo 10

```

9:

```

LblResult.Text = Result

```

10:

```

End Sub

```

لاحظ أن هذا الكود قريب بدرجة كبيرة من الكود الزائف "سودوكود" الذي كتبناه في إجابة السؤال الأول في الفصل الثالث، حيث استخدمنا جملة `Goto` الخاصة بفيجيوال بيزيك للقفز إلى العلامتين ٩ و ١٠.. لاحظ أن علامة السطر هي رقم أو اسم يأتي في أول السطر ويوضع بعده نقطتان.. في الحقيقة هذه الطريقة في كتابة الكود لم تعد مستخدمة اليوم كثيرا في فيجيوال بيزيك، فهي أقرب إلى لغة البيزيك القديمة التي كانت تعمل على الدوس، ونادرا ما يستخدم المبرمج اليوم `Goto` وعلامات السطور!.. وبدلا منها، يمكن استخدام الحل الآخر الذي اتبعناه في حل السؤال الأول في الفصل الثاني، وذلك باستخدام الجملة `Else`، وسنستفيد هنا من وجود جملة `ElseIf` في فيجيوال بيزيك، فهي تختصر الشروط المتداخلة.. افترض أن لديك الشروط التالية:

```

If Cond1 Then
  Block1
Else
  If Cond2 Then
    Block2
  Else
    If Cond3 Then
      Block3
    Else
      Block4
    End If
  End If
End If

```

يمكننا إعادة كتابة هذه الشروط باستخدام ElseIf كالتالي:

```
If Cond1 Then
  Block1
ElseIf Cond2 Then
  Block2
ElseIf Cond3 Then
  Block3
Else
  Block4
End If
```

كما هو واضح، هذا أوضح وأسهل وأكثر اختصاراً بكثير من الصيغة الأولى. لاحظ كذلك أن هناك خطأ سيحدث في البرنامج لو ضغط المستخدم الزر بدون كتابة أي شيء في مربع النص الأول أو الثاني، لأن فيجيوال بيزيك لن تستطيع تحويل النص الفارغ إلى عدد صحيح لوضعه في المتغير N1 أو N2.. لحل هذه المشكلة يمكننا تطوير هذا الكود قليلاً لفحص محتوى مربع النص أولاً، فإن كان فارغاً وضعنا صفراً في العدد.. هذا هو الكود الجديد الذي يحل هذه المشكلة، ويستخدم جملة ElseIf بدلاً من :Goto

```
Dim N1 As Integer
If TxtN1.Text = "" Then
  N1 = 0
Else
  N1 = TxtN1.Text
End If
```

```
Dim N2 As Integer
If TxtN2.Text = "" Then
  N2 = 0
Else
  N2 = TxtN2.Text
End If
```

```
Dim Op As String = TxtOperator.Text
```

```

If Op = "+" Then
  LblResult.Text = N1 + N2
ElseIf Op = "-" Then
  LblResult.Text = N1 - N2
ElseIf Op = "*" Then
  LblResult.Text = N1 * N2
ElseIf Op = "/" Then
  If N2 = 0 Then
    LblResult.Text = "Can't divide by zero."
  Else
    LblResult.Text = N1 / N2
  End If
Else
  LblResult.Text = "Wrong operation."
End If

```

وهناك طريقة أخرى لمنع حدوث خطأ في البرنامج إن لم يكتب المستخدم أحد العددين.. في هذه الطريقة يمكننا تعطيل الزر Calculate بوضع القيمة False في الخاصية Enabled الخاصة به، وذلك إذا كان أي من TxtN1 أو TxtN2 فارغا.. لفعل هذا اتبع الآتي:

- حدد الزر على النموذج، واضغط F4 لفتح نافذة الخصائص، واجعل للخاصية Enabled القيمة False.. هذا سيجعل الزر معطلا عند تشغيل البرنامج، لأن مربعات النصوص تكون فارغة في هذه اللحظة.
- انقر مرتين بالفأرة على مربع النص TxtN1.. سيؤدي هذا إلى فتح نافذة محرر الكود وعرض معالج الحدث تغيير النص TxtN1_TxtChanged.. اكتب في هذا المعالج ما يلي:

```

If TxtN1.Text = "" Then
  BtnCalculate.Enabled = False
ElseIf TxtN2.Text = "" Then
  BtnCalculate.Enabled = False
Else
  BtnCalculate.Enabled = True
End If

```

هذا الكود يقول ببساطة:

إذا كان مربع النص الأول فارغاً:

فقم بتعطيل الزر

غير هذا إذا كان مربع النص الثاني فارغاً:

فقم بتعطيل الزر

غير هذا:

قم بتفعيل الزر (لأن خطأ الشرطين السابقين يعني أن كلا المربعين غير

فارغين)

- انقر مرتين بالفأرة على مربع النص TxtN2.. سيؤدي هذا إلى فتح نافذة محرر

الكود وعرض معالج الحدث تغيير النص TxtN2_TxtChanged.. اكتب في هذا

المعالج نفس الكود الذي كتبناه في معالج الحدث TxtN1_TxtChanged.

بهذه الطريقة سنضمن أن أي تغيير سيقوم به المستخدم في مربع النص الأول أو الثاني

سيؤدي إلى تعطيل الزر أو تفعيله بالشكل الصحيح، وبهذا يبدو البرنامج اذكى وأكثر

دقة، وأقل عرضة لحدوث الأخطاء.

ملحوظة:

يمكنك كتابة معالج واحد لحدث تغيير النص في كل من TxtN1 و TxtN2، وذلك بدلا من تكرار نفس الكود في معالجات مختلفين.. سأخبرك كيف تفعل هذا على سبيل المعرفة، لكن هذا لي مطلوبا منك في الامتحان.

يحدث هذا باستخدام الجملة Handles الموجودة في نهاية معالج الحدث:

Handles TxtN1.TextChanged

هذه الجملة تقول إن المعالج يستجيب للحدث TextChanged الخاص بالكائن TxtN1..

كل ما سنفعله نحن هو وضع فاصلة في نهاية هذه الجملة، وكتابة اسم حدث آخر

ليستجيب له المعالج:

Handles TxtN1.TextChanged, TxtN2.TextChanged

وهذا هو كل شيء!

هكذا يجب أن يبدو معالج حدث تغيير النص في مربعي النص TxtN1 و TxtN3:

```
Private Sub TxtN1_TextChanged(ByVal sender As Object, _  
    ByVal e As EventArgs) _  
    Handles TxtN1.TextChanged, TxtN2.TextChanged  
  
    If TxtN1.Text = "" Then  
        BtnCalculate.Enabled = False  
    ElseIf TxtN2.Text = "" Then  
        BtnCalculate.Enabled = False  
    Else  
        BtnCalculate.Enabled = True  
    End If  
End Sub
```

نموذج امتحان رقم ١
للفصل الثالث الإعدادي، الفصل الدراسي الأول

س١: ضع العلامة (U) أمام الجملة الصحيحة، والعلامة (x) أمام الجملة الخاطئة، مع تصحيح الخطأ:

- ١- تستخدم نافذة الخصائص لإضافة خصائص جديدة إلى الأداة Control.
- ٢- الأداة TextBox تعتبر فئة Class، والنسخة TextBox1 الموضوع من هذه الأداة على النموذج تعتبر Object.
- ٣- جهاز الحاسب الشخصي الخاص بك هو نظام معلومات.
- ٤- تستطيع فيجيوال بيزيك تنفيذ الكود الزائف "سودوكود" مباشرة.
- ٥- يخفي التغليف خصائص ووسائل وأحداث الكائن عن المستخدم.

س٢: أجب عن الأسئلة التالية"

- أ- ما المقصود بـ IDE؟
- ب- اكتب خطوات فتح نافذة IDE على جهازك.
- ج- اكتب خطوات إنشاء وحفظ برنامج فيجيوال بيزيك جديد اسمه Prog1.
- د- كيف يمكن وضع زر على النموذج اسمه BtnMsg، يظهر عليه النص "Hello Egypt"؟
- هـ- كيف يمكنك جعل هذا الزر يعرض مربع رسالة تقول للمستخدم: " Hello Egypt"؟

س٣: إذا اشترى العميل منتجات إحدى الشركات بقيمة ١٠٠٠ جنيه أو أكثر، فإنها تقدم له خصما قيمته ١٠%، أما إذا كانت قيمة المشتريات أقل من ذلك لا يحصل على أي خصم:

- ١- ارسم خريطة التدفق لقراءة قيمة المشتريات Sales من العميل ، وعرض السعر المطلوب Price.

٢- اكتب الكود الزائف "سودوكو" المناظر لخريطة التدفق.

٣- لو لديك نموذج في برنامج فيجيوال بيزيك عليه مربع نص اسمه TxtSales يكتب فيه المستخدم قيمة المشتريات، ولافتة Label اسمها LblPrice تعرض فيها للمستخدم السعر النهائي.. اكتب كود فيجوال بيزيك الذي يقرأ قيمة المشتريات ويعرض السعر النهائي على اللافتة.

إجابة نموذج امتحان رقم ١

ج ١:

١- (×) الصواب: تستخدم نافذة الخصائص لتغيير قيم الخصائص الموجودة في الأداة، لكنها لا تستطيع حذف أو إضافة أي خصائص.

٢- (ü)

٣- (×) الصواب: يحتاج نظام المعلومات إلى برنامج لحفظ البيانات ومعالجتها، وموظفين يدخلون البيانات ويشغلون البرنامج، وفي الغالب يعتمد نظام المعلومات على شبكة متصلة ببعضها من أجهزة الحاسب Network وليس حاسبا واحدا.. ونظرا لأن جهازك الشخصي لا يتوفر فيه أي من هذا، فلا يمكن اعتباره نظام معلومات.

٤- (×) يجب أن يقوم المبرمج بتحويل الكود الزائف "سودوكود" إلى كود فيجيوال بيزيك أولا حتى يمكن تشغيله.

٥- (×) يخفي التغليف عن المستخدم بيانات الفئة Class، ويخفي كود الخصائص والوسائل والأحداث التي يستخدمها الكائن، لكن التغليف لا يخفي الخصائص والوسائل والأحداث نفسها إلا إذا تم تعريفها على أنها خاصة Private.

ج ٢: أ- راجع إجابة السؤال الأول في الفصل الخامس.

ب- خطوات فتح نافذة IDE على جهازك:

(١) من سطح المكتب اضغط قائمة البداية Start.

(٢) من القائمة Start اضغط Programs.

(٣) من القائمة Programs اضغط Microsoft Visual Studio 2005.

(٤) من القائمة Microsoft Visual Studio 2005 اضغط الأمر

Microsoft Visual Studio 2005.

ج- راجع الخطوات من ١ إلى ٦ في إجابة السؤال الثاني في الفصل السادس، مع

تسمية البرنامج Prog1.

د- من القائمة View اضغط الأمر Toolbox لعرض صندوق الأدوات، وافتح الشريط Common Controls وانقر مرتين بالفأرة على أيقونة الزر Button لإضافته إلى النموذج.. حدد الزر على النموذج بضغطه مرة واحدة بزر الفأرة واضغط F4 لعرض نافذة الخصائص.. استخدم الخاصية Name لتغيير اسم الزر إلى BtnMsg، واستخدم الخاصية Text لتغيير العنوان المعروض على الزر إلى "Hello Egypt".

هـ- انقر الزر مرتين بالفأرة.. سيعرض هذا صفحة محرر الكود Code Editor، وفيها معالج حدث ضغط الزر Sub BtnMsg_Click.. في هذا المعالج اكتب الكود التالي:

MsgBox("Hello Egypt")

ج ٣: لاحظ أن خصم ١٠% من قيمة المشتريات، يعني أن السعر النهائي سيكون ٩٠% من قيمة المشتريات:

٢ - السودوكود:

```

11- Read the Sales
12- IF the Sales is less than 1000
    THEN
        Price = Sales
    Else
        Price = 0.9 * Sales
    END IF
13- Print an output line
    showing the Price
14- Stop processing

```

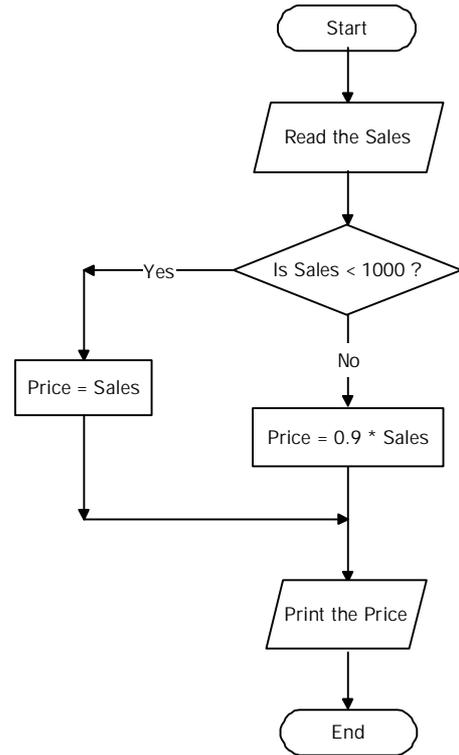
٣ - كود فيجوال بيزيك:

```

Dim Sales As Integer
Sales = TxtSales.Text
If Sales < 1000 Then
    LblPrice.Text = Sales
Else
    LblPrice.Text = 0.9 * Sales
End If

```

١ - خريطة التدفق:



نموذج امتحان رقم ٢

للفصل الثالث الإعدادي، الفصل الدراسي الأول

س ١: ضع العلامة (U) أمام الجملة الصحيحة، والعلامة (x) أمام الجملة الخاطئة، مع تصحيح الخطأ:

- ١- تستخدم خريطة التدفق لعرض خطوات لغة الآلة بشكل مبسط ليفهمها المبرمجون.
- ٢- يحفظ الحاسب قيم المتغيرات على القرص الصلب Hard Desk.
- ٣- عندما نرسم خريطة التدفق على الأوراق فإنها تعتبر فئة Class، بينما يعتبر البرنامج الذي كتبناه بناء على هذه الخريطة كائننا Object.
- ٤- استخدام نافذة الخصائص لتغيير قيم خصائص الزر الموضوع على النموذج، يؤدي إلى حفظ هذه التغييرات في فئة الزر Button Class.
- ٥- لا يستطيع المستخدم كتابة أي نص في المبين Label.

س ٢: اختر الإجابة الأصح لكل نقطة مما يلي:

- ١- معدل الزيادة السكانية في نظام معلومات هيئة التعداد والإحصاء هو:
أ- خاصة ب- بيان ج- وسيلة د- معلومة هـ - أ، د معا
- ٢- اسم صديقك (علي) يمكن أن يكون:
أ- خاصة ب- بيان ج- وسيلة د- معلومة هـ - أ، ب معا
- ٣- يعتبر قالب الطباعة الذي يتم نسخ الصفحة منه:
أ- تصنيف ب- كائن ج- وسيلة د- معلومة هـ - ب، د معا
- ٤- تستخدم لعرض معلومة لمشغل البرنامج:
أ- IDE ب- Show ج- MsgBox د- Text هـ - ج، د معا
- ٥- يسمى النص " " : TextBox1.Text
أ- Algorithm ب- Flowchart ج- Pseudo code د- Code هـ - ج، د معا

س٣: الشكل التالي يمثل مخطط تدفق:

١- إذا أدخل المشغل القيم التالية:

Base = 0

Increment = 2

NumberOfTerms = 5

فاكتب النتائج التي ستتم

طباعتها على الشاشة.

٢- ما الوظيفة التي يقوم بها هذا

المخطط؟

٣- ماذا سيحدث إذا وضع المشغل

في المتغير

NumberOfTerms صفرا أو

عددا سالبا؟.. وهل ستحدث

نفس المشكلة إذا وضع المشغل

في المتغير Base أو المتغير

Increment عددا سالبا؟..

ولماذا؟

٤- كيف يمكن تجنب مشكلة وضع

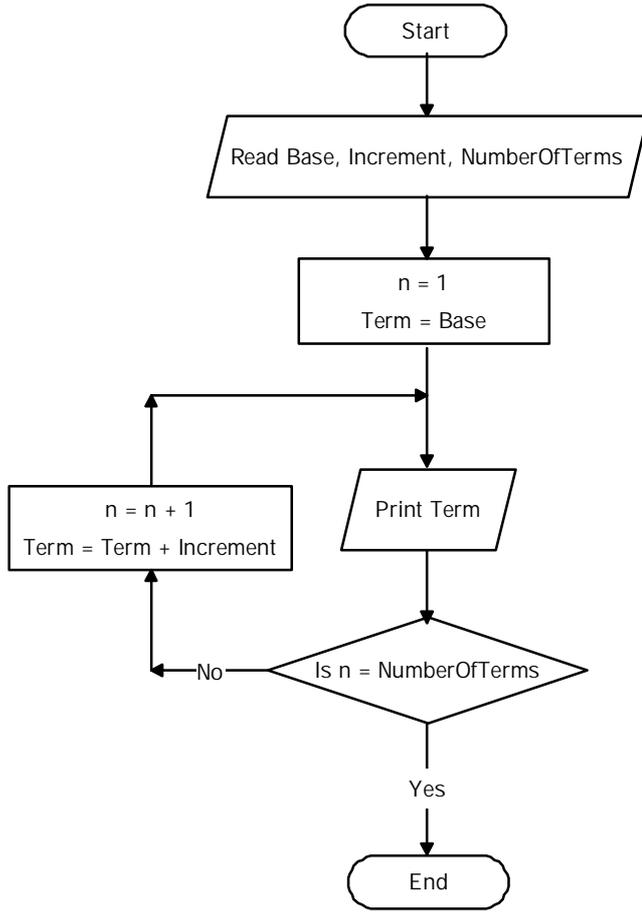
صفر أو عدد سالب في المتغير

NumberOfTerms.. ارسم

خريطة التدفق الجديدة التي تحل

هذه المشكلة.

٥- اكتب السودوكود لخريطة التدفق المعدلة.



إجابة نموذج امتحان رقم ٢

ج ١:

١- (×) الصواب: تستخدم خريطة التدفق لعرض خطوات حل المشكلة (الخوارزمية Algorithm) بشكل مبسط ليفهمها المبرمجون.

٢- (×) الصواب: يحفظ الحاسب قيم المتغيرات في الذاكرة RAM.

٣- (ü) تنويه: يمكنك كتابة الكود المناظر لخريطة التدفق بأية لغة برمجة، كما يمكنك كتابة نفس البرنامج في نفس اللغة بأكثر من صيغة مختلفة، حيث قد تختلف بعض الأوامر المستخدمة، كما قد يختلف عدد سطور الكود.. لهذا من حيث المفهوم، يمكننا اعتبار الكود كائنا، والأوامر المستخدمة وعدد السطور ولغة البرمجة خصائص لهذا الكائن، تماما كما اعتبرنا الحاسب والسيارة والمبنى كائنات.

٤- (×) الصواب: استخدام نافذة الخصائص لتغيير قيم خصائص الزر الموضوع على النموذج، يؤدي إلى تغيير شكل أو سلوك هذا الزر فقط، أما فئة الزر Button Class والأزرار الأخرى الموضوعة على النموذج، فلا تتأثر مطلقا بهذا التغيير.

٥- (ü)

ج ٢:

١- د.

٢- هـ.

تنويه: إذا اعتبرنا صديقك (علي) كائنا، فإن اسمه هو خاصية Property.. لكن في نفس الوقت، اسم صديقك (علي) في نظام معلومات الحكومة الالكترونية أو الشركة التي يعمل بها يعتبر بيانا Data.. لهذا فإن الإجابة الصحيحة هي أ و ب معا، لهذا اخترنا الإجابة رقم هـ.

لاحظ أننا لا نستطيع فعل المثل مع إجابة النقطة رقم ١.. فنحن لا نعرف إن كان معدل الزيادة السكانية خاصة لكائن ما أم لا، لهذا اعتبرناه معلومة فحسب.
٣-أ.

٤-هـ.

تنويه: ليس مذكورا إن كانت المعلومة ستظهر في مربع رسالة أم في أداة، لهذا يمكن عرض الرسالة في مربع رسالة باستخدام الدالة MsgBox، أو عرضها في مربع نص أو لافتة باستخدام الخاصية Text، لهذا اخترنا الإجابة رقم هـ.
٥-د.

ج ٣: ١- لمعرفة النتائج، يجب عليك تتبع خطوات تنفيذ خريطة التدفق، بالقيم: Base = 0 و Increment = 2 و NumberOfTerms = 5 .. دعنا نكتب الخطوات التي ستحدث وعلينا متابعتها على خريطة التدفق:

- في البداية، ستصير للمتغير n القيمة واحد، وستصير قيمة المتغير Term مساوية لقيمة المتغير Base، أي أنها ستساوي صفرا، وستتم طباعة هذه القيمة على الشاشة.

- سيتم فحص الشرط هل $n = \text{NumberOfTerms}$ ، ونظرا لأن n تساوي ١، و NumberOfTerms تساوي ٥ فستكون الإجابة هي لا، وستزيد قيمة n بواحد لتصير ٢، وستزيد قيمة Term بمقدار الزيادة Increment التي تساوي ٢، وبهذا ستصير قيمة Term تساوي ٢، وستطبعها على الشاشة.

- سنكرر الخطوة السابقة مرة أخرى، حيث ستكون إجابة الشرط هي لا، وستزيد قيمة n بواحد لتصير ٣، وستزيد قيمة Term بمقدار ٢، لتصير ٤، وستطبعها على الشاشة.

- سنكرر نفس الخطوة مرة أخرى، حيث ستكون إجابة الشرط هي لا، وستزيد قيمة n بواحد لتصير ٤، وستزيد قيمة Term بمقدار ٢، لتصير ٦، وستطبعها على الشاشة.

- سنكرر نفس الخطوة مرة أخرى، حيث ستكون إجابة الشرط هي لا، وسنزيد قيمة n بواحد لتصير ٥، وسنزيد قيمة Term بمقدار ٢، لتصير ٨، وسنطبعها على الشاشة.

- الآن صارت إجابة الشرط هي نعم (لأن $n = \text{NumberOfTerms} = ٥$)، لهذا سينتهي البرنامج.

لاحظ أنك لست مجبرا على كتابة هذه الخطوات، لكننا كتبناها فقط للإيضاح.

إذن فالنتائج التي سيعرضها هذا البرنامج هي:

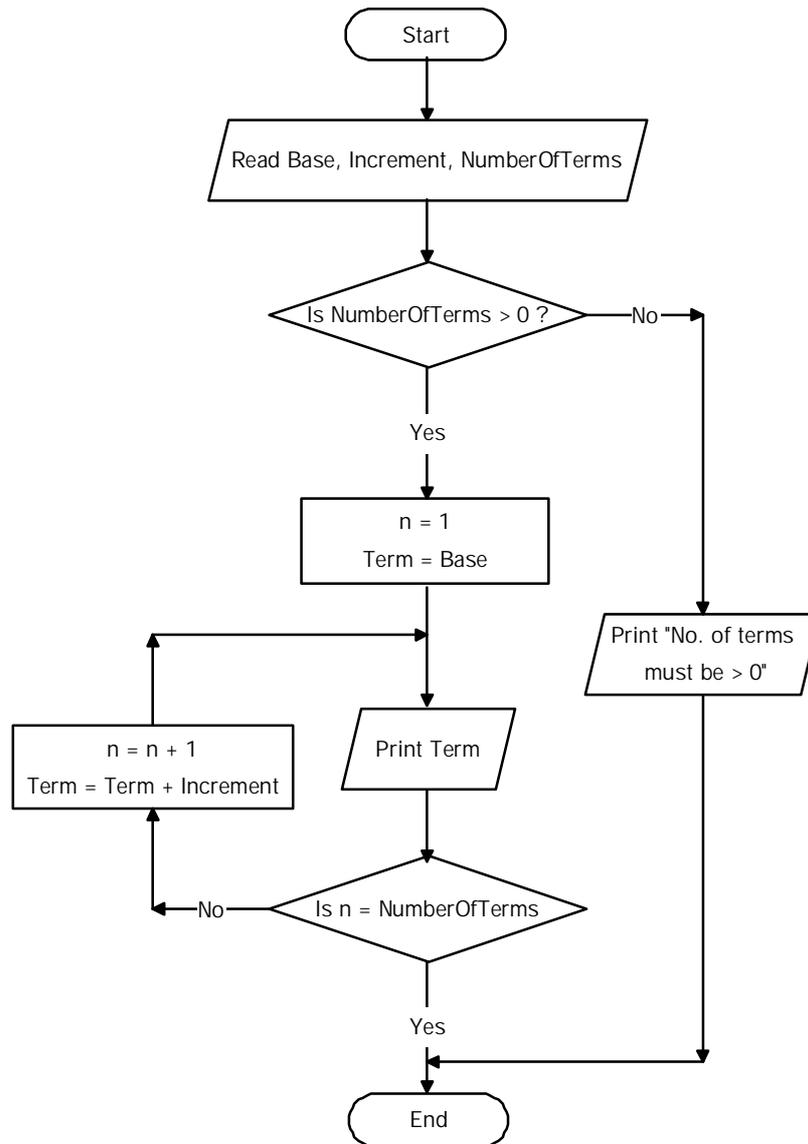
0
2
4
6
8

٢- وظيفة هذا البرنامج كما هو واضح من ناتجه، هي إنتاج متتابعة حسابية أساسها (أول عدد فيها) هو Base، ومعدل الزيادة هو Increment، وعدد الحدود هو NumberOfTerms.

٣- إذا وضع المشغل في المتغير NumberOfTerms صفرا أو عددا سالبا، فلن ينتهي تنفيذ البرنامج أبدا، وستستمر طباعة الأرقام على الشاشة إلى ما لا نهاية.. السبب في هذا أن المتغير n يبدأ بالقيمة ١ وتزيد قيمته باستمرار، وبهذا لن يتحقق أبدا الشرط "Is n = NumberOfTerms?"، لأن NumberOfTerms قيمته صفر أو عدد سالب، أي أنه أصغر من n دائما وأبدا.. ولن تحدث نفس هذه المشكلة إذا وضع المشغل في المتغير Base أو المتغير Increment عددا سالبا، لأن شرط توقف البرنامج لا يعتمد عليهما، لهذا فإن تأثيرهما ينحصر في ناتج المتابعة الحسابية، وليست لدينا مشكلة أن تكون حدود هذه المتابعة موجبة أو سالبة، أو أن تزيد قيمها تصاعديا، أو تتناقص تنازليا.

٤- لتجنب مشكلة وضع صفر أو عدد سالب في المتغير NumberOfTerms، يجب فحص قيمة هذا المتغير أولا، فإن كانت أكبر من صفر واصلنا البرنامج، وإلا فإن

علينا تحذير للمستخدم من أنه أدخل قيمة خاطئة وننهي البرنامج في الحال.. هذه هي خريطة التدفق الجديد بعد التعديل.



٥- هذا هو السودوكود المناظر لخريطة التدفق المعدلة:

- 1- Read the Base, Increment, and NumberOfTerms
- 2- IF the NumberOfTerms is not greater than 0 THEN
 Print an output line saying "No. of terms must be > 0"
 Go to 6
 END IF
- 3- Initialize the term number (n) to 1, and initialize the Term to the value of the Base:
 n = 1
 Term = Base

4- Print an output line showing the Term
5- IF the term No. (n) doesn't equal the NumberOfTerms THEN
 Increase the term No. by 1:
 $n = n + 1$
 Increase the term by the value of the Increment:
 Term = Term + Increment
 Perform steps 3 through 5 until the term No. (n) equals the
 NumberOfTerms
END IF
6- Stop processing

نموذج امتحان رقم ٣
للفصل الثالث الإعدادي، الفصل الدراسي الأول

س ١: ضع العلامة (U) أمام الجملة الصحيحة، والعلامة (x) أمام الجملة الخاطئة، مع تصحيح الخطأ:

- ١- أقصى عدد من الكائنات يمكن إنشاؤه من نفس التصنيف Class، هو ١٠.
- ٢- المفسر أسهل من المترجم في تصحيح الأخطاء.
- ٣- تعتبر لغة فيجيوال بيزيك لغة برمجة شبيهة OOP.
- ٤- يمكن استخدام التغليف Encapsulation لإخفاء الأجزاء السرية من مخططات التدفق.
- ٥- يمكن استخدام نافذة الخصائص لوضع سلسلة حروف فارغة Empty String في اسم الزر.

س ٢: اذكر سبب كل مما يلي:

- ١- استخدام المترجم يجعل تنفيذ البرنامج أسرع من المفسر.
- ٢- توفر الوراثة جهد المبرمج.
- ٣- لا تستخدم الوراثة في البرمجة التقليدية.
- ٤- يوضع الرمز & ضمن حروف النص المكتوب في الخاصية Text الخاصة بالمبين Label.
- ٥- تستخدم العلامة _ في نهاية بعض السطور في كود فيجيوال بيزيك.

س ٣: نريد كتابة برنامج لقراءة طول مستطيل L وعرضه W، ثم حساب محيطه P ومساحة سطحه A، علما بأن:

$$\text{محيط المستطيل} = (\text{الطول} + \text{العرض}) \times ٢$$
$$\text{مساحة سطح المستطيل} = \text{الطول} \times \text{العرض}.$$

لاحظ أن عليك رفض الطول والعرض إذا كانا صفراً أو أصغر من الصفر، وعرض جملة توضح هذا الخطأ للمستخدم.

١- ارسـم خريـطة تدفق لهذا البرنامج.

٢- اكتب الكود الزائف "سودوكود" لهذا البرنامج.

٣- إذا كان لدينا نموذج عليه الأدوات التالية:

الأداة	اسم الأداة	وظيفتها
مربع نص TextBox	TxtLength	يكتب فيه المستخدم طول المستطيل.
مربع نص TextBox	TxtWidth	يكتب فيه المستخدم عرض المستطيل.
لافتة Label	LblPerimeter	تعرض للمستخدم محيط المستطيل.
لافتة Label	LblArea	تعرض للمستخدم مساحة المستطيل.

أ- ما النص الذي سيتم عرضه على الالفتين LblArea و LblPerimeter عند تصميم البرنامج؟

ب- إذا عرفت أن اللافتة قد تكون Static Label أو Dynamic Label، فإلى أي النوعين تنتمي الالفتان LblArea و LblPerimeter؟ ولماذا؟.. وهل نحتاج إلى استخدام النوع الآخر على هذا النموذج؟.. ولماذا؟

ج- إذا أردنا حساب محيط ومساحة المستطيل، فما الأداة المناسبة لتنفيذ هذه العملية؟.. وما الاسم الذي تقترحه لهذه الأداة إذا اعتبرنا أن الوظيفة التي نقوم بها اسمها Calculate.. وما النص الذي ستعرضه هذه الأداة؟.. وما اسم الحدث الذي سيتم فيه تنفيذ هذه العملية الحسابية؟

د- اكتب كود فيجيوال بيزيك الذي يقرأ الطول والعرض ويحسب المحيط والمساحة ويعرضهما للمستخدم.

إجابة نموذج امتحان رقم ٣

ج ١:

١- (×) الصواب: لا يوجد حد أقصى لعدد الكائنات التي يمكن إنشاؤها من نفس التصنيف Class.. الحد الأقصى الوحيد هو قدرة ذاكرة الحاسب على استيعاب هذه الكائنات.

٢- (ü)

٣- (ü)

٤- (×) الصواب: يمكن استخدام التغليف Encapsulation لإخفاء الأجزاء السرية من التصنيف (الفئة) Class.

٥- (×) الصواب: يمكن وضع سلسلة حروف فارغة Empty String في الخاصية Text، ورغم أن هذا سيجعل الزر لا يعرض أي عنوان فهو ليس خطأ برمجياً.. لكن لا يمكن ترك اسم الزر Name فارغاً.

ج ٢:

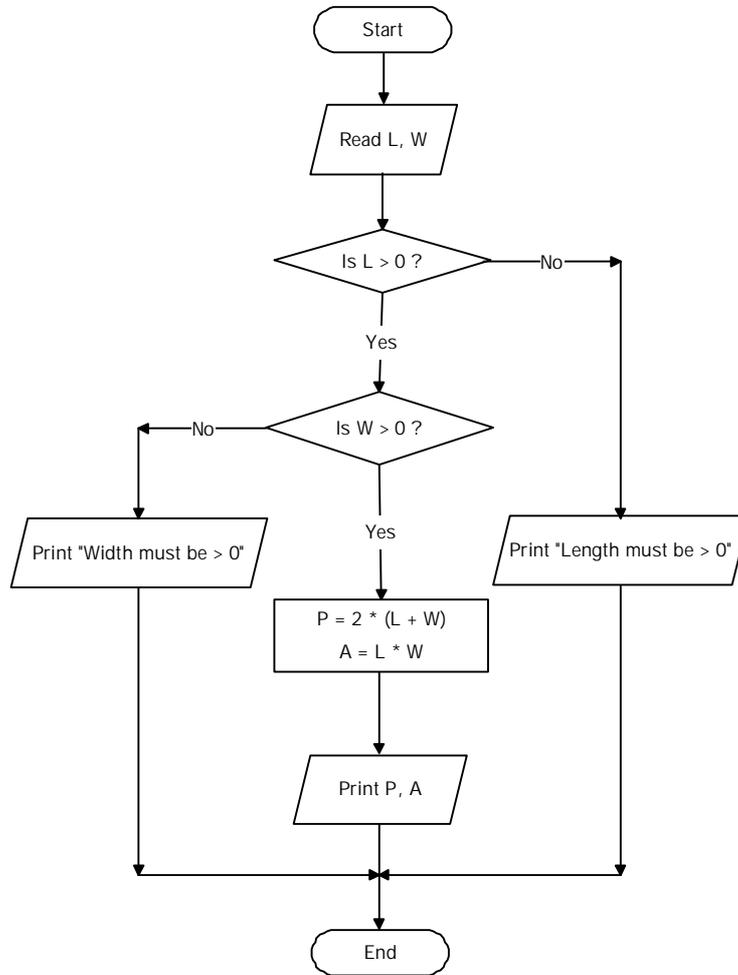
١- استخدام المترجم يجعل تنفيذ البرنامج أسرع من المفسر، لأن المترجم يحول كل أوامر البرنامج إلى لغة الآلة دفعة واحدة، وبعد ذلك يقوم بتشغيل البرنامج، وبهذا فإنه لا يضيع أي وقت عند تنفيذ أي أمر.. أما المفسر فهو يحول كل أمر إلى لغة الآلة قبل تنفيذ هذا الأمر مباشرة وأثناء تشغيل البرنامج، مما يجعل التنفيذ أبطأ.

٢- توفر الوراثة جهد المبرمج، لأنها تتيح له إنشاء فئات مشتقة Derived Classes من الفئات التي كتبها قبل ذلك، وبهذا لا يكرر كتابتها مرة أخرى، مع قدرته على إضافة خصائص ووسائل وأحداث جديدة إلى الفئة المشتقة.

٣- لا تستخدم الوراثة في البرمجة التقليدية، لأن البرمجة التقليدية لا تستخدم الفئات ولا الكائنات، ولهذا لا يوجد شيء يرثه في البرمجة التقليدية!

- ٤- يوضع الرمز & ضمن حروف النص المكتوب في الخاصية Text الخاصة بالمبين Label، لجعل الحرف التالي له حرف اختصار، وبهذا يمكن الانتقال إلى الأداة وتفعيلها مباشرة بضغط الزر Alt مع هذا الحرف من لوحة المفاتيح.
- ٥- تستخدم العلامة _ في نهاية بعض السطور في كود فيجيوال بيزيك، حتى تفهم فيجيوال بيزيك أن السطر التالي هو تكملة لهذا السطر وليس أمراً جديداً.

ج ٣: ١ - خريطة تدفق هذا البرنامج موضحة في الشكل:



٢ - هذا هو الكود الزائف "سودوكود" لهذا البرنامج:

- 1- Read the rectangle's length (L), and width (W)
- 2- IF the rectangle's length (L) is not greater than 0 THEN
Print an output line saying "Length must be > 0"
Go to 6
END IF

- 3- IF the rectangle's width (W) is not greater than 0 THEN
 Print an output line saying "Width must be > 0"
 Go to 6
 END IF
- 4- Calculate the rectangle's perimeter (P), and area (A):
 $P = 2 * (L + W)$
 $A = L * W$
- 5- Print an output line showing P and A
- 6- Stop processing

٣ - أ - ستعرض اللافنتان LblPerimeter و LblArea في البداية نصا فارغا
 .Empty String

ب - اللافنتان LblPerimeter و LblArea لافنتان متغيرتان Dynamic Labels، وذلك لأن النص الذي تعرضانه يتغير أثناء تشغيل البرنامج.. ويمكننا استخدام لافنتين ساكنتين Static Labels على النموذج، واحدة بجوار كل مربع نص، إحداهما تعرض النص "Length" والأخرى تعرض النص "Width"، وبهذا يعرف المستخدم مربع النص المخصص لكتابة طول المستطيل، ومربع النص المخصص لكتابة العرض.

ج - الزر Button هو الأداة المناسبة لتنفيذ عملية حساب المحيط والمساحة، واسمه المناسب هو BtnCalculate، والنص الذي يجب أن يعرضه الزر هو Calculate، والحدث الذي سيتم فيه تنفيذ العملية هو حدث ضغط الزر Click.

د - هذا هو الكود الذي يجب كتابته في معالج حدث ضغط الزر:

```

Dim L As Integer = TxtLength.Text
Dim W As Integer = TxtWidth.Text
If L > 0 Then
  If W > 0 Then
    LblPerimeter.Text = 2 * (L + W)
    LblArea.Text = L * W
  Else
    MsgBox("Width must be > 0")
  End If
Else
  MsgBox("Length must be > 0")
End If

```

نموذج امتحان رقم ٤
للفصل الثالث الإعدادي، الفصل الدراسي الأول

س ١: أجب عما يلي:

١- اذكر أسماء ثلاث أدوات Controls تستخدم في تصميم البرنامج في فيجوال بيزيك.

٢- اذكر أسماء خمس خصائص مشتركة بين هذه الأدوات، ووظيفة كل منها.

٣- كيف يمكن تغيير قيم خصائص هذه الأدوات أثناء تصميم البرنامج؟

٤- كيف يمكن تغيير قيم خصائص هذه الأدوات أثناء تشغيل البرنامج؟

س ٢: ضع العلامة (n) أمام الجملة الصحيحة، والعلامة (x) أمام الجملة الخاطئة، مع تصحيح الخطأ:

١- يؤدي ما يفعله المستخدم بالفأرة ولوحة المفاتيح إلى استدعاء الوسائل Methods الخاصة بالأداة.

٢- البرنامج الذي نكتبه بفيجوال بيزيك دوت نت يحدد للمستخدم الخطوات المتتالية التي يجب عليه اتباعها.

٣- تؤثر قيم الخصائص على شكل الأداة وسلوكها.

٤- بعد إنشاء الكائن يمكن استدعاء وسائله Methods، لكن لا يمكن تغيير قيم خصائصه Properties.

٥- عدد ملاك الأراضي الزراعية التي تزيد مساحتها عن ثلاثة أفدنة هو بيان Data.

س ٣: نريد كتابة برنامج يقرأ طول الشخص Tallness ووزنه Weight، ويستخدم العلاقة التالية لمعرفة إن كان وزن الشخص مناسباً أم لا:

$$\text{الوزن المثالي} = \text{الطول} - 100$$

فإن كان وزن الشخص أقل من الوزن المثالي يخبره البرنامج بأن عليه زيادة وزنه بعدد من الكيلوجرامات يساوي IncKg ليصل إلى الوزن المثالي.

- وإن كان وزنه أكبر من الوزن المثالي، يخبره البرنامج أن عليه إنقاص وزنه بعدد من الكيلوجرامات يساوي DecKg ليصل إلى الوزن المثالي.
- أما إن كان وزنه مثاليا يهنئه البرنامج على هذا.
- لاحظ أن عليك رفض الطول أو الوزن إن كان صفرا أو عددا سالبا.
- أ- ارسـم خريـطة تدفق هذا البرنامج.
- ب- اكتب الكود الزائف "سودوكود" لهذا البرنامج.
- ج- ما الأدوات التي نحتاج لوضعها على النموذج في فيجيوال بيزيك لقراءة الطول والوزن، وما الأداة التي نحتاجها لتنفيذ البرنامج، وكيف نعرض الناتج للمستخدم.. اذكر قيم الخاصيتين Name و Text لكل أداة من تلك الأدوات عند تصميم البرنامج.
- د- اكتب كود هذا البرنامج.
- هـ- ماذا سيحدث لو كان طول الشخص أصغر من ١٠٠ سم؟.. وضـح إجابتك في حالة إذا كان طول الشخص ٩٠ سم ووزنه ٣٠ كجم.
- و- اقترح حلا لمشكلة الشخص الأقصر من ١٠٠ سم.. اكتب الشرط المعدل بكود فيجيوال بيزيك، دون إعادة رسم خريطة التدفق ودون إعادة كتابة كود البرنامج كاملا.

إجابة نموذج امتحان رقم ٤

ج ١:

١- من الأدوات Controls التي تستخدم في تصميم البرنامج في فيجوال بيزيك:

اللافتة (المبين) Label، والزر Button، ومربع النص TextBox.

٢- الجدول التالي يوضح خمس خصائص مشتركة بين هذه الأدوات، ووظيفة كل

منها:

Name	تستخدم لوضع اسم للأداة.
Text	تستخدم لقراءة وتغيير النص الذي تعرضه الأداة.
BackColor	تستخدم لقراءة وتغيير لون خلفية الأداة.
ForeColor	تستخدم لقراءة وتغيير لون النص الذي تعرضه الأداة.
Font	تستخدم لقراءة وتغيير خط النص الذي تعرضه الأداة.

٣- يمكن تغيير قيم خصائص هذه الأدوات أثناء تصميم البرنامج، باستخدام نافذة الخصائص.

٤- يمكن تغيير قيم خصائص هذه الأدوات أثناء تشغيل البرنامج، بكتابة الكود على الصيغة التالية:

ControlName.PropertyName = Value

مثل:

TextBox1.Text = "Hello"

١- (×) الصواب: يؤدي ما يفعله المستخدم بالفأرة ولوحة المفاتيح إلى إطلاق الأحداث Events الخاصة بالأداة.

٢- (×) البرنامج الذي نكتبه بفيجيوال بيزيك دوت نت يستخدم الأدوات والأحداث الخاصة بها، ويترك للمستخدم حرية استخدام الأدوات لتنفيذ وظائف البرنامج بالترتيب الذي يريده هو.

٣- (ü)

٤- (×) بعد إنشاء الكائن يمكن استدعاء وسائله Methods، كما يمكن تغيير قيم خصائصه Properties في أي وقت، بشرط ألا تكون هذه الخصائص للقراءة فقط Read Only Properties، فمثل هذا النوع من الخصائص لا يمكن تغيير قيمته أبداً.

٥- (×) عدد ملاك الأراضي الزراعية التي تزيد مساحتها عن ثلاثة أفدنة هو معلومة Information.

ج ٣: أ - سنستخدم في هذا البرنامج المتغيرات التالية:

- Tallness: سنقرأ في هذا المتغير طول الشخص.

- Weight: سنقرأ في هذا المتغير وزن الشخص.

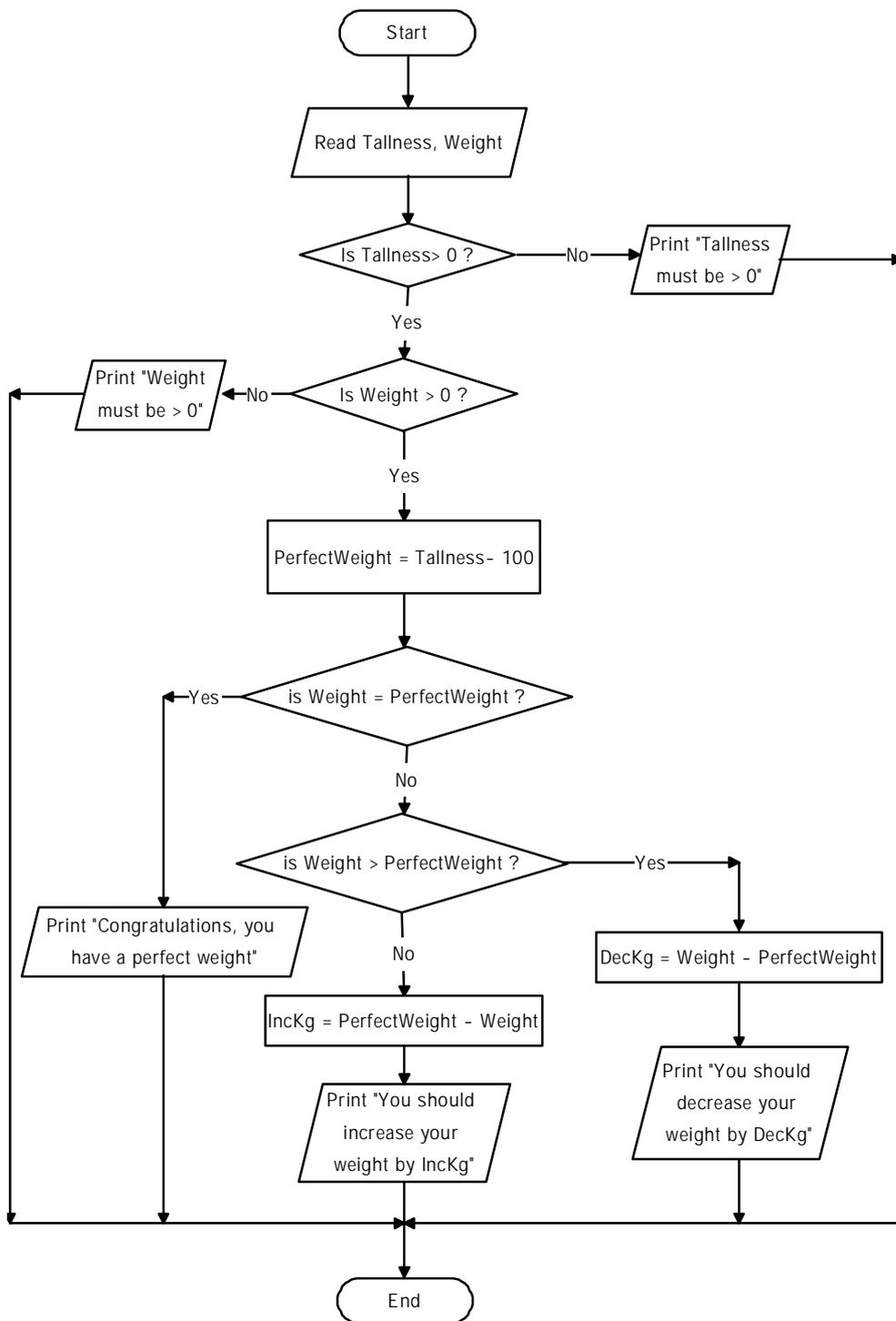
- PerfectWeight: سنحسب في هذا المتغير الوزن المثالي للشخص، تبعاً للعلاقة:

$$\text{PerfectWeight} = \text{Tallness} - 100$$

- DecKg: سنحسب في هذا المتغير عدد الكيلوجرامات التي ينبغي أن ينقصها الشخص من وزنه ليصل إلى الوزن المثالي.

- IncKg: سنحسب في هذا المتغير عدد الكيلوجرامات التي ينبغي أن يزيدها الشخص على وزنه ليصل إلى الوزن المثالي.

هذه هي خريطة التدفق:



ب - هذا هو الكود الزائف "سودوكود" لهذا البرنامج:

- 1- Read the person's Tallness, and Weight
- 2- IF the person's Tallness is not greater than 0 THEN
 Print an output line saying "Tallness must be > 0"
 Go to 6
 END IF
- 3- IF the person's Weight is not greater than 0 THEN
 Print an output line saying " Weight must be > 0"
 Go to 6
 END IF
- 4- Calculate the perfect weight:
 PerfectWeight = Tallness – 100
- 5- IF the person's Weight equals PerfectWeight THEN
 Print an output line saying "Congratulations, you have a perfect weight"
 ELSE
 IF the person's Weight is greater than PerfectWeight THEN
 Calculate the weight to lose:
 DecKg = Weight - PerfectWeight
 Print an output line saying "You should decrease your weight by DecKg"
 ELSE
 Calculate the weight to gain:
 IncKg = PerfectWeight - Weight
 Print an output line saying "You should increase your weight by IncKg"
 END IF
 END IF
- 6- Stop processing

ج- نحتاج إلى مربعي نص لقراءة الطول والوزن، و زر لتنفيذ البرنامج، و لافئة Label لعرض الناتج للمستخدم.. و الجدول التالي يوضح قيم الخاصيتين Name و Text لكل من هذه الأدوات:

Text	Name	الأداة
""	TxtTallness	مربع النص الخاص بالطول
""	TxtWeight	مربع النص الخاص بالوزن
""	LblResult	لافئة الناتج
Calculate	BtnCalculate	الزر

د - هذا كود هذا البرنامج، و يجب كتابته في معالج ضغط الزر BtnCalculate:

```

Dim Tallness As Integer = TxtTallness.Text
Dim Weight As Integer = TxtWeight.Text
If Tallness > 0 Then
  If Weight > 0 Then
    Dim PerfectWeight As Integer = Tallness - 100
    If Weight = PerfectWeight Then
      LblResult.Text = "You have a perfect weight"
    ElseIf Weight > PerfectWeight Then
      Dim DecKg As Integer = Weight - PerfectWeight
      LblResult.Text = "You should decrease your weight by " _
        & DecKg
    Else
      Dim IncKg As Integer = PerfectWeight - Weight
      LblResult.Text = "You should increase your weight by " _
        & IncKg
    End If
  Else
    MsgBox("Width must be > 0")
  End If
Else
  MsgBox("Tallness must be > 0")
End If

```

هـ- لو كان طول الشخص أصغر من ١٠٠ سم، فسيكون الوزن المثالي عددا سالبا، وهذا غير ممكن!.. مثلا في حالة إذا كان طول الشخص ٩٠ سم ووزنه ٣٠ كجم، فإن:

$$\text{الوزن المثالي} = \text{الطول} - ١٠٠ = ٩٠ - ١٠٠ = -١٠$$

وبما أن هذا الوزن المثالي أصغر من وزن الشخص، فإن المتغير DecKg ستكون قيمته:

$$\text{DecKg} = 30 - (-10) = 30 + 10 = 40$$

لهذا ستظهر للمستخدم رسالة تنصحه بإنقاص وزنه بمقدار ٤٠ كجم، وهو ما سيرعب المستخدم حتما لأن وزنه ٣٠ كجم فقط، هذا معناه أنه سيخفتي من على ظهر الأرض، لكن حتى هذا لن يكون كافيا ليصل إلى الوزن المثالي J !!

و- لحل هذا المشكلة، يمكننا قبول الطول، فقط إذا كان أكبر من ١٠٠.. لهذا يجب تعديل أول شرط في الكود السابق ليصير كالتالي:

If Tallness > 100 Then

 باقي الكود بدون أي تغيير '

Else

MsgBox("Tallness must be > 100")

End If

تدريب:

المعادلة التي استخدمناها لحساب الوزن المثالي في المثال السابق مجرد معادلة تقريبية، يسهل على الشخص العادي إجراؤها ذهنياً.. لكن هناك معادلة طبية تستخدم لمعرفة إن كان وزن الشخص مثالياً أم لا.. في هذه المعادلة يتم حساب رقم يسمى "رقم كتلة الجسم" Body Mass Index بالمعادلة التالية:

رقم كتلة الجسم (BMI) = الوزن بالكيلو جرام ÷ مربع الطول بالمتر

ويستخدم الرقم BMI لمعرفة إن كان وزن الشخص ملائماً أم لا تبعاً للشروط التالية:

إذا كان BMI أقل من ١٨,٥ إذن:

الوزن أقل من الطبيعي

غير هذا إذا كان BMI أقل من ٢٥ إذن:

الوزن ملائم جداً للطول

غير هذا إذا كان BMI أقل من ٣٠ إذن:

الوزن زائد

غير هذا إذا كان BMI أقل من ٤٠:

الوزن يشير إلى أن الشخص مصاب بالسمنة

غير هذا:

الوزن يشير إلى أن الشخص مصاب بالسمنة المفرطة

نهاية الشرط.

ونظراً لأن هذه المعادلة والحسابات صعبة على الذهن البشري، فهي ملائمة جداً

لبرنامج فيجيوال بيزيك.. لهذا كتدريب لك، ارسم خريطة تدفق هذا البرنامج، واكتب

الكود الزائف "سودودود" المناظر لها، ثم أنتشئ برنامج فيجيوال بيزيك ينفذها.

نموذج امتحان رقم ٥

للفصل الثالث الإعدادي، الفصل الدراسي الأول

س ١: ضع العلامة (U) أمام الجملة الصحيحة، والعلامة (x) أمام الجملة الخاطئة، مع تصحيح الخطأ:

- ١- يكتشف المفسر أخطاء البرنامج باستخدام الاستشعار الذكي IntelliSense.
- ٢- يمكن أن يظهر رمز النهاية أكثر من مرة في خريطة التدفق.
- ٣- تعديل خريطة التدفق أسهل من تعديل الكود الزائف "سودوكود".
- ٤- يعتبر التغليف أفضل من الوراثة في حماية بيانات الكائن.
- ٥- تستخدم الوسيلة SelectAll لتحديد واختيار كل النص الذي تعرضه اللافتة Label.

س ٢: في كل مجموعة من الاختيارات التالية يوجد اختيار واحد فقط خاطئ.. حدد رقم هذا الاختيار الخاطئ:

- ١- تعتبر فيجيوال بيزيك دوت نت:
 - أ- إحدى لغات بيئة Visual Studio .NET.
 - ب- لغة ذكاء صناعي.
 - ج- لغة برمجة شيئية OOP.
 - د- لغة برمجة عالية المستوى High Level Language.
 - هـ- لغة ذات واجهة مرئية Visual Interface.
- ٢- من البيانات Data التي يستخدمها نظام معلومات شركة الحديد والصلب:
 - أ- أرباح الشركة لهذا العام.
 - ب- اسم كل موظف في الشركة.
 - ج- الراتب الأساسي لكل موظف في الشركة.
 - د- الموزعون المتعاملون مع الشركة.
 - هـ- أنواع الخامات اللازمة لتصنيع الحديد.

٣- يعتبر التصنيف Class:

أ- نوع جديد يقوم المبرمج بتعريفه.

ب- عنصر من عناصر البرمجة الشيئية OOP.

ج- كائن Object تم إخفاء خصائصه باستخدام التغليف Encapsulation.

د- الأساس الذي تقوم عليه الوراثة Inheritance.

هـ- مصدر إنشاء الكائنات Objects.

٤- من الأحداث Events الخاصة بمربع النص TextBox:

أ- Enter . ب- KeyPress . ج- MouseMove .

د- TextChanged . هـ- Modified .

٥- تتحكم الخصائص التالية في مظهر النص المعروض في مربع النص:

أ- WordWrap .

ب- MultiLine .

ج- TextAlign .

د- TextLength .

هـ- PasswordCharacter .

س٣: نريد معرفة الأعداد الثلاثة التي حاصل ضربها يساوي حاصل جمعها، بشرط ألا تكون هذه الأعداد الثلاثة أصفارا.. ارسم خريطة التدفق لحل هذه المسألة.

إجابة نموذج امتحان رقم ٥

ج ١:

١- (×) الصواب: يكتشف المفسر أخطاء البرنامج باستخدام مزيل الأخطاء Debugger، أما الاستشعار الذكي IntelliSense فهي تقنية تساعد المبرمج أثناء كتابة الكود على اختيار أسماء المتغيرات والخصائص والوسائل من قائمة الأعضاء Members List وإكمال الكلمات تلقائياً.

٢- (ü)

٣- (×) الصواب: تعديل خريطة التدفق أصعب، لأنه يؤدي في الغالب إلى إعادة رسم الخريطة من البداية لرسم خطوط التوصيل بين الرموز المعدلة بالشكل الملائم.

٤- (×) الصواب: يستخدم التغليف لإخفاء تفاصيل التصنيف عن المبرمج الذي يستخدم الكائنات المنشأة منه، بينما تستخدم الوراثة لإعادة الاستفادة من تصنيف موجود في إنشاء تصنيف آخر، وتوجد رق في الوراثة لحماية بعض بيانات التصنيف الأب.. لهذا لا يمكن اعتبار الوراثة أفضل من التغليف أو العكس، فلكل منهما وظيفة خاصة به، وهما يعملان معا ضمن تقنية البرمجة الشيئية OOP.

٥- (×) الصواب: تستخدم الوسيلة SelectAll لتحديد واختيار كل النص الذي يعرضه مربع النص TextBox.

ج ٢: الاختيارات الخاطئة هي:

١- ب ٢- أ ٣- ج ٤- هـ ٥- د

ج ٣: ١ - حل هذه المسألة سهل بمجرد النظر، فنحن نعرف أن الأعداد الثلاثة هي ١، ٢، ٣.. لكن الحاسب ليس عنده نظر، ولا يمكن أن يحل مسألة رياضية بالفهولة.. لهذا، لحل أية مسألة برمجيا يجب أن نقدم إلى الحاسوب المعادلة التي يستخدمها في الحل.. وفي هذه المسألة لدينا هذه المعادلة:

$$n1 + n2 + n3 = n1 * n2 * n3$$

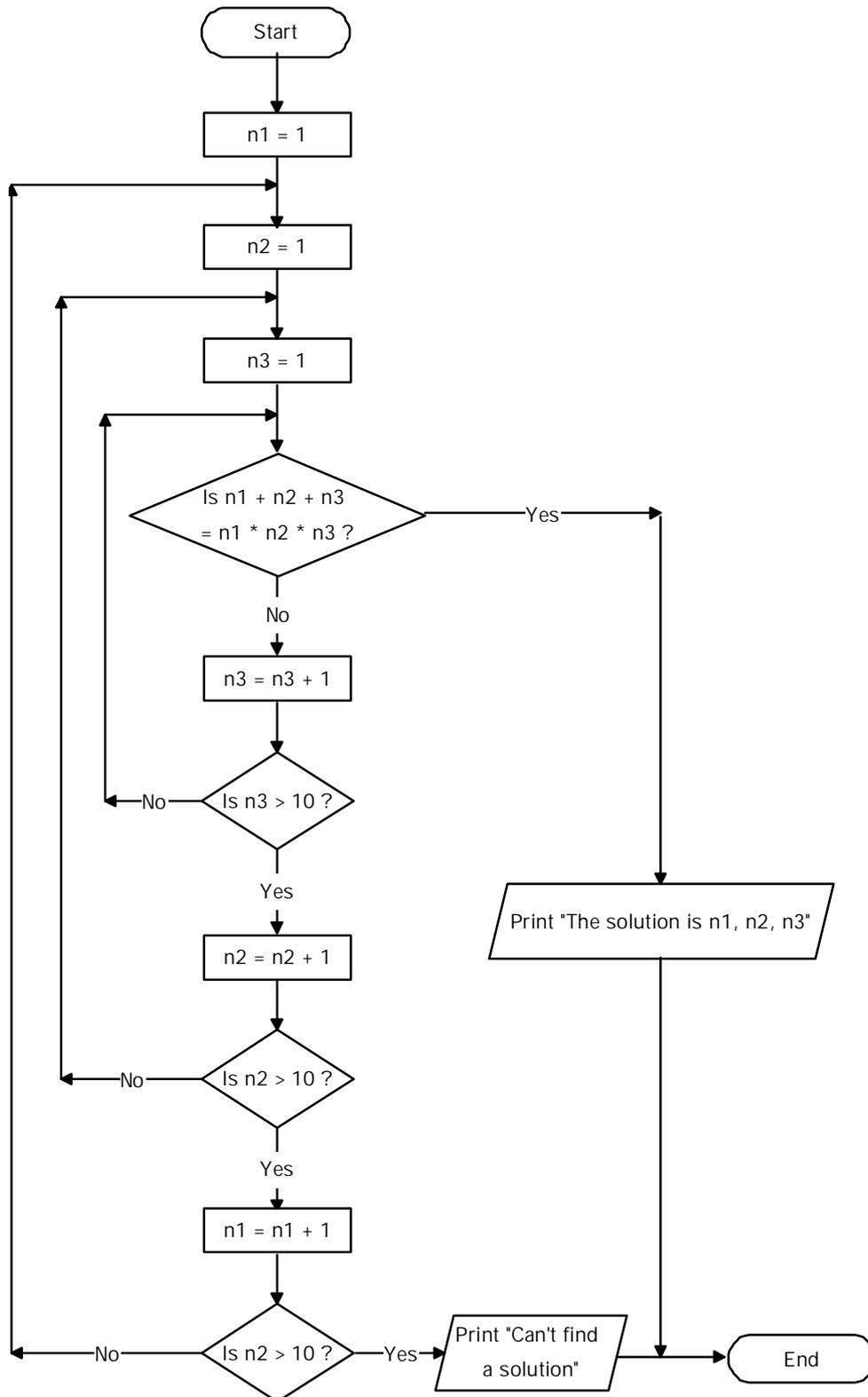
لكننا نعرف من علم الجبر، أن معرفة قيم المجاهيل يتطلب وجود عدد من المعادلات يساوي عدد المجاهيل.. وللأسف، لدينا هنا معادلة واحدة فيها ثلاثة مجاهيل هي الأعداد $n1$ و $n2$ و $n3$ ، وهذا لا يكفي لمعرفة قيمها!

في هذه الحالة يستخدم الرياضيون طريقة تسمى التجريب Iteration أو تسمى المحاولة والخطأ Try and error، حيث يتم افتراض ثلاث قيم للأعداد $n1$ و $n2$ و $n3$ ، وتجربة إن كانت ستحل هذه المعادلة أم لا.. ولكي يوصلنا التجريب إلى نتيجة، يجب أن نجرب تباديل كل القيم الممكنة لهذه الأعداد الثلاثة.. وسنكتفي هنا بحصر نطاق التجريب لكل متغير في الأرقام من ١ إلى ١٠، حتى لا يستمر التجريب إلى ما لا نهاية.. في هذه الحالة سنحتاج إلى ثلاث حلقات تكرار Loops متداخلة، لتجربة كل الاحتمالات المختلفة للأعداد الثلاثة:

- في حلقة التكرار الأولى سنجرب كل قيم $n1$ من ١ إلى ١٠.
- وفي حلقة التكرار الثانية سنجرب كل قيم $n2$ من ١ إلى ١٠.. هذه الجملة ستكون داخل حلقة التكرار الأولى، وبهذا نضمن تجربة كل قيم $n2$ الممكنة لكل قيمة نضعها في $n1$.
- وفي حلقة التكرار الثالثة سنجرب كل قيم $n3$ من ١ إلى ١٠.. هذه الجملة ستكون داخل حلقة التكرار الثانية، وبهذا نضمن تجربة كل قيم $n3$ الممكنة لكل قيمة نضعها في $n2$.

لاحظ أننا بهذه الطريقة سنجرب ١٠ قيم لكل متغير، وهذا يعني أن هناك ١٠٠٠ احتمال مختلف ($١٠ \times ١٠ \times ١٠$).. لاحظ أيضا أن فيجيوال بيزيك ستجد ستة حلول لكنها في النهاية هي نفس الحل، مثل (١، ٢، ٣) و (١، ٣، ٢)

و (٢، ١، ٣)، (٣، ٢، ١) .. وهكذا.. لهذا يجب علينا إيقاف التنفيذ عند الوصول إلى أول حل صحيح.. هذه هي خريطة التدفق لهذه المسألة:



قاموس الكلمات والمصطلحات اللازمة لهذا المنهج

A

Abort	يلغي، ينهي
Access Key	حرف وصول
Active Application	التطبيق الفعال
Active Control	الأداة الفعالة
Active Form	النموذج الفعال
Add	يجمع
Addition	الجمع
Algorithm	خوارزمية
Align	يحاذي
Alignment	محاذاة
Alt Key	زر التبديل
Appearance	المظهر
AppendText	إضافة نص
Application	تطبيق
Argument	معامل
Arrow	سهم
Assembly	لغة التجميع
Assignment	وضع القيمة في المتغير
AutoSize	حجم تلقائي
Average	متوسط

B

BackColor	لون الخلفية
BackGroundImage	صورة الخلفية
BackGroundImageLayout	تخطيط صورة الخلفية
Bold Font	خط سميك
Base Class	الفئة الأساسية، الفئة الرئيسية، التصنيف الرئيسي
BASIC	لغة البيزيك
Beam	شعاع
Binary	ثنائي
Bit	خانة ثنائية
Bitmap	صورة نقطية
Black	أسود
Blue	أزرق
BorderStyle	طراز الإطار، شكل الإطار
Bottom	أسفل
Brown	بني
Browse	يتصفح، يستعرض
Bug	خطأ برمجي
Button	زر
Byte	وحدة ثنائية (بايت)

C

C++	لغة سي بلاس بلاس
Calculate	يحسب قيمة
Cancel	إلغاء
Caret	علامة ضوئية، علامة الكتابة
Case	حالة
Case-insensitive	غير حساس لحالة الأحرف
Case-sensitive	حساس لحالة الأحرف
CD / DVD	قرص ضوئي
CD Driver / DVD Driver	محرك الأقراص الضوئية
Center	مركز
Change	تغيير
Character	حرف
CharacterCasing	حالة الأحرف Casing
Child Class	الفئة الابنة، التصنيف الابن
Class	فئة، تصنيف
Clear	يمحو
Click	يضغط زر الفأرة
Clipboard	لوحة القصاصات
Close	يغلق
COBOL	لغة كوبول
Code	الكود، الشفرة، الشيفرة
Code Editor	محرر الكود
Coder	كاتب الكود
Color	لون
Command	أمر

Command Lines	سطور الأوامر
Comment	تعليق
Common Controls	الأدوات شائعة الاستخدام
Comparison	مقارنة
Compilation	ترجمة الكود إلى لغة الآلة
Compiler	مترجم
Condition	شرط
Context Menu	القائمة الموضعية
Control	أداة تحكم، أداة
Control Key	زر التحكم
Copy	نسخ
Create	ينشئ
Critical	حرج، خطير
Cursor	أيقونة مؤشر الفأرة
Custom	مخصّص
Cut	قص

D

Data	بيان، بيانات
Data Entry	مُدخل البيانات
Data Hiding	إخفاء البيانات
Debug	يصحح الأخطاء
Debugger	مزيل الأخطاء، مصحح الأخطاء، المصحح
Decimal	عشري
Decision	قرار
Decrease	ينقص القيمة
Decrement	إنقاص القيمة
Default	افتراضي، مبدئي
Delete	حذف
Derived Class	الفئة الفرعية، الفئة المشتقة، التصنيف المشتق
Design Time	وقت التصميم
Desk Operating System (DOS)	نظام تشغيل الأقراص (الدوس)
Desktop	سطح المكتب
Developer	مطور
Digital	رقمي
Disabled	غير فعال، معطل
Divide	يقسم
Division	القسمة
Double-click	ينقر مرتين بالفأرة

E

Edit	يحرر
Edit Menu	قائمة التحرير
Effect	تأثير
Else	غير ذلك
Empty String	نص فارغ، سلسلة رموز فارغة
Enabled	فعّال
Encapsulation	التغليف
Enter	يدخل
Enter Key	زر الإدخال
Equal	يساوي
Error	خطأ
Error List	قائمة الأخطاء
Event	حدث
Event Driven Programming	البرمجة الموجهة بالأحداث
Event Handler	معالج الحدث، المستجيب للحدث
Exclamation	تعجب
Executable File	ملف تنفيذي
Exit	يخرج
Explorer	متصفح الويندوز

F

False	خطأ
File	ملف
Firing the Event	إطلاق الحدث
Flash Memory	الذاكرة النقالة
Floppy Disk	القرص المرن
Floppy Driver	محرك الأقراص المرنة
Flow Line	خط الاتجاه
Flowchart	مخطط التنفيذ، خريطة التدفق
Focus	المؤشر
Focused	يمتلك المؤشر
Font	الخط
Font Style	طراز الخط
ForeColor	لون الكتابة، لون الحروف
Form	نموذج
Form Designer	مصمم النماذج
Format	ينسق
Format Menu	قائمة التنسيق
FORTRAN	لغة فورتران
Framework	إطار العمل
Function	دالة، وظيفة
Function Overloading	تعدد صيغ الدالة

G

Go to	اذهب إلى، انتقل إلى
GotFocus	استقبل المؤشر
Gray	رمادي
greater than	أكبر من
greater than or equals	أكبر من أو يساوي
Green	أخضر
Grid	شبكة خطوط

H

Hand	يد
Handles	المقابض، يعالج، يستجيب لـ
Handling the Event	معالجة الحدث، الاستجابة للحدث
Hard Disk	القرص الصلب
Height	ارتفاع
Help	مساعدة، الاستعلام
Hexadecimal	سداسي عشري
Hide	إخفاء
High Level Language	لغة برمجة عالية المستوى
Horizontal	أفقي

I

Icon	أيقونة
If	إذا
IF Statement	جملة الشرط
Ignore	يتجاهل
Image	صورة
ImageAlign	محاذاة الصورة
Import	استيراد
Increase	يزيد القيمة
Increment	زيادة القيمة
Information	معلومة، معلومات
Information System	نظام المعلومات
Inherit	يرث
Inheritance	الوراثة
Initialize	يضع القيمة الابتدائية
Input Unit	وحدة إدخال
Instance	نسخة فورية من الفئة (اختصاراً: نسخة)
Integer	عدد صحيح
Integrated Development Environment (IDE)	بيئة التطوير المتكاملة
Intelligence	ذكاء
intelligent	ذكي
IntelliSense	الاستشعار الذكي
Interface	واجهة
Intermediate Language (IL)	اللغة الوسيطة
Internet	شبكة المعلومات الدولية (إنترنت)

Interpreter		مفسر
Italic Font		خط مائل
	J	
Java		لغة جافا
	K	
Keyboard		لوحة المفاتيح
KeyDown		هبوط الزر
KeyPress		ضغط الزر
KeyUp		صعود الزر
Keywords		الكلمات الأساسية، الكلمات المفتاحية
Kit		حزمة
	L	
Label		لافتة (مبين)
Leave		يترك، يغادر، يخرج
Left		يسار، شمال
less than		أقل من
less than or equals		أقل من أو يساوي
Line		سطر، خط
Linear Programming		البرمجة الخطية، البرمجة التقليدية
Link		رابط، وصلة
List		قائمة
Local Resource		مصدر محلي
Location		مكان، موضع

Lock	يغلق
Login Window	شاشة الدخول
Loop	حلقة التكرار
Lost Focus	فقد المؤشر
Low Level Language	لغة برمجة منخفضة المستوى
Lower	سفلي، حرف صغير

M

Machine Language	لغة الآلة
MaxLength	أقصى طول
Members	أعضاء
Menu	قائمة
Message Box	مربع الرسالة
Method	وسيلة، طريقة
Microsoft Developer Network (MSDN)	شبكة مطوري ميكروسوفت
Microsoft Intermediate Language (MSIL)	لغة ميكروسوفت الوسيطة
Middle	منتصف
Mnemonic	تذكيري
Mnemonic Key	حرف تذكيري
Modified	تمّ تعديله
Mouse	الفأرة
Mouse Pointer	مؤشر الفأرة
MouseDown	هبوط زر الفأرة
MouseEnter	دخول الفأرة
MouseHover	تحليق الفأرة

MouseLeave	مغادرة الفأرة
MouseMove	تحرك الفأرة
MouseUp	صعود زر الفأرة
MouseWheel	تحرك عجلة الفأرة
MsgBox	دالة عرض مربع الرسالة
MultiLine	متعدد الأسطر
Multiply	يضرب
Multiplying	الضرب

N

Name	اسم
Nested If-statements	جمل شرط متداخلة
Nested Loops	حلقات التكرار المتداخلة
Network	شبكة
None	ولا واحد
Normal	طبيعي
Not	ليس

O

Object	كائن
Object Browser	متصفح الكائنات
Object Oriented Programming (OOP)	البرمجة الشيئية (الكائنية)، البرمجة الموجهة بالكائنات
Octal	ثماني
Open	يفتح
Optional	اختياري

OR	أو
Orange	برتقالي
Order	نظام، ترتيب
Output Unit	وحدة إخراج
Output Window	نافذة المخرجات

P

Parameter	معامل
Parent Class	الفئة الأم، التصنيف الأب
Password	كلمة المرور
PasswordChar	حرف كلمة المرور
Paste	لصق
Perform	يؤدي، ينفذ
Performance	أداء
Print	يطبع
Printer	طابعة
Private	خاص
Procedural Programming	البرمجة الإجرائية
Procedure	إجراء
Process	معالجة، عملية
Program	برنامج
Programmer	مبرمج
Programming Language	لغة برمجة
Project	مشروع
Project Resource	مصدر عام للمشروع
Prompt Argument	معامل التوجيه

Properties

خصائص

Properties Window

نافذة الخصائص

Property

خاصية

Pseudo Code

الكود الزائف "سودوكود"

Public

عام

Q

Question

سؤال

R

Read	يقرأ
ReadOnly	للقراءة فقط
Read-Only Memory (RAM)	ذاكرة الوصول العشوائي، ذاكرة مؤقتة
Recent Projects	أحدث المشاريع
Red	أحمر
Redo	إعادة
Regular	عادي
Reinitialize	يعيد وضع القيمة الابتدائية
Remove	يزيل
Report	تقرير
Reset	استعادة الوضع الأصلي
Retry	يعيد المحاولة
Right	يمين
Right-Click	يضغط بزر الفأرة الأيمن
RightToLeft	من اليمين إلى اليسار
Run	يجري، ينفذ، يشغل
Runtime	وقت التشغيل

S

Sample	مثال، نموذج
Save	يحفظ
Scanner	الماسح الضوئي
Screen	شاشة
Script	نص المستند
Scrollbar	المُنزلق، شريط اللف

Select	يحدد، يختار
Select All	تحديد الكل
Selected Control	الأداة المحددة
SelectionLength	طول التحديد
SelectionStart	بداية التحديد
SelectionText	نص التحديد
Shift	زر التحويل
Shortcut key	حرف اختصار
Show	عرض (عكس إخفاء)
Size	المقاييس، الأبعاد، الحجم
Snap Lines	خطوط المحاذاة
Solution	حل، مجموعة مشاريع
Solution Explorer	متصفح الحل، متصفح المشاريع
Sort	يرتب
Sound	صوت
Space	مسافة
SQL Server	خادم قاعدة البيانات
Start menu	قائمة البداية
Start Page	صفحة البداية
Step	خطوة
Stop	يتوقف
Stop processing	يوقف العمليات، ينهي تنفيذ البرنامج
Stretch	يمطّ
Strikout	يعبره خط، مشطوب
String	نص، سلسلة حروف، سلسلة رموز

Sub	إجراء فرعي
Subtract	يطرح
Subtraction	الطرح
Sum / Summation	مجموع
Switch	تحويل، مفتاح
Symbol	رمز
Syntax	طريقة الصياغة، بناء الجملة
Syntax Error	خطأ في الصياغة
System	نظام
System Administrator	مدير النظام
System Colors	ألوان النظام

T

Tab	شريط
Tab Key	زر الجدولة
TabIndex	رقم الوصول
Terminal	طرفي
Text	نص
TextAlign	محاذاة النص
TextBox	مربع نص، مربع كتابة
TextChanged	تغير النص
TextImageRelation	علاقة النص والصورة
TextLength	طول النص
Themes	السمات
Three-Dimensional (3D)	ثلاثي الأبعاد (مجسم)
Tile	تبليط

Title	عنوان
Toggle	تبديل
ToLower	تحويل إلى حروف صغيرة
Toolbox	صندوق الأدوات
Top	أعلى
ToUpper	تحويل إلى حروف كبيرة
True	صحيح

U

UnderLine	تحت خط
Underscore	الشرطة المنخفضة _
Undo	تراجع
Until	حتى
Upper	علوي، حرف كبير
User	المستخدم، مشغل البرنامج

V

Value	قيمة
Variable	متغير
Vertical	رأسيّ (عكس أفقي)
Video	فيديو
View	يعرض، منظر
Visible	مرئي
Visual	بصري
Visual Interface	واجهة مرئية
Visual Studio .NET (VS.NET)	فيجيوال ستديو دوت نت

W

Warning	تحذير
Web Site	موقع على الشبكة الدولية، موقع ويب
While	بينما
White	أبيض
Width	عرض (عكس الطول)
Window	نافذة
Windows Application	تطبيق نوافذ، تطبيق ويندوز
WordWrap	التفاف الأسطر
Yellow	أصفر

Z

Zoom	تحجيم
------	-------