المحاضرة 16

◀ دكتور المادة: سمير جعفر

◀ عنوان المحاضرة: حل برامج مذاكرة العملي الأولى.

نظري ☑
عملي ☐

قام الدكتور في هذه المحاضرة بحل نموذجين من نماذج مذاكرة العملي الأولى الأربعة.

النموذج الأول:

اكتب دالة تقوم بحساب المتجه الناتج عن جمع متجهتين ممررين لها كوسطاء، ثم استدعها في برنامج يدخل متجهتين من نفس البعد ويطبع المتجه الناتج عن جمعهما ((حيث أن ناتج جمع متجهتين هو متجه كل عنصر فيهما هو مجموع العنصرين المقابلين له في المتجهتين))

```
class M1{
 static int[] sum(int[] A ,int[] B){          ①
    int[] C = new int[A.length];           ②
    for(int i=0 ; i< A.length ; i++)        ③
       C[i] = A[i] + B[i];
    return C;
              }
public static void main(String args[]){
    int n;                                   ④
  do{ n=Stdin.readInt(); } while (n<=0);
  int[] A = new int[n];          ]
  int[] B = new int[n];          ]→ ⑤

 for(int i=0 ; i<A.length ; i++)       ⑥
  A[i] = Stdin.readInt();
```

```
for(int i=0 ; i<A.length ;i++)
    B[i]=Stdin.readInt();

int[] C = new int[n];          ⑦
C=sum(A,B);                    ⑧

for(int i=0 ; i<C.length ; i++)
   System.out.print(C[i] + " ");
        }
           }
```

شرح البرنامج :

① دالة الجمع تأخذ وسيطين كلٌ منها متجهة ، وهي من النوع static حتى نستطيع استخدامها ضمن البرنامج لاحقاً ، كما أن هذه الدالة ترجع متجهة عناصرها صحيحة.

② تعريف وحجز متجهة من نفس بعد المتجهتين A,B لتكون متجهة جمعهما ، وعند تعريف بعدها يمكن استخدام A.length أو B.length لأن المتجهتين A و B ستكونان من نفس البعد.

③ حلقة for للمرور على جميع العناصر وإسناد مجموع كل عنصرين إلى العنصر المقابل في C ، وبعد الخروج من الحلقة إرجاع المتجهة C.

④ تعريف وإدخال n ليكون بعد المتجهتين.

⑤ تعريف وحجز متجهتين من نفس البعد.

⑥ إدخال قيم عناصر المتجهة.

⑦ تعريف وحجز متجهة ثالثة من نفس البعد لتكون المجموع.

⑧ استدعاء الدالة sum مع تمرير المتجهتين A,B كوسطاء لها ، وإسناد المتجهة الناتجة إلى C ، ثم طباعة عناصر C.

ملاحظة: هنا أمكننا استدعاء الدالة sum مباشرة دون الحاجة إلى اسم الصف حتى لأننا كتبنا البرنامج ضمن الصف نفسه ، أما إذا استخدمنا صفاً جديداً فيجب الاستدعاء

بالشكل: C = M1.sum(A, B) ;

النموذج الثاني :

اكتب دالة تقوم بحساب جداء متجهين تمرر لها كوسطاء، ثم استدعها في برنامج يدخل متجهين من نفس البعد ويطبع ناتج جدائهما.

((حيث جداء متجهين هو مجموع جداء العناصر المتقابلة))

```
class  M2{                                              ①
 static  int  mult (int[] A , int[] B){
    int  c=0 ;              ②
    for(int  i=0 ; i< A.length ; i++)
       C=C+ (A[i] * B[i]);                   ③
    return  c;        }

public static void main (String args[]){
   int  n;
   do{ n=Stdin.readInt();} while(n<=0);
   int[]  A = new  int[n];
   int[]  B = new  int[n];

   for(int  i=0 ; i<A.length ;i++)
      A[i] = Stdin.readInt();
   for(int  i=0 ;i<A.length ;i++)
     B[i] = Stdin.readInt();

   System.out.print(mult(A,B)) ;    }
                           ④          }
```

① دالة لحساب الجداء تمرر لها متجهتان كوسطاء.

② متحول صحيح نعطيه قيمة ابتدائية (0) ((لأن الصفر حيادي الجمع))

③ إسناد جداء المتجهتين إلى العنصر C.

④ استدعاء دالة الجداء من أجل المتجهتين A, B.
ونعمل طباعتها مباشرة لأنها ترجع عدداً صحيحاً.

النموذج الثالث :

اكتب دالة يمرر لها وسيطان : مصفوفة وعدد صحيح n ، تقوم بحساب مجموع عناصر العمود n ، ثم استدعها في برنامج يقوم بإدخال مصفوفة مربعة من البعد m ويطبع مجموع عناصر العمود x من هذه المصفوفة.

```
class M3{
static int sum(int[][] A, int n){      ①
  int sum =0;                ②
  for(int i=0 ; i< A.length ; i++)          ③
     sum= sum + A[i][n];
  return sum; }

public static void main(String args[]){
  int m;
  do{m=Stdin.readInt();}while(m<1);
  int[][] A =new int[m][m];
  for (int i=0 ;i<A.length ;i++)
  for(int j=0 ; j<A.length ;j++)
     A[i][j] = Stdin.readInt();
```

```
int x;                                                          ④
do{x=Stdin.readInt();}while(x<0 || x>=m);
System.out.print(sum(A,x));     ⑤
        }
          }
```

شرح البرنامج :

① دالة لحساب مجموع عناصر عمود ، يمرر لها وسيطان : مصفوفة A وعدد صحيح n يمثل العمود الذي نريد حساب مجموع عناصره، وهذه الدالة ترجع عدداً صحيحاً.

② متحول صحيح نعطيه القيمة الابتدائية (0) ، سنضع ضمنه ناتج مجموع عناصر العمود.

③ حلقة for للمرور على الأسطر فقط ((العمود n مثبت)) ، في كل دخول للحلقة نضيف إلى sum العنصر ذو الدليل (i , n) ، ثم نرجع قيمة sum.

④ المتحول الصحيح x يدل على العمود الذي نريد إيجاد مجموع عناصره ، لذلك عند إدخال قيمته يجب أن نتأكد أنه يدل على أحد أدلة أعمدة المصفوفة التي أدخلناها، وهنا سبب حصر قيمته بين (0) و (m) .

⑤ استدعاء الدالة من أجل المصفوفة A والعمود x ، ويمكن طباعة الناتج عنها مباشرةً لأنها ترجع عدداً صحيحاً.

النموذج الرابع:

اكتب دالة يمرر لها وسيطان : مصفوفة وعدد صحيح n ، تقوم بإرجاع عناصر العمود n على شكل متجهة ، ثم استخدمها في برنامج يقوم بإدخال مصفوفة مربعة من البعد m ، ويطبع المتجهة الناتجة من عناصر العمود x.

```
class M4{
 static   int[]   array(int[][] A, int n){       ①
  int[] a = new  int[A.length];         ②
```

```
for(int i=0;i<A.length;i++)          ③
    a[i] = A[i][n];
    return a;
                }

public static void main(String args[]){
    int m;
    do{m=Stdin.readInt();}while(m<1);
    int[][] A = new int[m][m];
  for(int i=0 ;i<A.length, i++)
   for(int j=0 ; j<A.length ;j++)
     A[i][j] = Stdin.readInt();

  int x;
  do{x= Stdin.readInt();}while(x<0 || x>=m);
  int[] a = new int[A.length];
  a= array(A,x);                      ④

  for(int i=0 ;i<a.length;i++)
   System.out.print(a[i] + " ");  }
                                  }
```

شرح البرنامج:

① دالة عدد وسطائها وسيطان: مصفوفة A وعدد صحيح n يمثل العمود الذي نريد جعل عناصره متجهة، وهذه الدالة ترجع متجهة أعداد صحيحة.

② تصريح وحجز متجهة a عدد عناصرها يساوي أحد بعدي A.
((A مصفوفة مربعة لذلك يمكن كتابة A[0].length أيضاً))

③ حلقة for تقوم بالمرور على أسطر A وتسند قيمة كل عنصر إلى العنصر المقابل له في المتجهة a. ((مع تثبيت العمود n))، ثم يتم إرجاع المتجهة a.

④ استدعاء الدالة array من أجل المصفوفة A والعمود x وإسنادها إلى المتجهة a التي عرّفناها، وهنا لا يمكن طباعة المتجهة مباشرةً، بل يجب كتابة حلقة for للمرور على عناصر المتجهة وطباعتها عنصر عنصر.

ملاحظات:

- في جافا يمكننا إسناد متجهة إلى متجهة أخرى مباشرةً، إلّا أنّ هنا لا يحجز متجهة جديدة، بل يصبح لدينا اسمان لنفس المكان في الذاكرة.
  إذا أردنا تعريف متجهة جديدة فيجب أن نحجز لها مكاناً في الذاكرة ثمّ نسند عناصر المتجهة الثانية إليها عنصر عنصر.

- في حال تعريف متجهة بالشكل التالي مثلاً:

```
int C[] ={2, 1, 0, 5, 7}
```

تم حجز ديناميكي لبعد هذه المتجهة.
((في هذه الحالة بعد المتجهة = 4))

انتهت المحاضرة