C# - حل تمارين مصفوفات

5-10 محمد أبو فول 10.5.2026

أهداف الدرس

- حل تمارين مصفوفات
- حل تمرين يحتوي على متوالية
- استخدام foreach
- فهم Bubble Sort

مراجعة سريعة - foreach

```
1   foreach (int x in arr)
2       Console.Write(x + " ");
3
```

تصلح: طباعة، مجموع، تصنيف (قراءة فقط)

لا تصلح: تعديل، فرز، تحتاج index

مراجعة سريعة - الأرقام العشوائية

```
// الخطوة 1: إنشاء كائن من فئة الأعداد العشوائية (مرة واحدة فقط!)
Random rnd = new Random();

// الخطوة 2: توليد عدد عشوائي
int num = rnd.Next(0, 51); // [0-50]

rnd.Next(min, max) [min-(max-1)]
rnd.Next(max) [0-(max-1)]

rnd.Next(10,100) [10-99]
rnd.Next(-50,51) [-50-50]
```

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3
الحد الثاني = 5
كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

تحليل سؤال 6 – ماذا نحتاج؟

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3
الحد الثاني = 5
كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 1

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3
الحد الثاني = 5
كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
```

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 1

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3
الحد الثاني = 5
كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
  for (int i = 2; i < 20; i++) arr[i] = 2 * arr[i - 1] - arr[i - 2];
```

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 2

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3

الحد الثاني = 5

كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
  for (int i = 2; i < 20; i++) arr[i] = 2 * arr[i - 1] - arr[i - 2];

  // طباعة جميع الحدود
  Console.WriteLine("The sequence:");
  foreach (int num in arr) Console.Write(num + " ");
  Console.WriteLine();
```

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 3

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3

الحد الثاني = 5

كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
  for (int i = 2; i < 20; i++) arr[i] = 2 * arr[i - 1] - arr[i - 2];

  // طباعة جميع الحدود
  Console.WriteLine("The sequence:");
  foreach (int num in arr) Console.Write(num + " ");
  Console.WriteLine();

  // مجموع الأماكن الزوجية
  int sumEven = 0;
  for (int i = 0; i < 20; i += 2) sumEven += arr[i];

```

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 4

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3

الحد الثاني = 5

كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
  for (int i = 2; i < 20; i++) arr[i] = 2 * arr[i - 1] - arr[i - 2];

  // طباعة جميع الحدود
  Console.WriteLine("The sequence:");
  foreach (int num in arr) Console.Write(num + " ");
  Console.WriteLine();

  // مجموع الأماكن الزوجية
  int sumEven = 0;
  for (int i = 0; i < 20; i += 2) sumEven += arr[i];

  // مجموع الأماكن الفردية
  int sumOdd = 0;
  for (int i = 1; i < 20; i += 2) sumOdd += arr[i];

```

- مصفوفة بحجم 20
- قاعدة: arr[i] = 2*arr[i-1] - arr[i-2]
- مجموعين: فهارس زوجية وفردية
- مقارنة

حل سؤال 6 - 5

السؤال 6: معالجة متوالية داخل مصفوفة

اكتب برنامجًا يبني مصفوفة حجمها 20 حسب القاعدة التالية:

الحد الأول = 3

الحد الثاني = 5

كل حد بعد ذلك يساوي:

الحد الحالي = 2 * الحد السابق - الحد الذي قبله

المطلوب:

1. خزّن المتوالية في مصفوفة .
2. اطبع جميع الحدود .
3. احسب مجموع الحدود الموجودة في الأماكن الزوجية من حيث الفهرس .
4. احسب مجموع الحدود الموجودة في الأماكن الفردية من حيث الفهرس .
5. اطبع أي المجموعين أكبر .

```
static void Main() {
  int[] arr = new int[20];

  // بناء المتوالية
  arr[0] = 3;
  arr[1] = 5;
  for (int i = 2; i < 20; i++) arr[i] = 2 * arr[i - 1] - arr[i - 2];

  // طباعة جميع الحدود
  Console.WriteLine("The sequence:");
  foreach (int num in arr) Console.Write(num + " ");
  Console.WriteLine();

  // مجموع الأماكن الزوجية
  int sumEven = 0;
  for (int i = 0; i < 20; i += 2) sumEven += arr[i];

  // مجموع الأماكن الفردية
  int sumOdd = 0;
  for (int i = 1; i < 20; i += 2) sumOdd += arr[i];

  Console.WriteLine("Sum at even indices: " + sumEven);
  Console.WriteLine("Sum at odd indices: " + sumOdd);

  if (sumEven > sumOdd)
    Console.WriteLine("Even indices sum is larger");
  else if (sumOdd > sumEven)
    Console.WriteLine("Odd indices sum is larger");
  else
    Console.WriteLine("Both sums are equal");
}
```

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.
المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

تحليل سؤال 10 – ماذا نحتاج؟

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.
المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

حل سؤال 10 – العلامات العشوائية

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

```

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

حل سؤال 10 – العلامات العشوائية

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

```

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

حل سؤال 10 – 1+2

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام Array.Sort.
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

```

شرح grade/10

```
9   // تصنيف باستخدام switch
10  foreach (int grade in grades) {
11    switch (grade / 10) {
12      case 10:
13      case 9:
14        categories[0]++;  // A: 90-100
15        break;
16      case 8:
17        categories[1]++;  // B: 80-89
18        break;
19      case 7:
20        categories[2]++;  // C: 70-79
21        break;
22      case 6:
23        categories[3]++;  // D: 60-69
24        break;
25      case 5:
26        if (grade >= 55)
27          categories[3]++;  // D: 55-59
28        else
29          categories[4]++;  // F: 50-54
30        break;
31      default:
32        categories[4]++;  // F: 0-49
33        break;
34    }
35  }
```

- 95 / 10 = 9 → case 9 → A
- 83 / 10 = 8 → case 8 → B
- 72 / 10 = 7 → case 7 → C
- 65 / 10 = 6 → case 6 → D
- 57 / 10 = 5 → case 5 → >=55? → D
- 43 / 10 = 4 → default → F

حل سؤال 10 – 1+2

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام Array.Sort.
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

```

حل سؤال 10 – 3

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام Array.Sort.
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();
```

حل سؤال 10 – 4

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();
```

```
  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }
```

ما هو Bubble Sort?

الفكرة: نقارن كل عنصر مع الذي بجانبه. إذا كان الأول أكبر نبدلهم. نكرر هذا عدة مرات حتى تترتب المصفوفة.

مثال:
[5, 3, 8, 1, 2]

فيديو مع مثال اخر:
https://youtu.be/hoJtA92G0KQ?si=JCX944QzgjAkRTsB

مثال Bubble Sort

[5, 3, 8, 1, 2]

```
الجولة الأولى (i=0):
[5, 3, 8, 1, 2]
[3, 5, 8, 1, 2]
[3, 5, 8, 1, 2]
[3, 5, 1, 8, 2]
[3, 5, 1, 2, 8]
```

```java
// فرز تصاعدي - Bubble Sort
for (int i = 0; i < 4; i++)
  for (int j = 0; j < 4 - i; j++)
    if (grades[j] > grades[j + 1]) {
      int temp = grades[j];
      grades[j] = grades[j + 1];
      grades[j + 1] = temp;
    }
```

مثال Bubble Sort

[3, 5, 1, 2, 8]

```
الجولة الثانية (i=1):
[3, 5, 1, 2, 8]
[3, 5, 1, 2, 8]
[3, 1, 5, 2, 8]
[3, 1, 2, 5, 8]
```

```java
// فرز تصاعدي - Bubble Sort
for (int i = 0; i < 4; i++)
  for (int j = 0; j < 4 - i; j++)
    if (grades[j] > grades[j + 1]) {
      int temp = grades[j];
      grades[j] = grades[j + 1];
      grades[j + 1] = temp;
    }
```

مثال Bubble Sort

[3, 1, 2, 5, 8]

```
الجولة الثالثة (i=2):
[3, 1, 2, 5, 8]
[1, 3, 2, 5, 8]
[1, 2, 3, 5, 8]
```

```java
// فرز تصاعدي - Bubble Sort
for (int i = 0; i < 4; i++)
  for (int j = 0; j < 4 - i; j++)
    if (grades[j] > grades[j + 1]) {
      int temp = grades[j];
      grades[j] = grades[j + 1];
      grades[j + 1] = temp;
    }
```

مثال Bubble Sort

[1, 2, 3, 5, 8]

```
الجولة الرابعة (i=3):
[1, 2, 3, 5, 8]
[1, 2, 3, 5, 8]
```

```java
// فرز تصاعدي - Bubble Sort
for (int i = 0; i < 4; i++)
  for (int j = 0; j < 4 - i; j++)
    if (grades[j] > grades[j + 1]) {
      int temp = grades[j];
      grades[j] = grades[j + 1];
      grades[j + 1] = temp;
    }
```

حل سؤال 10 – 4

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام .Array.Sort
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();
```

```
  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }
```

حل سؤال 10 – 4

السؤال 10: برنامج شامل: تصنيف وفرز ونتائج

اكتب برنامجًا ينتج 50 علامة عشوائية لطلاب في المجال [0, 100] ويخزنها في مصفوفة.

المطلوب:

1. استخدم switch لتصنيف كل علامة إلى :
   - A إذا كانت 90 فما فوق .
   - B إذا كانت من 80 إلى 89 .
   - C إذا كانت من 70 إلى 79 .
   - D إذا كانت من 55 إلى 69 .
   - F إذا كانت أقل من 55 .
2. استخدم مصفوفة عدادات لحساب عدد الطلاب في كل مستوى .
3. اطبع العلامات الأصلية .
4. رتّب العلامات تصاعديًا دون استخدام Array.Sort.
5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - أكثر تصنيف تكرر بين الطلاب.

- مصفوفة علامات بحجم 50
- مصفوفة عدادات بحجم 5 – A, B, C, D, F
- Switch لتصنيف العلامات
- Bubble Sort
- طباعة الإحصائيات

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }

  // طباعة العلامات مرتبة
  Console.WriteLine("\nSorted grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();
```

حل سؤال 10 – 5

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }

  // طباعة العلامات مرتبة
  Console.WriteLine("\nSorted grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // حساب المجموع للمتوسط
  int sum = 0;
  foreach (int grade in grades) sum += grade;

  // النتائج
  Console.WriteLine("\nLowest grade: " + grades[0]);
  Console.WriteLine("Highest grade: " + grades[49]);
```

5. اطبع:
   - أقل علامة.
   - أعلى علامة.
   - المتوسط.
   - عدد الراسبين.
   - طباعة اسم كل تصنيف وعدد الطلاب فيه.
   - أكثر تصنيف تكرر بين الطلاب.

حل سؤال 10 – 5

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();
```

```
  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }

  // طباعة العلامات مرتبة
  Console.WriteLine("\nSorted grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // حساب المجموع للمتوسط
  int sum = 0;
  foreach (int grade in grades) sum += grade;

  // النتائج
  Console.WriteLine("\nLowest grade: " + grades[0]);
  Console.WriteLine("Highest grade: " + grades[49]);
  Console.WriteLine("Average: " + (double)sum / 50);
  Console.WriteLine("Number of failures (F): " + categories[4]);
```

5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - طباعة اسم كل تصنيف وعدد الطلاب فيه.
   - أكثر تصنيف تكرر بين الطلاب.

حل سؤال 5 – 10

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }

  // طباعة العلامات مرتبة
  Console.WriteLine("\nSorted grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // حساب المجموع للمتوسط
  int sum = 0;
  foreach (int grade in grades) sum += grade;

  // النتائج
  Console.WriteLine("\nLowest grade: " + grades[0]);
  Console.WriteLine("Highest grade: " + grades[49]);
  Console.WriteLine("Average: " + (double)sum / 50);
  Console.WriteLine("Number of failures (F): " + categories[4]);

  // طباعة التصنيفات
  string[] catNames = { "A", "B", "C", "D", "F" };
  Console.WriteLine("\n--- Category Breakdown ---");
  for (int i = 0; i < 5; i++)
    Console.WriteLine(catNames[i] + ": " + categories[i] + " students");
```

5. اطبع :
   - أقل علامة .
   - أعلى علامة .
   - المتوسط .
   - عدد الراسبين .
   - طباعة اسم كل تصنيف وعدد الطلاب فيه.
   - أكثر تصنيف تكرر بين الطلاب.

حل سؤال 10 – 5

```
static void Main() {
  Random rnd = new Random();
  int[] grades = new int[50];
  int[] categories = new int[5];  // 0=A, 1=B, 2=C, 3=D, 4=F

  // علامات العشوائية
  for (int i = 0; i < 50; i++) grades[i] = rnd.Next(0, 101);

  // تصنيف باستخدام switch
  foreach (int grade in grades) {
    switch (grade / 10) {
      case 10:
      case 9:
        categories[0]++;  // A: 90-100
        break;
      case 8:
        categories[1]++;  // B: 80-89
        break;
      case 7:
        categories[2]++;  // C: 70-79
        break;
      case 6:
        categories[3]++;  // D: 60-69
        break;
      case 5:
        if (grade >= 55)
          categories[3]++;  // D: 55-59
        else
          categories[4]++;  // F: 50-54
        break;
      default:
        categories[4]++;  // F: 0-49
        break;
    }
  }

  // طباعة العلامات الأصلية
  Console.WriteLine("Original grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // فرز تصاعدي - Bubble Sort
  for (int i = 0; i < 49; i++)
    for (int j = 0; j < 49 - i; j++)
      if (grades[j] > grades[j + 1]) {
        int temp = grades[j];
        grades[j] = grades[j + 1];
        grades[j + 1] = temp;
      }

  // طباعة العلامات مرتبة
  Console.WriteLine("\nSorted grades:");
  foreach (int grade in grades) Console.Write(grade + " ");
  Console.WriteLine();

  // حساب المجموع للمتوسط
  int sum = 0;
  foreach (int grade in grades) sum += grade;

  // النتائج
  Console.WriteLine("\nLowest grade: " + grades[0]);
  Console.WriteLine("Highest grade: " + grades[49]);
  Console.WriteLine("Average: " + (double)sum / 50);
  Console.WriteLine("Number of failures (F): " + categories[4]);

  // طباعة التصنيفات
  string[] catNames = { "A", "B", "C", "D", "F" };
  Console.WriteLine("\n--- Category Breakdown ---");
  for (int i = 0; i < 5; i++)
    Console.WriteLine(catNames[i] + ": " + categories[i] + " students");

  // أكثر تصنيف تكرر
  int maxCat = 0;
  for (int i = 1; i < 5; i++)
    if (categories[i] > categories[maxCat])
      maxCat = i;
  Console.WriteLine("\nMost common category: " + catNames[maxCat]);
}
```

تلخيص الدرس

ما تعلمنا:

- بناء متوالية في مصفوفة
- متى نستخدم foreach و for
- Bubble Sort
- استعمال Switch مع مصفوفة عدادات
- الأرقام العشوائية