1.4التحليل الدلاليSemantic Analysis

بعد أن يقوم المحلل اللفظي بتقسيم البرنامج إلى رموز (Tokens) ويتم التأكد من صحة البنية النحوية في مرحلة التحليل النحوي، تأتي المرحلة الثالثة :التحليل الدلالي، وهي مسؤولة عن التأكد من صحة المعاني والقيم المستخدمة في البرنامج.

Semantic Analysisهي المرحلة التي تتحقق من أن البرنامج ليس فقط صحيحًا من حيث القواعد النحوية، بل أيضًا من حيث المعاني والاستخدامات الصحيحة للبيانات حيث ان الهدف من التحليل الدلالي هو الكشف عن الأخطاء التي لا يمكن اكتشافها في التحليل النحوي.

مثال:

```
int x;
x = "hello"; // صحيح نحويًا لكن خاطئ دلاليًا
```

1.1.4موقع التحليل الدلالي ضمن مراحل المترجم

تأتي مرحلة التحليل الدلالي مباشرةً بعد التحليل النحوي.(Syntax Analysis) في هذه المرحلة يقوم المحلل الدلالي (Semantic Analyzer) بفحص البرنامج المصدري للتأكد من خلوّه من الأخطاء الدلالية.(Semantic Errors)

الهدف الأساسي: ضمان أن البرنامج ليس فقط صحيحًا من الناحية النحوية، بل أيضًا صحيح من الناحية الدلالية (أي أن المعنى المنطقي للتعليمات متوافق مع قواعد اللغة البرمجية).

2.1.4 أهداف التحليل الدلالي

1. التحقق من أنواع البيانات.(Type Checking) التأكد من أن الأنواع متوافقة في العمليات والإسناد. مثال: لا يمكن إسناد سلسلة نصية (String) إلى متغير من النوع عدد صحيح (Integer).
2. التأكد من تعريف المتغيرات قبل استخدامها.
3. إدارة نطاقات المتغيرات والدوال.(Scope)
4. التحقق من توافق المعاملات في استدعاء الدوال وعددها وأنواعها.
5. تحديث جدول الرموز (Symbol Table) بالمعلومات الجديدة.
6. إعداد الأخطاء الدلالية للتعامل معها قبل مرحلة توليد الكود

3.1.4 تمريرات التحليل الدلالي(Semantic Passes)

يتم تنفيذ التحليل الدلالي عادةً عبر تمريرين:(Two Passes)

التمرير الأول:	التمرير الثاني:
التحقق من تعريف الأسماء.(Name Definition) فحص وجود أسماء مكررة (Duplicate Names). التأكد من سلامة التصريحات (Consistency Checks).	استكمال تحليل النطاقات (Scope Analysis). التحقق من الأسماء غير المعرفة (Undefined Names). إجراء فحص الأنواع (Type Analysis).

4.1.4 أدوات التحليل الدلالي(Semantic Passes)

أ. جدول الرموزSymbol Table

- يخزن معلومات عن المتغيرات، الثوابت، الدوال، ونطاقها وأنواعها.

مثال

Name	Type	Scope	Value
x	int	global	10
y	float	local	3.14
foo	Function	global	-

ب. Type Checking

- التحقق من توافق الأنواع في التعبيرات والدوال.

مثال

```
int a = 5
float b = 2.5;
a = a + b; // خطأ دلالي بدون تحويل صريح
```

ج. إدارة النطاقScope Management

- التأكد من أن المتغيرات تُستخدم ضمن نطاقها الصحيح.

مثال: متغير محلي في دالة لا يمكن الوصول إليه خارج الدالة.

4.1.4 الأخطاء الدلالية ةSemantic Errors

التفسير	مثال	نوع الخطأ
استخدام نوع غير متوافق	`int x; x = "hello";`	Type Mismatch
المتغير لم يتم تعريفه	`y = 10;`	Undeclared Variable
المتغير خارج نطاقه	`{ int x; } x = 5;`	Scope Errors
عدد أو نوع المعاملات لا يتوافق مع التعريف	`foo(5, "a");`	Function Call Errors

5.1.4 خطوات التحليل الدلالي

1. استلام **شجرة التحليل النحوي**.**(Parse Tree / AST)**
2. زيارة كل عقدة في الشجرة والتحقق من:
   - تعريف المتغيرات والدوال.
   - توافق الأنواع في التعبيرات.
   - نطاق المتغيرات والمعاملات.

3. تحديث **جدول الرموز** بالمعلومات الجديدة.
4. إخراج تقرير بالأخطاء الدلالية (إن وجدت).
5. تجهيز المعلومات للمرحلة التالية :**توليد الكود**.**(Code Generation)**

المثال 1: خطأ نوع البيانات

```
int x;
x = "hello"; // خطأ دلالي
```

المثال 2: متغير غير معرف

```
y = 10; // خطأ دلالي
```

المثال 3: استخدام خارج النطاق

```
{
  int x = 5;
}
x = 10; // خطأ دلالي
```

6.1.4العلاقة بين التحليل النحوي والدلالي

الأخطاء المكتشفة	الهدف	المرحلة
Syntax Errors	التحقق من صحة تركيب البرنامج	Syntax Analysis
Semantic Errors	التحقق من صحة المعاني والأنواع	Semantic Analysis