كلية العلوم

القسم : **الرياضيات**

السنة : **الرابعة**

المادة : ذكاء صنعي

المحاضرة : **السادسة /عملي/**

{{ مكتبة A to Z }}

Facebook Group : مكتبة A to Z



يمكنكم طلب المحاضرات برسالة نصية (SMS) أو عبر(What's app-Telegram)على الرقم 0931497960

# عملي ذكاء صنعي

## الجلسة السادسة

م.ريم بصل

م.ريم رقبو

**أكتب برنامج لإضافة قائمة في نهاية قائمة أخرى.**
فمثلاً union([3,4,5,3],[6,5,3]) يكون العنصر الموجود في القائمتين غير مكرر فالناتج هو [3,4,5,6].

```
append([],L,L).
append([H|T],L,[H|A]):-append(T,L,A).
```

نلاحظ أن شرط التوقف إذا كانت القائمة الأولى فارغة ومهما كانت القائمة الثانية فإنها تبقى كما هي، إما في الاستدعاء الذاتي فإننا نقوم بنقل العناصر من القائمة الأولى إلى القائمة الثالثة الناتجة إلى أن تنتهي كل عناصر القائمة الأولى.

**لو استدعينا:**

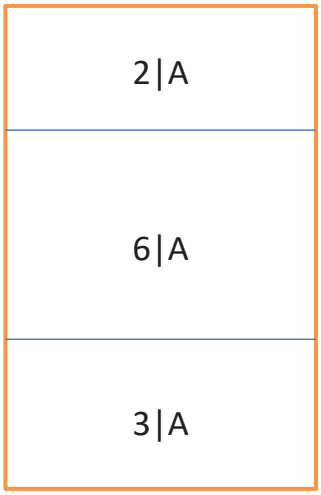
**append([3,6,2], [90,7] ,L).**

**R1:fail**
**R2: H=3 T=[6,2] L=[90,7] append([6,2],[90,7],A)**

**call append([6,2],[90,7],A).**
**R1: fail**
**R2: H=6 T=[2] L=[90,7] append([2],[90,7],A)**

**call append([2],[90,7],A).**
**R1:fail**
**R2: H=2T=[] L=[90,7] append([],[90,7],A)**

**call append([],[90,7],A).**
**R1: L=[90,7].**



**بعد أن نصل إلى شرط التوقف فان القائمة الناتجة سوف تحتوي كل عناصر القائمة الثانية وبعدها نقوم باستخراج العمليات الموجودة في المكدس وهي أضافة 2 , 6 , 3 فتكون القائمة الناتجة هي:**
**L=[3,6,2,90,7]**

**أكتب برنامج لإيجاد أصغر عنصر في القائمة.**

```
min([X],X):-!.
min([H,X|T],A):-H<X,!,min([H|T],A).
min([_|T],A):-min(T,A).
```

هنا توجد ثلاثة شروط الشرط الأول شرط التوقف إذا كانت القائمة تحتوي على عنصر واحد فأننا نرجع هذا العنصر كأصغر واحد الشرط الثاني يقارن العنصرين الأول والثاني إذا كان العنصر الأول أصغر من الثاني نحتفظ به ويهمل العنصر الثاني من القائمة وهكذا إلى أن تنتهي القائمة أما الشرط الثالث فإذا كان العنصر الثاني أصغر من الأول فيهمل العنصر الأول من القائمة وهكذا الحال بالنسبة لبقية عناصر القائمة.

**أكتب برنامج لإيجاد حاصل ضرب عنصر في قائمة.**

```
mult(_,[],[]).
mult(X,[H|T],[S|L]):-S is X*H,mult(X,T,L).
```

فعند استدعاء mult(3,[4,2,5],L) يكون الناتج L=[12,6,15].

قاعدة المعرفة لختبار اذا العنصر اخر القائمة ام لا:

```
last([X],X):-!.
last([_|T],X):-last(T,X).
```

**أكتب برنامج لترتيب عناصر قائمة ترتيب تصاعدي.**

```
sort([],[]).
sort(L,[H|A]):-min(L,H),del(H,L,L1),sort(L1,A).
```

**ال del هو برنامج حذف العنصر لمرة واحدة.( موجود بالجلسة السابقة).**
**ال min هو برنامج لايجاد اصغر عنصر بالقائمة.**

فكرة برنامج الترتيب تعتمد على أيجاد أصغر عنصر في القائمة ومن ثم يوضع هذا العنصر في بداية القائمة الناتجة ومن ثم يحذف من القائمة الأصلية في هذه الحالة وضعنا أصغر عنصر في بداية القائمة ومن ثم نكرر نفس العملية من جديد على العناصر المتبقية في القائمة وهكذا إلى أن تفرغ جميع العناصر من القائمة وتكون القائمة الناتجة قائمة مرتبة من الصغير إلى الكبير.

**ملاحظة(في الامتحان يجب كتابة برنامج ال delوبرنامج min لأننا استخدمناهم ضمن قاعدة في برنامج sort.**

**وظيفة:**
**أكتب برنامج لإضافة عدد صحيح في نهاية القائمة.**

مكتبة A to Z