

المحاضرة 09 · التحكم بالتكرار

09

الحلقات: كَرّر بذكاء ولا تقع في اللانهاية

إذا احتجت حساب متوسط ألف علامة، هل تكتب ألف تعليمة؟

تطبيق عملي

أمثلة C++17

رسم توضيحي

فهم قبل الحفظ

مستوحاة من برمجة 1 · صفحات المصدر 90-108 · إعداد تربوي: م.براءة علي رحيمو

++C

01 · المشكلة أولاً

قبل التعريف... لماذا نحتاج هذه الفكرة؟

إذا احتجت حساب متوسط ألف علامة، هل تكتب ألف تعليمة؟

مشهد من الواقع

آلة مبيعات تقرأ كل فاتورة وتجمعها. تكرار نفس التعليمات يدويًا غير عملي ويصعب التصحيح.

أين المشكلة في الطريقة الساذجة؟

من دون حلقة نكرر الشيفرة؛ ومع حلقة غير مضبوطة قد نعلق التطبيق أو نجمع قيمة زيادة واحدة.

كيف تساعدنا الفكرة؟

نختار for عندما نعرف نمط العداد، وwhile عندما يعتمد التوقف على شرط، وdo-while عندما يجب تنفيذ الجسم مرة.

شبهها بما تعرفه: حلقة التدريب الرياضي: كرر التمرين 10 مرات وعدّ، ثم توقف. العدّ والشرط هما التحكم.

لا تحفظ كلمة جديدة قبل أن تعرف المشكلة التي وُجدت لتساعد في حلها.

02 · النموذج العقلي

اجعل المفهوم مرئيًا

ثلاث أفكار رئيسية ثم مسار واضح يربط بينها.

for

تهيئة ← شرط ← جسم ← تحديث، مناسبة كثيرًا للتكرار المحسوب.

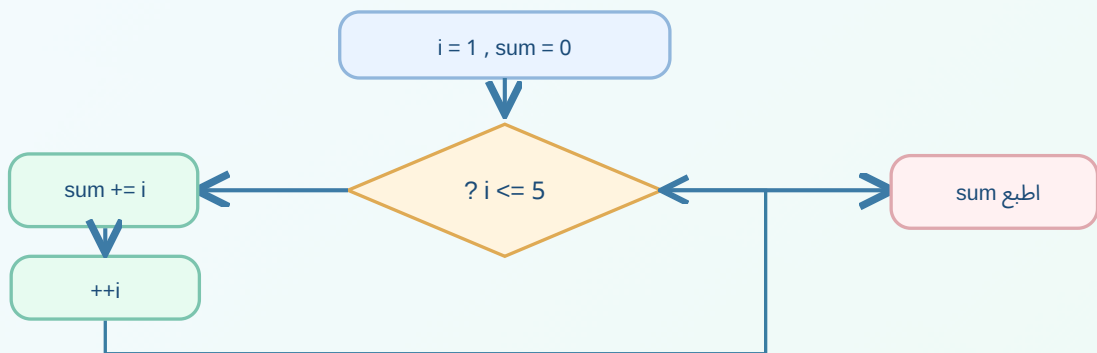
while

قد لا تنفذ إطلاقًا إذا كان الشرط خاطئًا من البداية.

do while

تنفذ الجسم مرة واحدة على الأقل، ثم تفحص الشرط.

الرسم التوضيحي: كيف تعمل الفكرة؟



السؤال الأهم: إذا احتجت حساب متوسط ألف علامة، هل تكتب ألف تعليمة؟

03 · مثال عملي أول

شغل وافهم كل سطر

مجموع الأعداد من 1 إلى 5

C++17 / source.cpp

```
#include <iostream>
using namespace std;
int main() {
    int sum = 0;
    for (int i=1; i<=5; ++i) {
        sum += i;
    }
    cout << sum;
}
```

الخرج المتوقع 15

لماذا كتبنا البرنامج بهذا الشكل؟

sum خزان تجميع يبدأ بالصفر؛ كل تكرار يضيف i ثم يزيد i بواحد.

💡 قبل تجربة البرنامج: حاول توقع النتيجة على الورق، ثم قارن توقّعتك بالخرج الفعلي.

04 · مثال إضافي

نفس المبدأ، حالة استخدام مختلفة

تكرار داخلي: جدول صغير

• • • C++17 / source.cpp

```
#include <iostream>
using namespace std;
int main() {
    for (int row=1; row<=2; ++row) {
        for (int col=1; col<=3; ++col)
            cout << row*col << ' ';
        cout << '\n';
    }
}
```

الخروج المتوقع 1 2 3 / 2 4 6

ما الذي نتعلمه من المثال الثاني؟

الدورة الداخلية تنهي الأعمدة لكل صف ثم تتغير قيمة row في الخارجية.

جرب تغيير قيمة واحدة فقط، ثم توقع أثرها: هل تتغير مسار البرنامج أم الناتج فقط؟

05 · مختبر الأخطاء

الخطأ ليس للحفظ... اكتشف سببه

قارن الكود الخاطئ بالصحيح ولا تكتفِ بمعرفة الإجابة.

الكود الذي يحتاج انتباهًا

```
int i=1; while(i<=5) cout << i;
```

تصحيح وتفسير

```
int i=1; while(i<=5) { cout << i; ++i; }
```

لماذا يُعد هذا خطأ؟

بدون تغيير العداد تبقى المقارنة صحيحة فتحدث حلقة لا نهائية.

جرب بنفسك

تحدي 01

تتبع sum في كل تكرار عند $n=4$.

تحدي 02

اكتب while تقبل أعدادًا حتى يكتب المستخدم صفرًا.

تحدي 03

استخدم continue لتجاوز مضاعفات 3، و break لإنهاء مبكر.

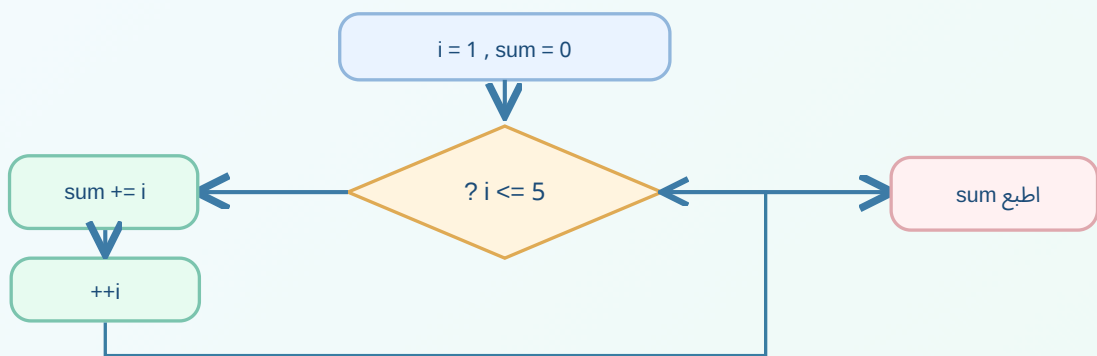
06 · تثبيت الفهم

ماذا يجب أن يبقى في ذاكرتك؟

لا تقرأ القواعد فقط؛ اربط كل قاعدة بخطأ تمنعه أو مشكلة تحلها.

- ✓ هين المجمع إلى 0 أو قيمة مناسبة للوظيفة.
- ✓ أسباب عدم الانتهاء غالبًا: تحديث مفقود أو شرط خاطئ.
- ✓ في التكرار المتداخل، الأعرق يكتمل لكل خطوة خارجية.

الرسم التوضيحي: كيف تعمل الفكرة؟



تحضير الكويز التفاعلي

قبل فتح اختبار هذه المحاضرة، حاول شرح المثال الأول لشخص آخر دون النظر إلى الشيفرة. إذا فهم لماذا تعمل الخطوات، فأنت جاهز.

الجسر إلى المحاضرة التالية

بعد أن تعلمنا الحلقات سنخزن مجموعات بيانات من النوع نفسه في المصفوفات.

المرجع: «برمجة 1» لم. براءة سراقبي، الصفحات 90–108 من الملف المرفق. الأمثلة والشروح الإضافية صيغت للأغراض التعليمية وتستخدم C++17.