

نکات:

- پروژه را می‌توانید به صورت فردی یا در گروه حداکثر چهار نفره انجام دهید.
- پاسخ پروژه توسط ایمیل فقط به آدرس educompsci در جیمیل ارسال شود.
- اسامی گروه باید به فارسی در متن ایمیل نوشته شود.
- پاسخ پروژه می‌تواند بیشتر از یک فایل داشته باشد.
- برنامه‌های نوشته شده در پایتون ۲ مورد قبول نیست. برنامه‌ها در پایتون ۳.۴.۴ یا ۳.۵.۲ تست خواهند شد.
- فقط استفاده از ماژول turtle و os مجاز است.
- استفاده از مطالب مطرح نشده در کلاس مجاز نیست.
- در صورت مشابه بودن پروژه گروه‌ها، تمام طرفین نمره **منفی** خواهند گرفت.
- جریمه تأخیر برای هر ساعت یک درصد نمره است.
- نکات خوانانویسی نمره دارد: کامنت‌گذاری، docstring، فاصله‌گذاری افقی و عمودی، نام‌گذاری، محل import و توابع. افراط در کامنت‌گذاری، فاصله‌گذاری، و نام‌گذاری باعث کاهش نمره می‌شود.
- برنامه تحت هیچ شرایطی نباید دچار خطا شود.
- داشتن امکانات اضافی در برنامه، نمره اضافی نخواهد داشت.

می‌خواهیم برنامه‌ای بنویسیم که با استفاده از آن کاربر بتواند بازی ۱۲۳ را انجام دهد. این بازی در یک جدول ۵ در ۵ انجام می‌شود. در هر سطر و ستون این جدول دو خانه خالی وجود دارد. در هر یک از سه خانه دیگر باید یکی از اعداد ۱، ۲، و ۳ قرار گیرد طوری که در هیچ سطر یا ستونی عدد تکراری وجود نداشته باشد. در ابتدا و انتهای بعضی از سطرها و ستون‌ها یک راهنما وجود دارد. این راهنما نشان می‌دهد که در آن سطر یا ستون عدد نزدیک به آن سمت چه عددی است. همچنین ممکن است بعضی از اعداد جدول معلوم باشند. این بازی طوری طراحی می‌شود که فقط یک جواب داشته باشد. شکل ۱ مثالی از صفحه آغازین این بازی را نمایش می‌دهد.

	2		1	2	3	
	a	b	c	d	e	
	f	g	h	i	j	1
3						
	k	l	m	n	o	
1				3		
	p	q	r	s	t	
2						
	u	v	w	x	y	
3						2
			2	1		

شکل ۱

در بازی فوق مثلاً عدد ۲ بالای خانه a نشان می‌دهد که در ستون اول عدد ۲ بالاتر از اعداد دیگر قرار دارد، عدد ۱ در سمت راست خانه j نشان می‌دهد که در سطر دوم عدد ۱ در سمت راست اعداد دیگر قرار دارد. در خانه n نیز عدد معلوم ۳ قرار دارد. شکل ۲ برای این مثال، بازی حل شده را نمایش می‌دهد.

	2		1	2	3	
	a	b	c	d	e	
	2		1		3	
3	f	g	h	i	j	1
		3		2	1	
1	k	l	m	n	o	
	1			3	2	
2	p	q	r	s	t	
		2	3	1		
3	u	v	w	x	y	2
	3	1	2			
			2	1		

شکل ۲

اطلاعات بازی در یک فایل متنی به این صورت ذخیره می‌شود:

- فایل حاوی ۷ سطر است.
- در هر سطر فایل ۷ عدد وجود دارد که با فاصله از هم جدا شده‌اند.
- اولین و هفتمین عدد سطر اول و آخر برابر صفر هستند.
- اعداد دوم تا ششم سطر اول راهنماهای بالای ستون‌ها هستند.
- اعداد دوم تا ششم سطر آخر راهنماهای پایین ستون‌ها هستند.
- اولین اعداد سطرهای دوم تا ششم راهنماهای سمت چپ سطرها هستند.
- آخرین اعداد سطرهای دوم تا ششم راهنماهای سمت راست سطرها هستند.
- اگر راهنمایی وجود نداشته باشد، در مکان مربوطه عدد صفر گذاشته می‌شود.
- اعداد دوم تا ششم هر سطر، به ترتیب اعداد سطرهای اول تا پنجم خانه‌های جدول هستند.
- اگر یک خانه در جدول خالی باشد به جای آن در مکان مربوطه صفر گذاشته می‌شود.
- اگر عدد یک خانه جدول معلوم باشد، در مکان مربوطه عدد منفی آن گذاشته می‌شود.

فایل‌های نمونه پیوست شده است.



برنامه‌ای برای انجام این بازی با شرایط زیر بنویسید:

- برای دریافت دستورات از کاربر از **input** استفاده می‌شود.
- برنامه در ابتدا نام یک فایل را از کاربر می‌گیرد. سپس صفحه بازی را مانند شکل ۱ نمایش می‌دهد.
- اگر فایل وجود نداشته باشد باید پیغام مناسب داده شود و دوباره درخواست ورودی شود.
- فرض کنید محتویات فایل درست است.
- بعد از خواندن فایل کاربر باید بتواند بازی را انجام دهد.
- هنگام بازی کاربر نام خانه و یکی از اعداد ۰ تا ۳ را وارد می‌کند. مثلاً **b1** که به معنی درج عدد ۱ در خانه **b** است. اگر خانه **b** در حال حاضر خالی باشد باید عدد ۱ در آن درج شود. اگر عدد دیگری در آن موجود باشد، باید آن عدد پاک شده و عدد ۱ درج شود. عدد ۰ به معنی خالی کردن است، مثلاً وقتی کاربر **b0** وارد می‌کند باید عدد خانه **b** پاک شود و اگر خانه **b** خالی باشد، کاری انجام نمی‌شود.
- کاربر نباید بتواند عدد خانه‌های معلوم بازی را تغییر دهد یا این خانه‌ها را خالی کند. در صورت وارد کردن نام این خانه‌ها باید پیغام مناسب داده شود.
- هر گاه کاربر رشته **quit** را وارد کند برنامه خاتمه پیدا می‌کند.
- هر گاه کاربر رشته **new** را وارد کند، برنامه به مرحله شروع (دریافت نام فایل) می‌رود.
- با گرفتن نام یک فایل جدید باید صفحه بازی قبلی پاک شده و صفحه مربوط به بازی جدید نمایش داده شود.
- هر گاه کاربر رشته **check** را وارد کند برنامه بررسی می‌کند که آیا بازی درست حل شده است یا خیر. در صورت درستی باید پیغام **solved** و در غیر این صورت **not solved** می‌دهد. در صورت نادرست بودن، بازی ادامه پیدا می‌کند. در صورت درست بودن، برنامه از کاربر می‌پرسد که آیا می‌خواهد بازی جدیدی را شروع کند یا خیر، در صورت مثبت بودن پاسخ، برنامه به مرحله شروع (دریافت نام فایل) می‌رود. در صورت منفی بودن پاسخ، برنامه خاتمه پیدا می‌کند.
- هر گاه کاربر رشته **show** را وارد کند، پاسخ بازی نمایش داده می‌شود. سپس برنامه از کاربر می‌پرسد که آیا می‌خواهد بازی جدیدی را شروع کند یا خیر، در صورت مثبت بودن پاسخ، برنامه به مرحله شروع (دریافت نام فایل) می‌رود. در صورت منفی بودن پاسخ، برنامه خاتمه پیدا می‌کند.
- هرگاه کاربر ورودی نامعتبری وارد کند، برنامه باید پیغام **invalid input** بدهد و دوباره درخواست ورودی کند.