

## عمليات الجمع والطرح

### 1. إيعاز الجمع (add)

يقوم هذا الإيعاز بإضافة معامل المصدر إلى معامل الهدف بشرط أن يكون كلا المعاملين بنفس الحجم والصيغة العامة هي:

Add distention , source

#### ملاحظات :

1. لا تتغير قيمة المصدر في هذا الإيعاز ويتم خزن المجموع في معامل الهدف وكأنما العملية تتم بالشكل الآتي .

Distention=distention+source

2. مجموعة المعاملات المقبولة هي نفس المعاملات الخاصة بالإيعاز mov.

مثال 1: اكتب برنامج لجمع معاملي ذاكرة بحجم 32bit , يحتوي المتغير الأول على القيمة 5 والمتغير الثاني على القيمة 3.

The screenshot shows a debugger window with assembly code on the left and a command prompt window on the right. The assembly code is as follows:

```

1 Include Irvine32.inc
2 .data
3 x dword 5
4 y dword 3
5 .code
6 main proc
7 mov ebx,x
8 add ebx,y
9 call dumpregs
10 exit
11 main endp
12 end main
13
14

```

The command prompt window displays the following register values:

```

EAX=770E3C33 EBX=00000005 ECX=00000000 EDI=010C1005
ESI=00000000 EDI=00000000 EBP=0019FB24 ESP=0019FB1C
EIP=010C1021 EFL=00000202 CF=0 SF=0 ZF=0 OF=0 AF=0 PF=0
Press any key to continue . . .

```

### 2. إيعاز الطرح (sub)

يقوم هذا الإيعاز بطرح معامل المصدر من معامل الهدف ونتيجة الطرح تخزن في معامل الهدف ولهو نفس المعاملات الخاصة بالإيعازات (mov,add). والصيغة العامة هي:

Sub distention , source

مثال 2: اكتب برنامج لطرح معاملي ذاكرة من نوع word يحتوي المتغير الأول القيمة 10 والمتغير الثاني يحتوي القيمة 5.

The screenshot shows a Global Scope window with the following assembly code:

```

1  Include Irvine32.inc
2  .data
3  x word 10
4  y word 5
5  .code
6  main proc
7  mov ax,x
8  sub ax,y
9  call dumpregs
10 exit
11 main endp
12 end main
13

```

Overlaid on this is a Windows command prompt window titled "C:\Windows\system32\cmd.exe" showing the following register values:

```

EAX=76120005  EBX=7FFD3000  ECX=00000000  EDX=00041005
ESI=00000000  EDI=00000000  EBP=0014FA34  ESP=0014FA2C
EIP=00041022  EFL=00000206  CF=0  SF=0  ZF=0  OF=0  AF=0
Press any key to continue . . .

```

### إيعازات الزيادة والنقصان (INC,DEC)

يقوم إيعاز الزيادة (inc) بالاضافة واحد إلى معام معين بينما إيعاز النقصان (dec) يقوم بنقصان المعامل بمقدار واحد .

والصيغة ألعامه لهما هي:

Inc reg / mem

Dec reg / mem

**ملاحظة:** لا يمكن إن يأخذ هذان الإيعازان قيمة ثابتة .

**مثال 3:** اكتب برنامج يقوم بقراءة متغير من نوع word قيمة 1000h ثم قم بزيادة قيمة هذا المتغير بمقدار واحد ونقله إلى مسجل bx ومن ثم نقصان قيمة bx بمقدار واحد مع مراعاة طباعة الناتج بعد التغير.

The screenshot shows a Visual Studio IDE with an assembly file named 'al Scope'. The code is as follows:

```

1  Include Irvine32.inc
2  .data
3  x word 1000h
4  .code
5  main proc
6  inc x
7  mov bx,x
8  dec bx
9  call dumpregs
10 exit
11 main endp
12 end main
13
14

```

Overlaid on the IDE is a Windows command prompt window titled 'C:\Windows\system32\cmd.exe'. It displays the following register values:

```

EAX=76123C33  EBX=7FFD1000  ECX=00000000  EDI=00000000
ESI=00000000  EDI=00000000  EBP=002DFE88  ESP=002DFE
EIP=00161025  EFL=00000206  CF=0  SF=0  ZF=0  OF=0

```

Below the register values, it says 'Press any key to continue . . .'

**ملاحظة:**

1. إذ طلب بالسؤال زيادة قيمة الناتج بمقدار اثنين نكرر inc مرتين.

```
Inc ebx
Inc ebx
```

2. إذ طلب بالسؤال نقصان قيمة الناتج بمقدار اثنين نكرر dec مرتين.

```
Dec ebx
Dec ebx
```

#### 4. إيعاز الضرب (Mul)

إيعاز الضرب يأخذ قيمة واحدة إما مسجل أو معامل ذاكرة والصيغة العامة لهو:

Mul reg / mem
---------------

**ملاحظات:**

1. إحدى القيم يجب خزنها ضمن المسجل A وحسب النوع البياني لتلك القيمة أي إذا كان نوع بياني byte تكون ضمن al وإذا كان نوع بياني word ضمن ax وإذا كان نوع بياني dword ضمن eax.

2. إما القيمة الثانية تخزن ضمن مسجلات الأخرى أو بمتغير.

مثال 4: اكتب برنامج لضرب القيمة  $2 \times 3$  علما إن المتغيرات من نوع byte .

```

1  Include Irvine32.inc
2  .data
3  x byte 3
4  y byte 2
5  .code
6  main proc
7  mov al,x
8  mul y
9  call dumpregs
10 exit
11 main endp
12 end main
13
14

```

Command Prompt Output:

```

EAX=770E0006 EBX=7FFD8000 ECX=00000000 EDX=01391005
ESI=00000000 EDI=00000000 EBP=0031FBA0 ESP=0031FB98
EIP=01391020 EFL=00000206 CF=0 SF=0 ZF=0 OF=0 AF=0 PF=1
Press any key to continue . . .

```

حل آخر بدون تعريف المتغيرات

```

1  Include Irvine32.inc
2  .code
3  main proc
4  mov al,3
5  mov bl,2
6  mul bl
7  call dumpregs
8  exit
9  main endp
10 end main
11
12
13

```

Command Prompt Output:

```

EAX=770E0006 EBX=7FFDF002 ECX=00000000 EDX=00061005
ESI=00000000 EDI=00000000 EBP=001EF964 ESP=001EF95C
EIP=0006101B EFL=00000206 CF=0 SF=0 ZF=0 OF=0 AF=0 PF=1
Press any key to continue . . .

```

### بعض الاجراءات التي يمكن استدعاء باستخدام الايجاز CALL

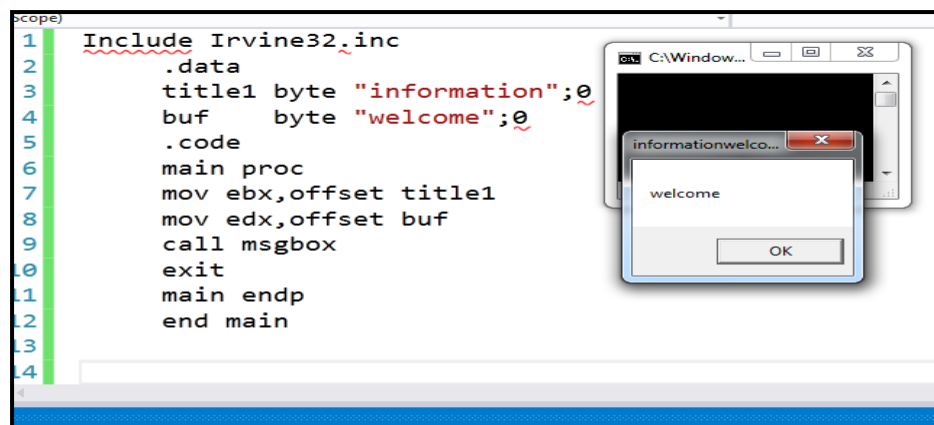
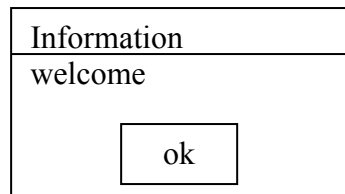
#### 1. الإجراء (crlf)

يقوم بوضع المؤشر عندي بداية السطر التالي في شاشة console إي إن عمل هذا الإجراء هو نزول سطر جديد.

#### 2. الإجراء (msgbox)

يقوم هذا الإجراء بعرض صندوق الرسالة ذات (1 button) بعنوان اختياري حيث يتم نقل إزاحة سلسلة لرموز (string) التي ستظهر داخل صندوق الرسالة في مسجل (edx) ويتم نقل إزاحة

سلسلة لرموز التي تمثل عنوان الصندوق في مسجل (ebx) وفي حالة عدم تخصيص عنوان الصندوق يتم وضع zero في مسجل (ebx).  
مثال 5: اكتب برنامج لتصميم الشكل التالي:

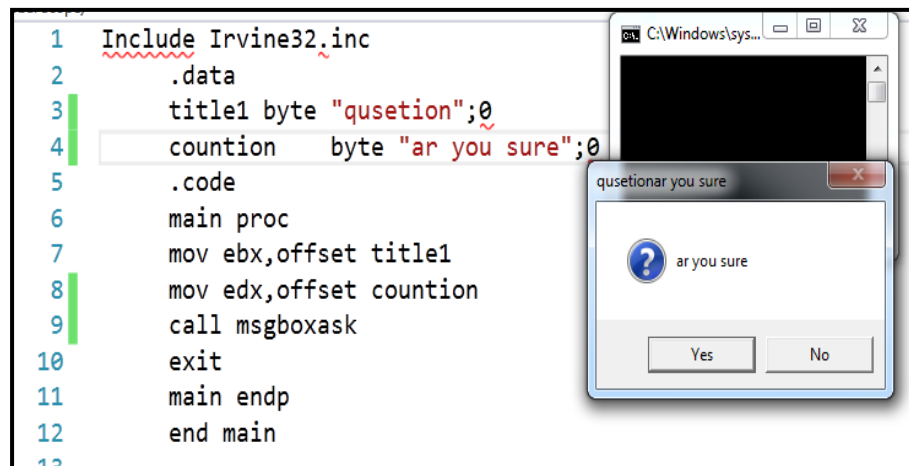


## 2. الإجراء (msgboxAsk)

عمل هذا الإجراء يكون مشابه للإجراء السابق مع اختلاف بسيط وهو عرض صندوق الرسالة مع وجود 2button.

مثال 6: اكتب برنامج لتصميم الشكل التالي :

Question	
Ar you sure	
Yes	No



```

1  Include Irvine32.inc
2  .data
3  title1 byte "qusetion";0
4  countion byte "ar you sure";0
5  .code
6  main proc
7  mov ebx,offset title1
8  mov edx,offset countion
9  call msgboxask
10 exit
11 main endp
12 end main
13

```

واجب:

اكتب برنامج لحل المعادلة الاتيه  $var4 = (var1 + var2) * var3$  علما ان المتغير الأول يحتوي القيمة (2) والمتغير الثاني يحتوي القيمة (3) والمتغير الثالث يحتوي القيمة (5).