

الدرس 1

الوحدة 1

المُكدّس والطابور Stack & Queue



وصف الدرس

الغرض العام من هذا الدرس هو أن يصنف الطلبة هياكل البيانات إلى نوعيها وهما هياكل بيانات بسيطة وغير بسيطة. سيتعرفون أيضًا على بعض الهياكل الأكثر استخدامًا في حياتنا اليومية مثل المكّدسات والطوابير.

ما سيتعلمه الطالب

- < التعرف على المُكدّس Stack في البرمجة.
- < شرح استخدامات المُكدّس في البرمجة.
- < التعرف على أهم العمليات التي يمكن إجراؤها على المُكدّس.
- < كتابة برامج لحل بعض المسائل باستخدام المُكدّس.
- < التعرف على الطابور Queue.
- < التعرف على أهم العمليات التي يمكن إجراؤها على الطابور.
- < كتابة برامج لحل بعض المسائل البرمجية باستخدام الطابور.

نتائج التعلم

- < استخدام هياكل البيانات المتقدمة.
- < المقارنة ما بين هياكل البيانات واستخداماتها.
- < تمثيل هياكل البيانات المتقدمة باستخدام البرمجة بلغة Python.

اللغة الإنجليزية	اللغة العربية
Data Structure	هيكل البيانات
Primitive	بسيطة
Non Primitive	غير بسيطة
Stack	مكدس
Queue	طابور
Push	إضافة
Pop	إزالة
Overflow	تجاوز الحد الأعلى
Underflow	تجاوز الحد الأدنى
Pointer	مؤشر
Enqueue	إضافة للطابور
Dequeue	إزالة من الطابور
Top	مؤشر للمكدس
Front	أمامي
Rear	خلفي



التحديات المتوقعة



< قد يواجه الطلبة صعوبة في تمييز أوجه الاختلاف بين هياكل البيانات البسيطة وغير البسيطة. استعن بالمخطط الموجود في كتاب الطالب لمساعدتهم على تصنيفها بصورة صحيحة.

< قد يواجه الطلبة صعوبة في فهم وظيفة المؤشر (pointer). اشرح لهم بأن المؤشر هو عبارة عن متغير يشير إلى عنوان عنصر في المكدس أو في الطابور.



التمهيد

< قدّم الغرض من هذا الدرس من خلال تحفيز الطلبة للتعرف على هياكل البيانات. اعرض لهم تصنيف هياكل البيانات بالاستعانة بالصورة

< **G12a_CS_U1_data_structures.jpg**، وابدأ النقاش بالأسئلة التالية:

- ما هي هياكل البيانات التي يمكنكم تمييزها في هذا المخطط؟

- كيف يمكن استخدام هياكل البيانات السابقة؟

< يمكنك باستخدام الأسئلة الشفوية التعرف على معارف الطلبة السابقة ومساعدتهم في ربط المفاهيم الجديدة مع معرفتهم السابقة.



التلميحات الخاصة بالتنفيذ

< ابدأ بتذكير الطلبة بهياكل البيانات، واستعن بالمخطط الخاص بتصنيف هياكل البيانات (الصورة "G12a_CS_U1_data_structures.jpg").

< استخدم الإرشادات المباشرة لشرح كيفية تصنيف هياكل البيانات، وشرح هياكل البيانات البسيطة وغير البسيطة وأوجه الاختلاف بينها.

< ضع مجموعة من الكتب بشكل مكّدد على المكتب، ثم اطلب من الطلبة وصف الخطوات التي سيقومون بها للحصول على أحد الكتب في أسفل المجموعة.

< قم بشرح هياكل بيانات المكدس، مع الإشارة إلى وجه الشبه بين مفهوم التكدس في

الحياة الواقعية (كتكديس الكتب مثلاً)، والمُكدّس في علم الحاسوب، والذي يحتوي على بيانات.

< اشرح لهم بأن المُكدّس يعتمد على مبدأ أن آخر عنصر تتم إضافته في المُكدّس هو أول عنصر يتم إزالته، ويطلق على هذه العملية اختصاراً (Last In First Out-LIFO).

< استخدم الإرشادات المباشرة في شرح وظيفة أمري الإضافة والحذف من المُكدّس (push) و (pop).

< استخدم مجموعة الكتب كمثال على مكّس لتوضيح الأمرين السابقين.

< أكد على دور المؤشر **top** بصفته يشير إلى آخر عنصر تمت إضافته إلى القائمة.

< اعرض للطلبة الدوال المستخدمة في **Python** لتنفيذ هيكل المُكدّس كدالة **append()** المستخدمة لإدخال العناصر، ودالة **pop()** لإزالة العناصر.

< قم بالتأكيد على ضرورة التحقق من عدم امتلاء المُكدّس قبل إضافة عنصر، وذلك لتجنب تجاوز السعة المحددة لذلك المُكدّس. يجب التحقق من أن المُكدّس يحتوي على عنصر واحد على الأقل قبل إزالة عنصر من المُكدّس، وإلا فسيتم التسبب في خطأ يطلق عليه خطأ تجاوز الحد الأدنى (**underflow**).

< استعن بأمثلة الكتاب واستراتيجية التعلم القائمة على المشروع لإنشاء برامج تستخدم المُكدّس مع تقديم التوجيه اللازم عند الحاجة.

< تابع الشرح بطرح مثال إجراء التراجع في برنامج مايكروسوفت وورد. يمكن للطلبة من خلال هذا المثال الاطلاع على الاستخدام العملي للمكدس، ويعتبر هذا مثالاً يجمع بين المعرفة النظرية والتطبيق العملي للمفهوم.

< قم بوضع بعض الأشياء التي يحتويها المكتب على شكل صف، ثم اطلب من الطلبة وصف ما سيقومون به للحصول على أحد تلك الأشياء الموجود في منتصف الصف. وضّح للطلبة أنهم سيتعرفون على أحد هياكل البيانات شائع الاستخدام في علم الحاسوب والذي يشبه الصف بخواصه ويسمى الطابور (**queue**).

< ا طرح مثال طابور انتظار السيارات على إشارة المرور عند شرح الخاصية -أول من يدخل أول من يخرج (**FIFO**) - الخاصة بالطابور، فأول سيارة تصل إلى إشارة المرور هي أول سيارة ستغادر الطابور عندما تتحول إشارة المرور إلى اللون الأخضر.

< استخدم الإرشادات المباشرة في شرح دوال الطابور **enqueue** و **dequeue**.

< استخدم صف الأشياء السابق على المكتب وتقنية العرض لتمثيل هاتين الدالتين.

< قم بالتأكيد على الوظيفة التي يقوم بها مؤشر الطابور، فمؤشر **front** يشير إلى العنصر الأول الذي ستم إزالته، ومؤشر **rear** يشير إلى آخر عنصر في قائمة الطابور.

< اشرح للطلبة أنه كما في المُكدّس فإننا في **Python** نستخدم نفس الدوال لتنفيذ هيكل قائمة الطابور، حيث أن دالة **append()** تستخدم لإدراج عنصر في الطابور، أما دالة (**pop**) فتستخدم لإزالة عنصر منه.

- < اشرح للطلبة أنه يمكن تمثيل المكّس والطابور في لغة البايثون من خلال القوائم أو من خلال مكتبة Queue.
- < في النهاية، استخدم الإرشادات المباشرة لتعريف الطلبة بمكتبة Queue الموجودة في Python لتمثيل هيكل الطابور أو المكّس.
- < تتضمن مكتبة Queue بعض الدوال الجاهزة للاستخدام التي يمكن استخدامها مع المكّس أو الطابور.
- < قم بتوضيح الاختلافات بين الدوال المختلفة في مكتبة Queue واستخدامها مع هيكل بيانات المكّس.
- < استعن بأمثلة الكتاب واستراتيجية التعلم القائمة على المشروع لإنشاء برامج تستخدم الطابور مع تقديم التوجيه للطلبة عند الحاجة.
- < تحقق أثناء الدرس من فهم كل طالب للخطوات واتباعها بشكل صحيح وقدم إرشادات فردية لمن يحتاج إلى المساعدة، وتحقق أيضًا من أي جزء من الدرس لم يفهمه الطلبة تمامًا وقم بإجراء أي تغييرات مطلوبة في عملية التدريس لكي يفهموه.
- < يمكنك الاستعانة بالطلبة المتميزين لتقديم التغذية الراجعة لبقية زملائهم أثناء حل التطبيقات العملية، بعد أن يقوم هؤلاء الطلبة بحلها بشكل فردي.



استراتيجيات غلق الدرس

في نهاية الدرس تأكد من فهم الطلبة لجميع أهداف الدرس وقم بتقييم معرفتهم من خلال أسئلة على سبيل المثال لا الحصر:

< هل تستطيع أن تتذكر:

- هياكل البيانات الأساسية وغير الأساسية.
- ما الدوال الرئيسية الخاصة بهيكل بيانات المكّس؟
- ما هي وظيفة الدالة **push** والدالة **pop** في المكّس؟
- ما هي كيفية إنشاء برنامج يستخدم المكّس في Python؟
- اذكر أحد الأمثلة على المكّس في الحياة الواقعية؟
- ما الدوال الرئيسية الخاصة بهيكل بيانات الطابور؟
- ما هي وظيفة الدالة **enqueue** والدالة **dequeue** في الطابور؟
- ما هي كيفية إنشاء برنامج يستخدم الطابور في Python؟

• ما المؤشرات المستخدمة في هيكل بيانات المُكدّس والطابور؟

• ما هو الغرض من استخدام المؤشرات في هيكل بيانات المُكدّس والطابور؟

• ما هو الفرق بين اختصارات LIFO و FIFO؟

< ذكّر الطلبة بالمصطلحات الهامة وكررها معهم.

< يمكنك الاستعانة بتدريبات الكتاب ضمن الاستراتيجيات التي ستستخدمها لغلق الدرس.

التدريبات المقترحة لغلق الدرس



يمكنك استخدام التمرين الرابع والتمرين الخامس في هذا الدرس كاستراتيجية ختامية لتقييم وتعزيز قدرة الطلبة على فهم المصطلحات الرئيسة التي وردت في هذا الدرس.

الصف الثاني عشر | الفصل الأول | كتاب الطالب | صفحة 42

الفروق الفردية

تمارين إضافية للطلبة ذوي التحصيل المرتفع

< بعد الانتهاء من التمرين السابع لهذا الدرس، اطلب من الطلبة كتابة برنامج يقوم بإنشاء الطابور الذي شرحه سابقاً، باستخدام مكتبة **Queue** القياسية.

1



اختر الإجابة الصحيحة مما يلي.

<input checked="" type="radio"/>	LIFO rule	1. يتبع المُكدّس قاعدة
<input type="radio"/>	FIFO rule	
<input type="radio"/>	FIKO rule	
<input type="radio"/>	LIFO rule	2. يتبع الطابور قاعدة
<input checked="" type="radio"/>	FIFO rule	
<input type="radio"/>	FIKO rule	
<input type="radio"/>	عملية push	3. لإضافة عنصر جديد في الطابور نستخدم
<input type="radio"/>	عملية dequeue	
<input checked="" type="radio"/>	عملية enqueue	
<input type="radio"/>	عملية push	4. لإزالة عنصر من المُكدّس نستخدم
<input checked="" type="radio"/>	عملية pop	
<input type="radio"/>	عملية enqueue	
<input type="radio"/>	أول عنصر في الطابور	5. مؤشر Rear يشير إلى
<input checked="" type="radio"/>	آخر عنصر في الطابور	
<input type="radio"/>	العنصر الأوسط في الطابور	
<input type="radio"/>	أول عنصر في الطابور	6. مؤشر Top يشير إلى
<input type="radio"/>	آخر عنصر في الطابور	
<input checked="" type="radio"/>	آخر عنصر تمت إضافته في المُكدّس	



2

اكتب مثالاً على استخدام المُكَدَّس في حياتنا اليومية.

تلميح:

يمكنك إيجاد المثال صفحة 12
من كتاب الطالب.



3

اكتب مثالاً على استخدام الطابور في حياتنا اليومية.

تلميح:

يمكنك إيجاد المثال صفحة 24
من كتاب الطالب.

4



أكمل تعبئة المُكدّس الذي تم إنشاؤه بالمقطع البرمجي التالي موضحًا موضع المؤشر Top.

D	3	← top
o	2	
H	1	
a	0	

```
import queue

myStack = queue.LifoQueue()

myStack.put("a")
myStack.put("H")
myStack.put("o")
myStack.put("D")
```

5

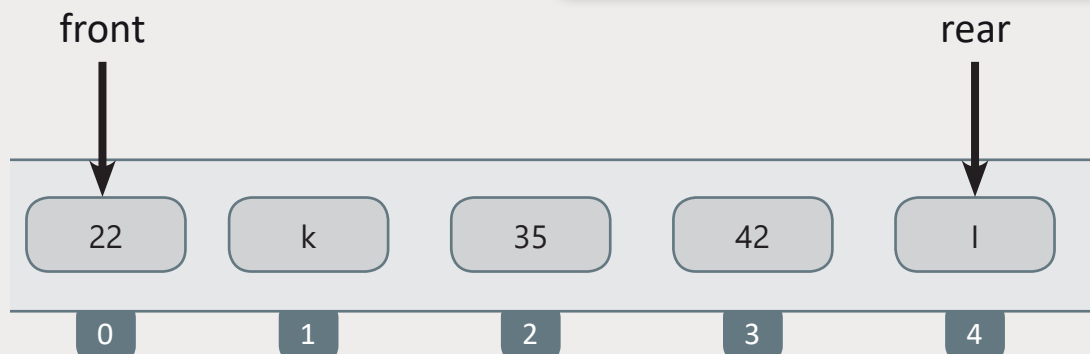


أكمل تعبئة الطابور الذي تم إنشاؤه بالمقطع البرمجي التالي موضحًا موضع مؤشري Front و Rear.

```
import queue

q = queue.Queue()

q.put(22)
q.put("k")
q.put(35)
q.put(42)
q.put("l")
```





6

لدينا مكّس بستة أماكن فارغة.

< سنضيف الحروف التالية C و E و B و A و D في المواضع من 0 إلى 4.

< قم بتعبئة المكّس مبيّنًا موضع المؤشر Top.

< في ورقة خارجية، نفّذ العمليات التالية:

	5	
D	4	← top
A	3	
B	2	
E	1	
C	0	

pop <

pop <

pop <

push X <

push K <

pop <

< وضح الناتج النهائي بعد تنفيذ العمليات أعلاه مبيّنًا موضع المؤشر top.

	5	
	4	
	3	
X	2	← top
E	1	
C	0	

< اكتب برنامجًا ينشئ المكّس الموضح أعلاه، ثم نفّذ العمليات المذكورة باستخدام مكتبة Queue القياسية.

تلميح:

يمكنك إيجاد إجابة التمرين المقترحة في مستند الأنشطة الذي يحمل الاسم

G11b_CS_U1_L1_ex6.py

7



لديك تسلسل الأعداد التالي: 4، 8، 2، 5، 9، 13.

< ما هي العملية المستخدمة لإضافة العناصر أعلاه إلى الطابور.

enqueue

< أكمل الطابور بعد إضافة العناصر.

0	1	2	3	4	5
4	8	2	5	9	13

< ما هي العملية المستخدمة لإزالة العناصر من الطابور؟

dequeue

< كم مرة يجب تنفيذ العملية السابقة لإزالة العنصر الذي يحمل القيمة 5؟

4 مرات.

8



أنشئ الطابور السابق باستخدام البرمجة بلغة Python.

< قم بإزالة جميع العناصر من الطابور.

< تحقق مما إذا كان الطابور فارغاً في النهاية.

تلميح:

يمكنك إيجاد إجابة التمرين المقترحة في مستند
الأنشطة الذي يحمل الاسم

G12a_CS_U1_L1_ex8.py



ترغب إدارة أحد البنوك في تنفيذ برنامج لإدارة خدمة العملاء. يستقبل البرنامج من موظف البنك أرقام العملاء الموجودين في الانتظار ويخزنهم في طابور مكون من 10 خانات، يقوم الموظف نفسه بعرض رقم العميل على الشاشة عندما يأتي دوره وإنهاء البرنامج عند الانتهاء من خدمة جميع العملاء.

قم بإعداد برنامج يقوم بـ :

< إدخال أرقام العملاء من المستخدم.

< إذا تم إعطاء كلمة Next فسيخرج البرنامج رقم العميل التالي ويعرضه على الشاشة، إذا لم يكن هناك عملاء في الانتظار فسيعرض الرسالة "No one waits".

< إذا تم إعطاء الكلمة END فلن يسمح البرنامج بإدخال أرقام أخرى باستثناء قيمة Next.

< في أي حالة أخرى يتم إدخال رقم العميل في الطابور إلا إذا كان الطابور ممتلئ، يتم عرض الرسالة "No more space".

< يتوقف البرنامج عندما يتم إعطاء الكلمة END ويكون الطابور فارغ.

تلميح:

يمكنك إيجاد إجابة التمرين المقترحة في مستند الأنشطة الذي يحمل الاسم
G12a_CS_U1_L1_ex9.py