

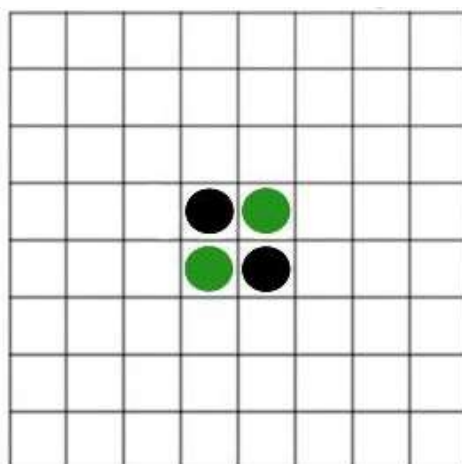


## Othello

بازی اتللو در رده اول بازی های فکری دنیا است که از تعدادی مهره با دو وجه رنگی و یک صفحه بازی تشکیل شده است. این بازی به صورت دو نفره انجام می شود. شما قرار است این بازی را در دو بخش پیاده سازی کنید.

### توضیح بازی

در ابتدا بازی با دو مهره سیاه و دو مهره سفید آغاز می شود.



بازیکن صاحب مهره های سیاه بازی را آغاز می کند. هر یک از دو بازیکن به نوبت یک حرکت انجام می دهند. هر بازیکن فقط می تواند مهره اش را در جایی قرار دهد که یک یا چند مهره حریف را محاصره کند. انجام حرکت به معنی گذاشتن یک مهره (همرنگ بازیکن) در صفحه و محصور کردن یک یا چند مهره حریف در یک یا چند راستا است (عمودی، افقی و یا قطری). در نتیجه مهره های محاصره شده را برگردانده و به رنگ مهره خود در آورید. این به معنی محاصره و تصاحب مهره های حریف و تغییر آن ها به مهره ی هم رنگ بازیکن است.

در انتهای بازی، بازیکنی که صاحب مهره های بیشتری باشد، به عنوان برنده انتخاب می شود.

## توضیح پروژه

در بخش دوم پروژه‌ی اتللو قرار است که بازی را به صورت گرافیکی پیاده‌سازی شود. برای آموزش نحوه‌ی استفاده از دستورات گرافیکی Visual Studio ، یک ویدیوی آموزشی در صفحه‌ی درس قرار داده شده که با نوشتن یک برنامه‌ی نمونه، نحوه‌ی عملکرد آن آموزش داده شده‌است. همچنین declaration بعضی از توابع این کتابخانه که می‌تواند مورد استفاده قرار گیرد به صورت زیر است:

- void line (int x1, int y1, int x2, int y2)
- void circle (int x, int y, int radius)
- void setcolor (int color)
- void floodfill (int x, int y, int border)
- void setfillstyle (int pattern, int color)
- int initwindow (int width, int height, const char\* title="Windows BGI", int left=0, int top=0)

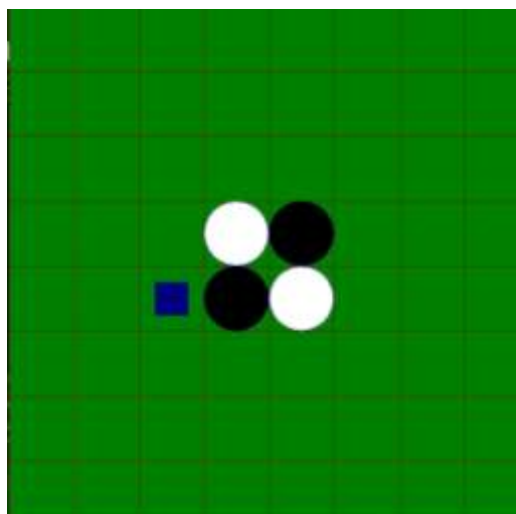
همچنین از این [لینک](#) می‌توانید برای آشنایی با توابع این کتابخانه استفاده کنید.

### 1 - نمایش صفحه‌ی بازی

در این مرحله شما باید به جای نمایش نقشه‌ی بازی در صفحه command line، صفحه‌ی گرافیکی بازی را نشان دهید. همانطور که می‌دانید، این صفحه مربعی 8 در 8 است و در ابتدای بازی در 4 خانه‌ی وسط صفحه 2 مهره‌ی سفید و 2 مهره‌ی سیاه به صورت ضربدری قرار گرفته‌اند.

همچنین در خانه‌ی 0 جدول (مبدأ)، یک مربع کوچک قرار گرفته است که نشان دهنده‌ی جای انتخاب شده برای قرار دادن مهره است که این مربع در مراحل بعدی برای انتخاب جای مهره در صفحه حرکت می‌کند. توجه کنید که این مربع نباید از صفحه‌ی بازی خارج شود.

ابعاد و رنگ‌های استفاده شده در این صفحه در اختیار خودتان است و مثالی از آن در شکل زیر آمده است.



ورودی خروجی 1 : نمایش نقشه بازی

## 2 - مشخص کردن جای مهره

در هر مرحله از بازی با توجه به نوبت بازی، شما باید با گرفتن کاراکتر های w, a, s, d جای مهره جدید را در صفحه-ی بازی مشخص کنید، که w جهت بالا، s جهت پایین، a جهت چپ و d جهت راست است. لازم به ذکر است که با هر جهتی که وارد می‌شود، کاربر باید بتواند مربع فعلی (مکان فعلی) را ببیند و در انتها با کلید space مهره را در آنجا قرار دهد.

```
BLACK player's turn :
```

توجه کنید که در این قسمت اگر ورودی اشتباه وارد شد باید با پیغام مناسب به کاربر اطلاع دهید و دوباره از کاربر ورودی بگیرید.

## 3 - پایان بازی

در هر مرحله از بازی، شما موظف هستید که صفحه بازی را به درستی چک کنید. حالتی وجود دارد که در آن‌ها به هیچ یک از بازیکن‌ها اجازه‌ی انجام حرکت بیشتر داده نمی‌شود و یا تمامی خانه‌های جدول، پر شده‌است. در این حالات بازی باید به اتمام برسد و خاتمه یابد.

## 4 - نمایش نتایج

در پایان بازی برنامه شما باید امتیاز هر بازیکن و برنده‌ی بازی را نمایش دهد. این نمایش به انتخاب شما می‌تواند در فضای Command line و یا محیط گرافیکی باشد.

```
The WHITE player has won!  
Black: 25  
White: 39
```

ورودی خروجی ۲ : خروجی نمونه برای اعلام نتایج نهایی

## چند نکته و راهنمایی

- برای پیاده‌سازی این پروژه، کافی است از روند بازی در فاز قبلی این پروژه استفاده کنید و در هر مرحله به جای نمایش صفحه‌ی بازی در **command line**، تغییرات لازم را در صفحه‌ی گرافیکی بازی انجام دهید.
- برای دانشجویانی که نتوانسته اند فاز یک پروژه را کامل انجام دهند، کد کامل فاز یک بر روی صفحه درس قرار خواهد گرفت.
- برای آشنایی با توابع گرافیکی ویدیو و لینکی در صفحه‌ی درس قرار میگیرد که میتوانید از آن ها استفاده کنید.

## نحوه‌ی نمره‌دهی

عنوان	نمره
نام‌گذاری مناسب و اصولی متغیرها	۵
استفاده از تمام ورودی‌های تابع در آن	۵
نام‌گذاری مناسب توابع	۵
عدم وجود قطعه کد تکراری	۱۰
نمایش درست صفحه‌ی گرافیکی بازی	۱۰
دریافت صحیح ورودی در محیط گرافیکی	۵
نشان دادن محل فعلی با استفاده از یک مربع	۵
رسیدگی به خطاها	۱۰
تغییر درست جای مربع مشخص کننده‌ی مکان فعلی مهره	۱۰
تغییر درست نوبت ها با توجه به شرایط بازی و خطاها	۵
به‌روز رسانی درست صفحه بازی	۱۰
کامنت‌گذاری مناسب در هرجایی که نیاز به مستندسازی دارد	۵
تست و اجرای برنامه	۱۵
پیاده سازی زیبای محیط گرافیکی (امتیازی)	۱۰

## نحوه‌ی تحویل پروژه :

شما باید فایل‌های c و h خود را در قسمت در نظر گرفته شده با عنوان CA2 Phase2[Upload here] در صفحه‌ی درس در سایت CECM آپلود نمایید. مهلت آپلود تا ساعت ۲۳:۵۵ روز دوشنبه ۲ دی است. تأخیر در سه روز نخست به ازای هر روز ۱۰ درصد و در دو روز دوم به ازای هر روز ۱۵ درصد جریمه خواهد داشت. پس از این پنج روز، به هیچ وجه نمره‌ای در نظر گرفته نخواهد شد. همچنین توجه کنید که با توجه به تأخیر در نظر گرفته شده به هیچ وجه امکان تمدید پروژه‌ها وجود ندارد.

فایل‌های خود را در یک فایل زیپ با فرمت "zip" و با نام CA2P2\_SID.zip قرار دهید که SID همان شماره‌ی دانشجویی شماست. برای مثال اگر شماره‌ی دانشجویی شما ۸۱۰۱۹۸۰۰۰ باشد باید نام فایل خود را CA1\_810198000.zip قرار دهید. در صورت عدم رعایت این موضوع، برای پروژه شما نمره صفر منظور می‌گردد.

پروژه‌ها برای یادگیری برنامه نویسی و مباحث مطرح شده در کلاس طراحی میشوند و انجام آنها به صورت انفرادی خواهد بود. همچنین، در صورت شباهت میان دو پروژه (که به وسیله ی نرم افزارهای مربوطه چک میشود) برای هر دو نفر نمره ی صفر در نظر گرفته خواهد شد.

در صورت وجود هرگونه سوال میتوانید پرسشهای خود را در فروم درس (در بخش مربوط به این پروژه) مطرح نمایید یا به [icsp98.ca@gmail.com](mailto:icsp98.ca@gmail.com) ایمیل بزنید.