



تمرین کامپیوتری اول بازی دوما



مبانی کامپیوتر و برنامه‌سازی - پاییز
۱۴۰۰

دانشکده مهندسی برق و کامپیوتر

مسئولان تمرین:
شایان حمیدی و حمید سالمی

موعد تحویل: ۱۹ آذر ۱۴۰۰

استاد:
دکتر مرادی، دکتر توسلی پور

مقدمه:

در این تمرین کامپیوتری از شما انتظار می‌رود که یک بازی را به زبان C پیاده‌سازی کنید. در این تمرین مهارت و علم شما در مباحث آرایه‌ها به ویژه آرایه دو بعدی، حلقه‌های for و while، پوینترها، کار با توابع و ساختارهای شرطی و Definition ها محک زده می‌شود؛ همینطور کار با توابع رندوم هم کار می‌کنید. انتظار می‌رود که از ابزار visual studio برای توسعه استفاده کنید.

شرح بازی:

احتمالاً همه شما بازی مار (snake)^۱ را انجام داده‌اید. بازی به صورت یک نفره است. کنترل یک مار، که در صفحه حرکت می‌کند، دست شماست. شما سعی می‌کنید غذاهایی را روی صفحه بخورید و امتیاز بیشتری کسب کنید. بازی که قرار است پیاده‌سازی کنید بر اساس همین بازی مار است، اما تفاوت‌های مهمی دارد. در بازی جدید، دو مار در صفحه داریم که با هم رقابت می‌کنند پس بازی دیگر رکوردی نیست بلکه رقابتی است و یک مار می‌برد و دیگری می‌سوزد. در ادامه پروژه به شرح دقیق این بازی و قوانین و عناصر آن می‌پردازیم.

شروع بازی:

دو مار در صفحه داریم که هر بازیکن یکی را انتخاب می‌کند. این دو مار در دو گوشه صفحه (پایین-راست و بالا-چپ یا بالا-راست و پایین-چپ) قرار می‌گیرند. این مارها برای دو بازیکن ۱ و ۲ هستند. طول اولیه این دو مار به اندازه ۳ کاراکتر است. در ابتدا هر کدام یکی به سمت بالا و دیگری به سمت پایین حرکت می‌کند.

عناصر بازی:

مارها:

¹ <https://playsnake.org/>

در طول بازی تمام بدن هر یک از مارها را با کاراکتر "o" و سر مارها با کاراکتر "@" نشان داده می‌شوند. این خانه‌ها در طول بازی همیشه در حال حرکت هستند، زیرا مارها همیشه در حال حرکت هستند. جهت حرکت مارها توسط کلیدهای صفحه کلید مشخص می‌شود. یکی از مارها با چهار کلید w,s,d,a حرکت می‌کند که بازیکن ۱ آنرا کنترل می‌کند و مار دیگر با چهار کلید i,j,k,l که بازیکن ۲ آنرا کنترل می‌کند. این کلیدها در حرکت هر مار برای هر بازیکن اختصاصی هستند و در طول بازی ثابت می‌مانند. تغییر جهت حرکت با توجه به چهار جهت معمول مشخص می‌شود. (یعنی w به جهت بالا، s به سمت پایین، a به سمت چپ و d هم به سمت راست. همینطور i به جهت بالا، k به سمت پایین، j به سمت چپ و l هم به سمت راست.)

مارها طول حداکثر دارند که ۳۰ کاراکتر است و اگر بیشتر موش بخورند، فایده ای برایشان ندارد.

محدوده بازی:

بازی طبیعتاً یک محدوده‌ای دارد و طول و عرض آن بر اساس تعداد کاراکتر به اندازه ۴۰ در ۴۰ است (بدون در نظرگیری دیواره‌ها). اگر مار هنگام حرکت از محدوده‌ها خارج شد از سمت معکوس خود در خانه متناظر و همینطور در همان جهت حرکت خود را ادامه می‌دهد. (مارها نمی‌بازند)

دیواره بالا و پایین با کاراکتر "-" و دیواره چپ و راست با کاراکتر "|" مشخص می‌شوند.

موش‌ها:

در صفحه بازی به صورت تصادفی پخش شده‌اند. در هر لحظه حتماً باید ۲۰ عدد موش در صفحه باشد. (در واقع اگر موشی خورده شد بلافاصله همان لحظه یکی دیگر جای دیگری ظاهر شود که آن خانه خالی باشد). موش‌ها خوراک مارها هستند. در صفحه ثابت هستند و مارها آنها را می‌توانند بخورند و یکی به طول خود اضافه کنند. موش‌ها در صفحه با کاراکتر "m" مشخص می‌شوند.

ستاره طلایی (امتیازی):

در صورتی که ماری آن را بخورد از طول مار حریف یک کاراکتر کم می‌شود. در هر لحظه حتماً باید ۳ عدد ستاره در صفحه باشد. (در واقع اگر ستاره‌ای خورده شد بلافاصله همان لحظه یک ستاره ی دیگر در جای دیگری که خالی باشد، ظاهر شود). این شی در صفحه با کاراکتر "*" (astrix) مشخص می‌شود. توجه شود اگر ماری آن را خورد و طول حریف کمتر از دو رسید، حریف باید طبق قوانین پایان بازی بسوزد.

سم (امتیازی):

در صورتی که ماری آن را بخورد از طول خود یک کاراکتر کم می‌شود. در هر لحظه حتماً باید ۵ عدد سم در صفحه باشد. (در واقع اگر سمی خورده شد بلافاصله همان لحظه یکی دیگر جای دیگری ظاهر شود که آن خانه خالی باشد). این شی در صفحه با کاراکتر "x" مشخص می‌شود.

توجه شود اگر ماری آن را خورد و به طول کمتر از دو رسید، باید طبق قوانین پایان بازی بسوزد.

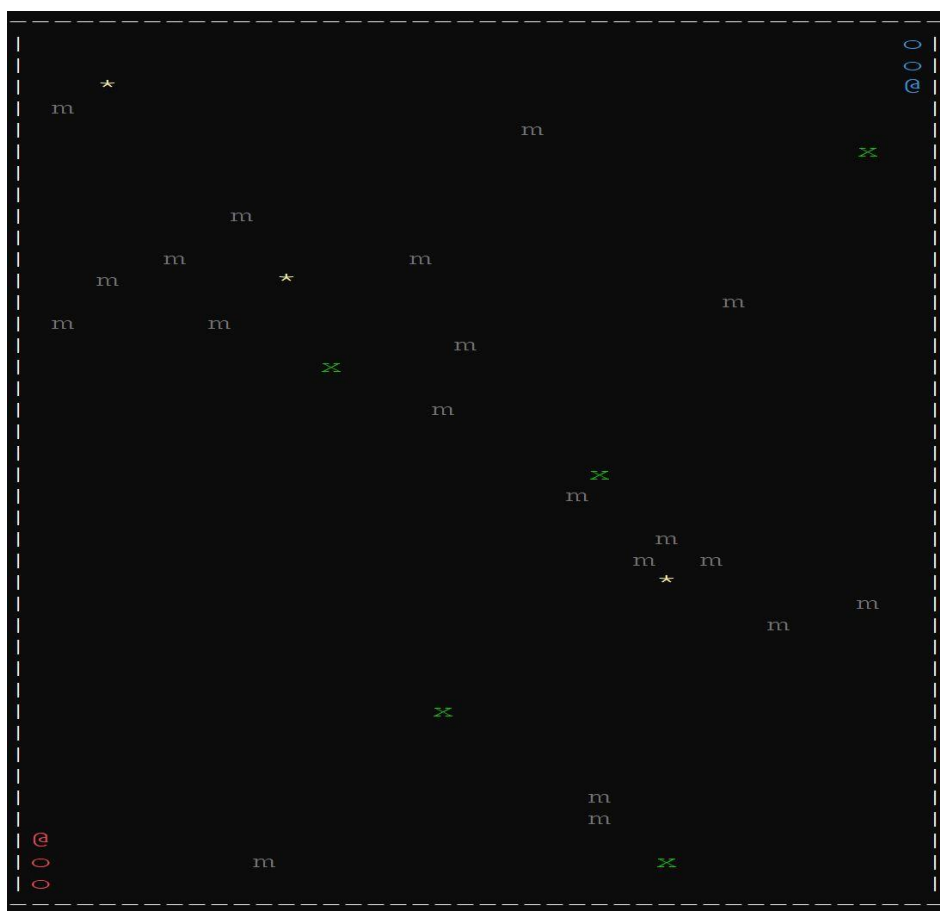
ساعت برنارد (امتیازی):

این ساعت در هر لحظه حداکثر یکی در صفحه موجود است. با کاراکتر "c" نشان داده می‌شود. هر مار با خوردن آن حریف خود را به مدت ۳ ثانیه سر جای خود فریز می‌کند. پس از ۴ ثانیه مار دیگر حرکت خود را ادامه می‌دهد. بازی باید در هر ۲ ثانیه با احتمال ۲۰٪ ساعت را در مکان رندومی که خالی باشد، ظاهر کند. برای پیاده‌سازی هر دو ثانیه بازی باید یک عدد رندوم بین ۰ تا ۱ بسازد و اگر کمتر از ۰,۲ بود آنگاه ساعت را ظاهر کند.

پایان بازی:

- هر ماری اگر به طول کمتر از دو برسد، در اثر خوردن سم یا خوردن ستاره طلایی توسط حریف، آنگاه می‌سوزد.
- هر ماری که طبق قوانین زیر بسوزد، مار حریف برنده می‌شود و بازی به پایان می‌رسد.
 - ۱. هر ماری که به بدن خود برخورد داشته باشد می‌سوزد.
 - ۲. اگر ماری با بدن مار دیگری برخورد کند، آن مار که به بدن دیگری برخورد کرده می‌سوزد.
 - ۳. اگر دو مار سر به سر شوند ماری که طول کمتری دارد می‌سوزد. اگر طول ها برابر باشند، بازی مساوی می‌شود.

پس از پایان بازی، در یک پیام کوتاه روی صفحه نتیجه بازی (برد یک مار یا تساوی) اعلام شود. به شکل زیر هنگام شروع بازی توجه کنید:



آنچه که شما باید پیاده سازی کنید:

در این پروژه برای بخش اجباری کافی است بخش های زیر را پیاده سازی کنید:

۱. پیاده سازی دو مار و حرکت آنها
 ۲. پیاده سازی موش ها و خورده شدن آنها
 ۳. شروط پایان بازی در اثر برخورد مارها به هم (به جز طول کمتر از ۲ مارها که در اثر خوردن ستاره طلایی یا سم ایجاد می شود)
- در واقع در بخش اجباری نیازی به پیاده سازی ساعت برنارد، سم یا ستاره طلایی ندارید. بدون آنها در صورت پیاده سازی درست ۱۰۰ نمره ی این تمرین را به صورت کامل می گیرید.

بخش امتیازی:

توجه کنید از سه گزینه زیر، فقط و فقط حق انتخاب یک گزینه را دارید. اگر چند گزینه امتیازی را پیاده سازی کنید، فقط یک گزینه به انتخاب شما تحویل گرفته می شود و نمره داده می شود. پس بهتر است با توجه به نمره هر یک، یکی را انتخاب و پیاده سازی کنید.

گزینه ۱: پیاده سازی سم (۱۰ نمره)

توجه کنید باید طبق شرح بازی تمام ویژگی های این بخش از جمله کاهش طول و سوختن در صورت طول کمتر از ۲ را پیاده سازی کنید.

گزینه ۲: پیاده سازی ستاره طلایی (۱۵ نمره)

توجه کنید باید طبق شرح بازی تمام ویژگی های این بخش از جمله کاهش طول حریف و سوختن حریف در صورت طول کمتر از ۲ را پیاده سازی کنید.

گزینه ۳: پیاده سازی ساعت برنارد (۲۵ نمره)

توجه کنید باید طبق شرح بازی تمام ویژگی های این بخش از جمله ظاهر شدن آن و یخ زدن مارها سر جایشان را پیاده سازی کنید. توجه کنید برای این بخش باید از تابع `GetTickCount` ویندوز استفاده کنید. برای اطلاعات بیشتر از این تابع به [لینک](#) مراجعه کنید. همینطور برای نحوه استفاده از آن می توانید به تابع `wait_and_get_direction` مراجعه کنید که به شما داده شده است.

چیزی که به شما می دهیم:

یک فایل به نام `Domar.c` به شما تحویل داده می شود. در این فایل، موارد زیر را می بینید:

تعدادی کتابخانه:

این کتابخانه ها در طول برنامه استفاده می شوند. پس به آنها دست نزنید. اگر کتابخانه بیشتری خواستید می توانید `include` کنید.

تعدادی Definition:

- به بخش colors جهت تغییر رنگ کاراکتر در صفحه استفاده شده است ترجیحا با این بخش کاری نداشته باشید.
 - بخش directions هم تعریف چهار جهت اصلی است که احتمالا به کار شما هم بیاید.
 - بخش general بخشی است که در آن اعدادی که در صورت پروژه تعریف شده است را define میکنیم.
- به طور مثال ضلع صفحه بر اساس تعداد کاراکتر، طول ابتدایی مارها، حداقل و حداکثر طول مارها و تعداد موش در صفحه در این بخش آمده اند. در طول پروژه اگر این اعداد را می خواستید، از این بخش استفاده کنید.
- اما مهمترین بخش board_characters است. از این کاراکترها حتما باید استفاده کنید تا در ادامه بتوانید از تابع draw_board استفاده کنید. در واقع برای وضعیت صفحه نیاز به یک آرایه دو بعدی از جنس char دارید که باید به این تابع بدهید.
 - بخش آخر، مربوط به گزینه های امتیازی است. در طول پروژه می توانید Definition های دیگری هم تعریف کنید و از سپس آنها استفاده کنید.

تعدادی تابع:

پیاده سازی چند تابع به شما داده شده است. مهم نیست که چگونه پیاده سازی شده اند، مهم این است که چگونه از آنها استفاده می کنید. امضا (prototype) چهار تابع آمده است. (در بخش prototypes بالای تابع main). این توابع، توابع اصلی هستند که مفید هستند و ما انتظار داریم از آنها استفاده کنید. بقیه توابع که امضایی هم ندارند، به عنوان تابع کمکی هستند و در این چهار تابع استفاده می شوند. پیاده سازی و استفاده از آنها هم مهم نیست و انتظار می رود استفاده نکنید. سه متغیر گلوبال هم هستند که مهم نیستند و توجهی نکنید. در ادامه به توضیح این توابع می پردازیم:

- **init_screen:** این تابع یک بار اول بازی صدا بزنید تا کنسول برای بازی آماده شود.
- **reset_screen:** این تابع را در صورت نیاز صدا بزنید تا کنسول به حالت اولیه بازگردد.
- **draw_board:** یک آرایه دوبعدی به اندازه BOARD_SIZE در BOARD_SIZE به آن دهید که وضعیت صفحه را مشخص می کند، یعنی در هر خانه که به اندازه یک کاراکتر است، چه موجودیتی قرار دارد. (بدن مار اول، موش و ...) طبق بخش اول باید از کاراکترهایی که در Definition ها آمده است، استفاده کنید. این تابع کنسول را پاک می کند، و وضعیت جدید (که به آن دادید) را روی کنسول می کشد.
- **wait_and_get_direction:** پوینتر دو متغیری که برای جهت حرکت دو مار نگه داشتید را به این تابع بدهید، این تابع در یک بازه زمانی از صفحه کلید، آخرین وضعیت فشرده شدن کلیدها را می گیرد، و جهت های جدید مارها را اعمال می کند.

- **تابع main:** برنامه از این نقطه شروع می شود، پس شما از اینجا شروع به کد زدن کنید. بدنه کلی این تابع به عنوان راهنمایی قرار داده شده است. در کامنتها توضیحات اضافه را می بینید و نشان می دهند که در هر تکه باید چه کنید. توجه کنید پیاده سازی خود را در این فایل باید انجام دهید تا تابع ها را در کنار آن داشته باشید. در نهایت فقط یک فایل تحویل دهید.

معیارهای نمره دهی:

نمره(از ۱۰۰)	معیار
5	نام گذاری مناسب متغیرها و توابع
5	استفاده از Define به جای اعداد و کاراکترهای ثابت
5	استفاده مناسب از توابع
5	کامنت گذاری
10	شرایط درست صفحه هنگام شروع بازی
10	پیاده سازی درست حرکت مارها
10	پیاده سازی درست خوردن موش ها
10	پیاده سازی درست پایان بازی و نمایش نتیجه
40	تسلط بر پروژه
۱۰ + یا ۱۵ + یا ۲۵ +	بخش امتیازی(بالا توضیح داده شد)

نکات:

۱. به تقارن بازی توجه کنید! هر تکه کدی که برای مار اول استفاده می شود، با تغییراتی برای مار دوم هم می تواند استفاده شود.(یا برعکس)

۲. هر بار حلقه while در تابع main دوباره اجرا می شود، صفحه پاک و دوباره کشیده می شود. این تکنیک باعث می شود حرکت مارها و پویایی صفحه کنسول ایجاد شود.

۳. دقت کنید که هر بار صفحه پاک می شود و دوباره کشیده می شود، حس می کنید کنسول در حال چشمک زدن است. دلیل آن این است که با تناوب کم داریم حدود ۱۶۰۰ کاراکتر را هر بار می کشیم. این طبیعی است زیرا ما از ابزار گرافیکی استفاده نمی کنیم و فقط داریم از خود کنسول استفاده می کنیم. پس مشکلی نیست.

۴. فراموش نکنید در ابتدا متغیرها را به ویژه board_content را مقداردهی اولیه کنید.

قوانین تحویل:

مهلت تحویل این پروژه تا واپسین ساعات روز جمعه ۱۹ آذر ماه خواهد بود. و پس از آن به ازای هر روز تاخیر ۱۰ درصد از نمره ی پروژه کسر خواهد شد. لازم به ذکر است تاخیر بیش از سه روز در این پروژه مجاز نخواهد بود.

استفاده از دستور goto ممنوع بوده و در صورت استفاده نمره ی صفر در نظر گرفته خواهد شد.

استفاده از متغیر های جهانی (global) نیز ممنوع بوده و در صورت استفاده نمره ی بخش پیاده سازی کسر خواهد شد. پروژه ها برای یادگیری برنامه نویسی و مباحث مطرح شده در کلاس طراحی می شوند و انجام آنها به صورت انفرادی خواهد بود. همچنین در صورت کشف شباهت میان دو پروژه (که به وسیله ی نرم افزارهای مربوطه بررسی میشود)، برای هر دو نفر نمره صفر در نظر گرفته خواهد شد.

فایل های خود با فرمت c و h را در یک فایل zip با نام CA1_[SID].zip در سایت Elearn و در بخش CA1(upload here) بارگذاری نمایید.

برای مثال اگر شماره دانشجویی شما ۸۱۰۱۰۰۰۰ باشد باید نام فایل خود را CA1_810100000.zip قرار دهید.

در صورت وجود هرگونه سوال می توانید پرسش های خود را در تالار پرسش و پاسخ مربوط به این پروژه درس در صفحه Elearn درس مطرح نمایید، یا به icsp00.ca@gmail.com ایمیل بزنید.

اعلانات مربوط به این پروژه و نیز سوالات مطرح شده در تالار پرسش و پاسخ، بخشی از پروژه بوده و رعایت کردن آن ها الزامی است.

موفق باشید