

بسم الله الرحمن الرحيم

CareMaze: A Game With Algorithmic Thinking



شناسه ی اثر:

نام گروه: HMH

زبان برنامه نویسی استفاده شده: Microsoft C#

اعضای گروه: ابوالفضل دهاقین - محسن مشرفی



نام دبیرستان: دبیرستان نمونه دولتی شیخ زاده هراتی

متقاضی شرکت در سومین دوره مسابقات کشوری کدنویسی
پژوهش سراهای دانش آموزی: برنامه نویسی ویندوز با تفکر
الگوریتمی - سال تحصیلی ۱۳۹۹-۱۴۰۰

راهنمای بازی:

در این بازی کاربر باید در ابتدا بر روی **پیکسل سبز رنگ** در صفحه شطرنجی کلیک کند. کلیک کردن بر روی این پیکسل به منزله ی شروع بازی می باشد. سپس با استفاده از موس، مسیر مشخص شده را دنبال کند (مثل بازی اعصاب سنج). اگر موس کاربر از مسیر مشخص شده خارج شود بازی را می بازد و پیامی تحت عنوان You Lose نمایش داده می شود. همچنین کاربر می تواند با نگه داشتن یکی از کلید های موس «استراحت» بکند. در این حالت کاربر تا زمانی که یکی از کلید های موس را نگه داشته آزاد است موس را در تمام صفحه حرکت دهد اما باید توجه داشته باشد که هنگام رها کردن کلید، نشانگر موس روی مسیر قرار گرفته باشد. همچنین کاربر میتواند مسیر را بریده بریده طی کند اما باید تمام مسیر را طی کرده باشد. در نهایت پس از رسیدن به **پیکسل قرمز رنگ** اگر تمام مسیر بدون مشکل طی شده باشد کاربر برنده شده و پیامی تحت عنوان You Win نمایش داده می شود.

دکمه های زیر صفحه ی شطرنجی به ترتیب از راست به چپ برای:

- بارگذاری مجدد صفحه ی بازی (در اصل، تغییر مسیر بازی)
- ورود به بخش تنظیمات بازی
- ورود به راهنما و توضیحات اثر می باشند.

توضیحات و ارزش الگوریتمی بازی:

این بازی با استفاده از تکنولوژی Windows Form Application و در زبان برنامه نویسی سی شارپ نوشته شده که پیش نیاز اجرای آن نصب بودن NET Frame Work 4.0. به بعد می باشد.

می توان گفت چالش های اصلی نوشتن برنامه ی این بازی برای ما الگوریتم ساختن مسیر و نوشتن توابع برای پیکسل ها در زمان بهینه و در نهایت ایجاد تنظیمات برای بازی بود.

ما از کلاس System.Random برای ساختن مختصات شانسی (رندم) استفاده کردیم تا بتوانیم با هر بار بارگذاری مجدد بازی، مسیری جدید برای کاربر ایجاد کنیم. تمام مراحل نصب پیکسل ها در هنگام بارگذاری فرم انجام می شود (علت زمان ببری بارگذاری بازی هم همین است) و از آنجا که بیش از ۱۰۰۰ پیکسل در این صفحه موجود است مدیریت کردن تک تک آنها غیر ممکن می باشد.

پس کلاس دیگری تحت عنوان lbl ساختیم تا با توجه به نوع پیکسل رویداد های شخصی آن را تعریف کنیم که در نتیجه به طور کلی ۴ نوع پیکسل در برنامه داریم:

- پیکسل نقطه ی شروع

- پیکسل مسیر

- پیکسل خالی یا پیش فرض

- پیکسل نقطه ی پایان

که برای هر کدام کد مربوط به رویداد آن را نوشتیم.

همچنین برای تعریف و ذخیره کردن نقاط مسیر از یک داده ساختار پویا مانند List، و برای اضافه کردن نقاط مسیر(برای ساختن مسیر معتبر) از شروط زیر استفاده کردیم:

- نقطه ای که قرار است اضافه شود در لیست مختصات ها نباشد(از قبل رنگی نشده باشد)

- دقیقا یک نقطه اطراف نقطه ی مورد نظر در لیست مختصات ها باشد(تنها یک نقطه اطراف نقطه ی مورد نظر رنگی باشد که یعنی راهی که قرار است بسازیم خودش را قطع نکند)

- مختصات نقطه ای که قرار است اضافه شود کمتر یا بیشتر از مختصات صفحه ی شطرنجی نباشد(از صفحه ی بازی بیرون نزنند)

در نهایت از کلاس System.Xml برای کار با فایل های ایکس ام ال و ذخیره ی تنظیمات بازی از قبیل رنگ و اندازه ی پیکسل ها استفاده کردیم و برخی از توابع و متغیر های کاربردی آن را در کلاس Ibl که خود، ساخته بودیم تعریف کردیم که آن خود چالشی دیگر بود....

شما می توانید برای درک بهتر عملکرد برنامه و کدنویسی انجام شده به آدرس مخزن زیر در گیت هاب بروید و SourceCode این اثر را ملاحظه بفرمایید(تمامی کد ها با زبان انگلیسی کامنت گذاری و تشریح شده اند):

<https://github.com/hooshemadhoosh/CareMaze.git>