مختصر كتابي

Problem Inputs Outputs and Constraints



ماذا يجب أن تعرف عن المشكلة قبل أن تكتب سطرًا واحدًا من الكود؟

تخيّل أن طلبًا وصلك في العمل: «لدينا نظام لحجز قاعة اجتماعات، ونريد أن نعرف إن كان هناك تعارض في الحجوزات». قد تبدأ فورًا في تخيّل الـLoops والمقارنات، لكن قبل الكود توجد تفاصيل يجب أن تفهمها: ما البيانات التي ستصلك؟ وما معنى التعارض؟ وهل المطلوب معرفة وجود التعارض فقط أم المواعيد المتعارضة نفسها؟ إجابات هذه الأسئلة هي التي تحدّد شكل الحل.

أولًا، **المدخلات** (Inputs): قائمة حجوزات، لكل حجز وقت بداية (Start Time) ووقت نهاية (End Time). ولتبسيط التمثيل نعبّر عن الوقت بعدد الدقائق من بداية اليوم؛ فالساعة 9 تساوي 540، والساعة 10 تساوي 600. ونتفق على أن كل الحجوزات للقاعة نفسها وفي اليوم نفسه، ولا يمتد أي حجز إلى اليوم التالي. هذا التفصيل مهم؛ فالموعد نفسه في قاعتين مختلفتين لا يُعدّ تعارضًا.

ثانيًا، **المخرجات** (Outputs): النتيجة Boolean؛ `true` إذا تعارض حجزان على الأقل، و `false` إذا لم يوجد أي تعارض. أما لو كان المطلوب عدد التعارضات أو قائمة بالحجوزات المتعارضة، فهذه مسألة مختلفة.

ثم نحدّد معنى التعارض نفسه. الحجز من 9 إلى 10 والحجز من 9:30 إلى 10:30 يحتاجان القاعة معًا من 9:30 إلى 10، فهناك تعارض. لكن ماذا عن حجز ينتهي في 10 وآخر يبدأ في 10؟ هنا تعتمد الإجابة على **قاعدة العمل** (Business Rule). في مسألتنا نسمح بذلك، ولا يوجد وقت تجهيز بين الحجوزات؛ فتلامس النهاية مع البداية ليس تعارضًا. ولو احتاج النظام الحقيقي 10 دقائق للتجهيز، فيجب أن تدخل هذه المعلومة في وصف المشكلة، لا أن نفترضها من عندنا.

ثالثًا، **القيود** (Constraints): وقت البداية أصغر من وقت النهاية، والأوقات Integers داخل اليوم نفسه، والقائمة قد تكون فارغة أو فيها حجز واحد أو حجوزات كثيرة، وقد تأتي بأي ترتيب؛ فلا ضمان أنها مرتبة حسب وقت البداية. لذلك فأي حل يعتمد على ترتيب معيّن يجب أن يتأكد منه أو يرتّب القائمة بنفسه. ويهمّنا أيضًا **حجم المدخلات** (Input Size)؛ فحلّ يناسب 10 حجوزات قد يكون مكلفًا جدًا مع 100000 حجز. وفي تمريننا نسمح بتغيير ترتيب القائمة الأصلية (Input Mutation) إن احتجنا إليه.

- القائمة الفارغة أو ذات الحجز الواحد: النتيجة `false`.
- حجزان بالموعد نفسه تمامًا: تعارض، والنتيجة `true`.
- حجز من 9 إلى 12 بداخله حجز من 10 إلى 11: تعارض أيضًا.
- الحجوزات 11–12 و9–10 و10–11 غير مرتبة لكنها متتالية بلا تداخل: النتيجة `false`.
- إذا وجدت معلومة مؤثرة ناقصة في العمل فاسأل صاحب الطلب، وفي التمارين راجع الوصف والقيود والأمثلة قبل أن تفترض قاعدة.

بعد هذا التحليل يمكنك أن تلخّص المطلوب بدقة: قائمة حجوزات لقاعة واحدة في يوم واحد، بداية كل حجز أصغر من نهايته، والقائمة ليست بالضرورة مرتبة، والمطلوب معرفة هل يشترك حجزان في فترة زمنية، مع

السماح لحجز أن يبدأ لحظة انتهاء آخر. جرّب الآن أن تكتب 3 حالات اختبار (Test Cases) مع النتيجة المتوقعة وسببها: واحدة فيها تعارض، وواحدة بلا تعارض، وواحدة تختبر حالة حدّية (Edge Case).

الخلاصة: حدّد المدخلات والمخرجات ومعنى المصطلحات والقيود والحالات الحدية قبل الكود؛ فوصف المشكلة الدقيق هو ما يتيح لك معرفة النتيجة المتوقعة قبل أن تكتب الحل.