



پروژه نهایی درس ریزپردازنده و زبان اسمبلی

بازی Plants vs Zombies

بهار 98



## 1 موضوع

در این پروژه، طراحی و پیاده‌سازی بازی Plants vs Zombies مطلوب است.

## 2 زمان بازگردانی

زمان تحویل مجازی پروژه 10 تیر ماه می‌باشد. زمان تحویل حضوری آن متعاقبا اعلام خواهد شد.

## 3 سناریو

این بازی از 8 مرحله تشکیل شده و شامل یک نشانگر (Cursor) و یک صفحه بازی  $20 \times 4$  می‌باشد. کاربر می‌تواند نشانگر را به هر یک از خانه‌های صفحه حرکت دهد و در آن خانه مانعی قرار دهد. نشانگر بر روی هر خانه که قرار می‌گیرد باید به صورت چشمک زن باشد به صورتی که کاراکتر فعلی آن خانه نیز قابل مشاهده باشد.

شما در ابتدا 5 شانس بازی دارید که بر روی 7-segment قابل مشاهده است. در ابتدا و پیش از شروع بازی، بازیکن می‌تواند 7 مانع بر روی صفحه در نقاط دلخواه بگذارد و آماده شروع بازی شود. پس از شروع بازی، دشمنان از بالای صفحه (LCD) را به صورت افقی در نظر بگیرید) به صورت رندم وارد صفحه می‌شوند. نشانگر در طول زمان بازی، به کمک Volume به سمت چپ و راست حرکت می‌کند. برای بالا و پایین بردن نشانگر دو عدد دکمه به صورت دلخواه از رو کی پد مورد استفاده قرار می‌گیرد. زمان انتقال دشمنان به خانه پایینی (سرعت دشمنان) در ابتدا دو ثانیه می‌باشد و با گذشت هر مرحله 10 درصد سریعتر می‌شود. حرکت هر دشمن مستقل از دشمن دیگر است **بنابراین بازی نباید به نحوی باشد که همواره تمام دشمنان با هم حرکت کنند.** هر مرحله بازی 20 ثانیه طول کشیده و پس از گذشت این مدت بازی وارد مرحله بعدی می‌شود. شماره مرحله مورد نظر از طریق LED های روی بورد باید مشخص شود. با گذشت هر مرحله به تعداد دشمنانی که از ابتدا تا انتهای مرحله وارد صفحه می‌شوند دو نفر افزوده می‌شود. توجه داشته باشید که در هر خانه از بازی فقط یک دشمن در یک زمان باید قرار بگیرد. تعداد دشمنان در مرحله اول 8 عدد بوده که بیشتر از 5 دشمن به صورت همزمان در صفحه نباید باشند.

مدت زمان بازی باید بر روی سه رقم سمت راست یک نمایشگر 7-segment که به وسیله IC مربوطه راه‌اندازی شده نشان داده شود. همچنین در سمت چپ‌ترین رقم تعداد شانس های کاربر نمایش داده می‌شود که با "." از سه رقم سمت راست جدا شده است.



شکل 1

در این بازی 4 نوع دشمن و 3 نوع مانع وجود دارد. یک نمونه کاراکتر زامبی را می‌توانید در پروژه custom character (از قسمت پروژه‌های آماده موجود در کتابخانه LiquidCrystal) مشاهده کنید. هر کدام از موانعی که کاربر می‌تواند استفاده کند قدرت خاصی داشته و استفاده مجدد از آن بعد از یک زمان مشخص خواهد بود. برای ساختن موانع و دشمنان می‌توانید از special character استفاده کنید. مانع نوع اول هر 4 ثانیه امکان استفاده مجدد دارد، مانع دوم هر 8 ثانیه و مانع نوع سوم هر 10 ثانیه. یک LED خارجی روی برد امکان استفاده مجدد از آن مانع را مشخص می‌کند. یعنی هر زمان که مانع مجدداً قابل استفاده باشد LED مربوط به آن مانع روشن می‌شود. هنگام برخورد یک دشمن با مانع با توجه به انرژی باقیمانده برای مانع، **مانع و دشمن شروع به چشمک زدن می‌کنند**. اگر انرژی مانع تمام شده باشد مانع از بین رفته و دشمن به مسیر خود ادامه می‌دهد در غیر این صورت دشمن از بین می‌رود. انرژی مانع نوع اول برابر 1، مانع نوع دوم برابر 2 و مانع نوع سوم برابر 4 می‌باشد. قدرت دشمن نوع اول برابر 1، دشمن نوع دوم برابر 2، دشمن نوع سوم برابر 3 و دشمن نوع چهارم برابر 4 می‌باشد. نحوه ورود انواع دشمنان (نوع دشمنان و زمان‌بندی ورود آن‌ها) به صفحه بازی باید متناسب با سختی مرحله بازی باشد.

**نکته:** دشمنان باید به صورت کاملاً تصادفی و نه به صورت Hard Code شده وارد صفحه بازی شوند. بنابراین با هر بار اجرای بازی ورود دشمنان متفاوت خواهد بود.

در طول مسابقه در زمان‌هایی تصادفی باید یک کاراکتر خاص به عنوان جایزه (Bonus) بر روی قسمتی از صفحه بازی پدیدار شود و پس از گذشت 4 ثانیه ناپدید شود. جایزه می‌تواند استفاده از یک مانع زودتر از موعد یا از بین بردن دشمن یا اضافه شدن امتیاز باشد که به صورت کاملاً تصادفی می‌باشد. اگر جایزه امتیاز باشد 100 امتیاز به کاربر اضافه می‌شود. اگر جایزه از بین بردن دشمن باشد دشمنی که از همه نزدیکتر به پایان صفحه است حذف می‌شود و در صورتی که استفاده زود هنگام از مانع باشد، LED مربوط به قوی‌ترین مانع خاموش، روشن شده و آن مانع قابل استفاده خواهد شد. برای برداشتن جایزه باید از طریق نشانگر بر روی خانه جایزه رفته و دکمه آبی روی برد را فشار دهید.

در صورت رسیدن هر دشمن به پایین صفحه یکی از شانس‌های بازی شما از بین می‌رود.

نور تمامی LED های خارجی و روی میکرو باید مبتنی بر میزان نور محیط بیرونی که از طریق سنسور LDR دریافت می‌شود، **تعدیل** شوند؛ این امر از طریق PWM ممکن خواهد بود.

در صورتی که کاربر برنده بازی شود، یعنی 8 مرحله را به پایان برساند عبارت You Won به همراه امتیاز کاربر روی صفحه به صورت چشمک زن نمایش داده می شود. سایر ماژول ها نیز همانند آغاز بازی شروع به چشمک زدن می کنند. در این زمان با زدن دکمه آبی روی برد بازی مجدداً با سبک آخرین مرحله پیگیری می شود. در صورت اتمام شانس های بازیکن عبارت Game Over به صورت چشمک زن روی صفحه به نمایش در می آید.

امتیاز کاربر از طریق فرمول زیر محاسبه می شود:

$$\text{Score} = (\text{مدت زمان سپری شده از آخرین شانس از دست رفته} + \text{مدت زمان سپری شده از بازی}) * \text{شماره مرحله بازی}$$

قبل از شروع بازی سون سگمنت و LED های خارجی و LED های روی میکرو و همچنین نام بازی بر روی ال سی دی باید به صورت چشمک زن باشد.

### منوی بازی

شما باید یک منو برای بازی خود شامل گزینه های New Game، Load Game (از UART) و About طراحی کنید.

### ورود نام برای ذخیره بازی با استفاده از کی پد

بعد از دریافت فرمان از یکی از کلیدهای دلخواه کی پد برنامه شما باید صفحه ای را جهت وارد کردن نام برای ذخیره سازی بازی نمایش دهد. در این صفحه با استفاده از کی پد یک عدد (چند رقمی) به عنوان اسم وارد شده (و همزمان در صفحه نمایش نشان داده می شود) و پس از فشردن کلید پایین سمت چپ در کی پد اطلاعات مربوط به بازی شامل موارد گفته شده در قسمت بعد از طریق UART به کامپیوتر ارسال می شود. ترتیب کلیدها در کی پد باید مانند شکل زیر باشد.



شکل 2

## ذخیره سازی بازی

هرگاه بازی وارد مود ذخیره سازی شد عبارت Enter your name بر روی LCD ظاهر می شود. آن گاه کاربر از طریق کی پد یک id وارد می کند و دکمه تایید را فشار می دهد. سپس موارد زیر از طریق UART به کامپیوتر فرستاده شده و بر روی ترمینال نمایش داده می شود:

- تعداد، موقعیت و انرژی باقی مانده برای هر کدام از دشمنان و موانع باقیمانده بر روی صفحه.
- تعداد شانس های باقیمانده کاربر.
- زمان سپری شده تا به اینجای بازی.
- مرحله کنونی بازی.
- تایم استفاده مجدد از هر مانع.
- امتیاز.

سپس بازی ادامه می یابد. نحوه ذخیره سازی و فرمت ارسال اطلاعات شما باید به نحوی باشد که در قسمت بارگذاری قابل استفاده باشد.

## بارگذاری بازی

هنگامی که از منو گزینه Load Game انتخاب شد آنگاه اطلاعاتی که از ذخیره بازی بر روی ترمینال وجود داشت را به برد ارسال می کنیم. بدین ترتیب که اطلاعات یکی از دفعات ذخیره سازی را از روی ترمینال کپی کرده و همانجا paste می کنیم و Enter میزنیم. این اطلاعات از طریق UART به میکروکنترلر فرستاده شده و بازی از همان حالتی که ذخیره شده بود ادامه می یابد.

## نکات قابل توجه:

- ❖ سعی کنید تمامی اعداد گفته شده در بازی (مانند انرژی دشمنان و زمان استفاده مجدد موانع) را به صورت پارامتری پیاده‌سازی کنید تا به راحتی امکان تغییر در آن را در صورت تغییر صورت بازی داشته باشید.
- ❖ تمامی قسمت های بازی باید بصورت وقفه ای پیاده سازی شده باشد.
- ❖ به ازای هر خط دستور داخل حلقه (1) while 50 بیت منفی تعلق می‌گیرد.
- ❖ جزئیاتی که در صورت پروژه تعیین نشده‌اند را می‌توانید بصورت دلخواه خودتان تعریف نمایید.
- ❖ زمان بازگردانی قابل تمدید نیست.
- ❖ پروژه مجموعاً 4.5 نمره از نمره کل درس را به خود اختصاص می‌دهد. (بدون احتساب نمره اضافه)
- ❖ در صورت مشاهده تشابه و اثبات آن، به میزان تشابه **نمره منفی** به افراد خاطی داده خواهد شد.

## توصیه‌ها برای پیاده سازی

- با توجه به متصل بودن ماژول‌های متفاوت به میکروکنترلر و وجود وقفه‌های متعدد، پیشنهاد می‌شود که فرکانس کاری میکروکنترلر را روی 72 مگاهرتز تنظیم نمایید.
- با توجه به اینکه واحدهای ADC1 و ADC2 و همچنین ADC3 و ADC4 وقفه‌های ادغام شده دارند، توصیه می‌شود Volume و LDR هر کدام روی یکی از دو گروه وقفه فوق قرار گیرند تا وقفه‌های جداگانه برای هر واحد داشته باشید.
- با توجه به اینکه دقت در Volume و LDR زیاد اهمیت ندارند، پیشنهاد می‌شود تا 6 بیت Resolution برای هر کدام در نظر بگیرید.
- با توجه به وجود وقفه‌های متعدد در برنامه نهایی و سرعت کاری پایین واحد ADC، توصیه می‌شود تا Sampling Time در واحد ADC را روی 601.5 سیکل قرار دهید تا میکروکنترلر زمان رسیدگی به تمامی وقفه‌ها را داشته باشد.
- در راستای حفظ هیجان بازی، برای جابجایی نشانگر در صفحه سعی کنید تا حد ممکن از تابع clear که باعث ایجاد تاخیر نسبتاً زیادی در اجرا می‌شود استفاده نکنید.

## کاراکترهای LCD Character

- توجه داشته باشید LCD کاراکتری که در اختیار دارید، علاوه بر توانایی نمایش کاراکترهای ASCII، توانایی پشتیبانی از زبان ژاپنی را نیز دارد که می‌توانید از کاراکترهای این زبان هم به شکل مستقیم استفاده کنید. (برای توضیحات بیشتر این [لینک](#) را مشاهده کنید)

	<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>A</b>	<b>B</b>	<b>C</b>	<b>D</b>	<b>E</b>	<b>F</b>
<b>0</b>	CC0			0	@	P	`	p				-	タ	ミ	α	p
<b>1</b>	CC1		!	1	A	Q	a	q			。	ア	チ	ム	ä	q
<b>2</b>	CC2		"	2	B	R	b	r			「	イ	ツ	メ	β	θ
<b>3</b>	CC3		#	3	C	S	c	s			」	ウ	テ	モ	ε	∞
<b>4</b>	CC4		\$	4	D	T	d	t			、	エ	ト	ヤ	μ	Ω
<b>5</b>	CC5		%	5	E	U	e	u			・	オ	ナ	ユ	σ	ü
<b>6</b>	CC6		&	6	F	V	f	v			ヲ	カ	ニ	ヨ	ρ	Σ
<b>7</b>	CC7		'	7	G	W	g	w			ア	キ	ヌ	ラ	g	π
<b>8</b>	CC0		(	8	H	X	h	x			イ	ク	ネ	リ	√	x̄
<b>9</b>	CC1		)	9	I	Y	i	y			ウ	ケ	ノ	ル	-1	y
<b>A</b>	CC2		*	:	J	Z	j	z			エ	コ	ハ	レ	j	千
<b>B</b>	CC3		+	;	K	[	k	{			オ	サ	ヒ	ロ	x	万
<b>C</b>	CC4		,	<	L	¥	l				ヤ	シ	フ	ワ	ç	円
<b>D</b>	CC5		-	=	M	]	m	}			ユ	ス	ヘ	ソ	£	÷
<b>E</b>	CC6		.	>	N	^	n	→			ヨ	セ	ホ	`	ñ	
<b>F</b>	CC7		/	?	O	_	o	←			ツ	ソ	マ	°	ö	■

نمره اضافه

□ پیاده سازی کی پد به صورت کاراکتری (مانند کیبورد موبایل های قدیمی - شکل 2) نمره اضافی خواهد داشت. برای پیاده سازی این مورد می توانید از **یکی** از دو روش زیر استفاده کنید. طبیعتاً به دلیل پیچیدگی بالاتر روش دوم نمره بیشتری خواهد داشت.

1. با هر بار فشردن یک کلید، کارکتر بعدی مربوط به آن کلید نمایش داده شود و در نهایت با استفاده از یک کلید تایید (دکمه پایین سمت چپ کی پد) آن کارکتر ثبت شود. (250 بیت)

2. ثبت هر کارکتر دقیقاً مانند کیبورد های موبایل بر اساس تاخیر و یا فشرده شدن کلید دیگری باشد. به این صورت که با گذشت 1.5 ثانیه از آخرین فشرده شدن یک کلید و یا فشرده شدن کلیدی غیر از کلید قبلی آن کاراکتر ثبت شود. (500 بیت)

➤ نمایش بازی به صورت آنلاین

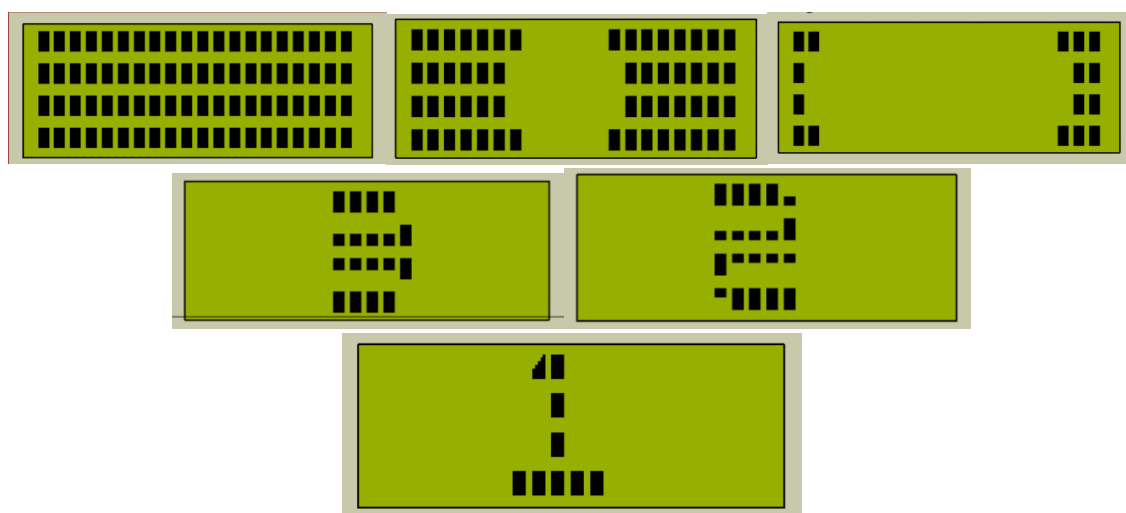
○ می‌توانید برنامه‌ی واسطی جهت نمایش زنده بازی بر روی لپ‌تاپ یا موبایل یا یک صفحه وب طراحی کنید. (500 بیت)

○ برنامه واسط می‌تواند قابلیت کنترل بازی را نیز داشته باشد. (300 بیت)

➤ الگوریتم هوش مصنوعی

○ می‌توانید الگوریتم هوش مصنوعی برای ایجاد دشمنان استفاده کنید و سختی بازی را تغییر دهید. الگوریتم شما باید سختی مراحل را متناسب با شماره مرحله تغییر دهد. (300 بیت)

➤ افزودن افکتهای گرافیکی در ابتدای شروع بازی به منظور جذاب‌تر کردن بیشتر بازی. (200 بیت)



➤ اگر برای قسمت ذخیره سازی و بارگذاری بازی کلاینت مناسب نوشته شود (با قابلیت دریافت خودکار در قسمت ذخیره‌سازی و ارسال اطلاعات هر یک از save ها و نمایش اطلاعات هر save) شامل 500 بیت نمره اضافه می‌باشد.

با آرزوی موفقیت

تیم بخش پروژه درس ریزپردازنده و زبان اسمبلی