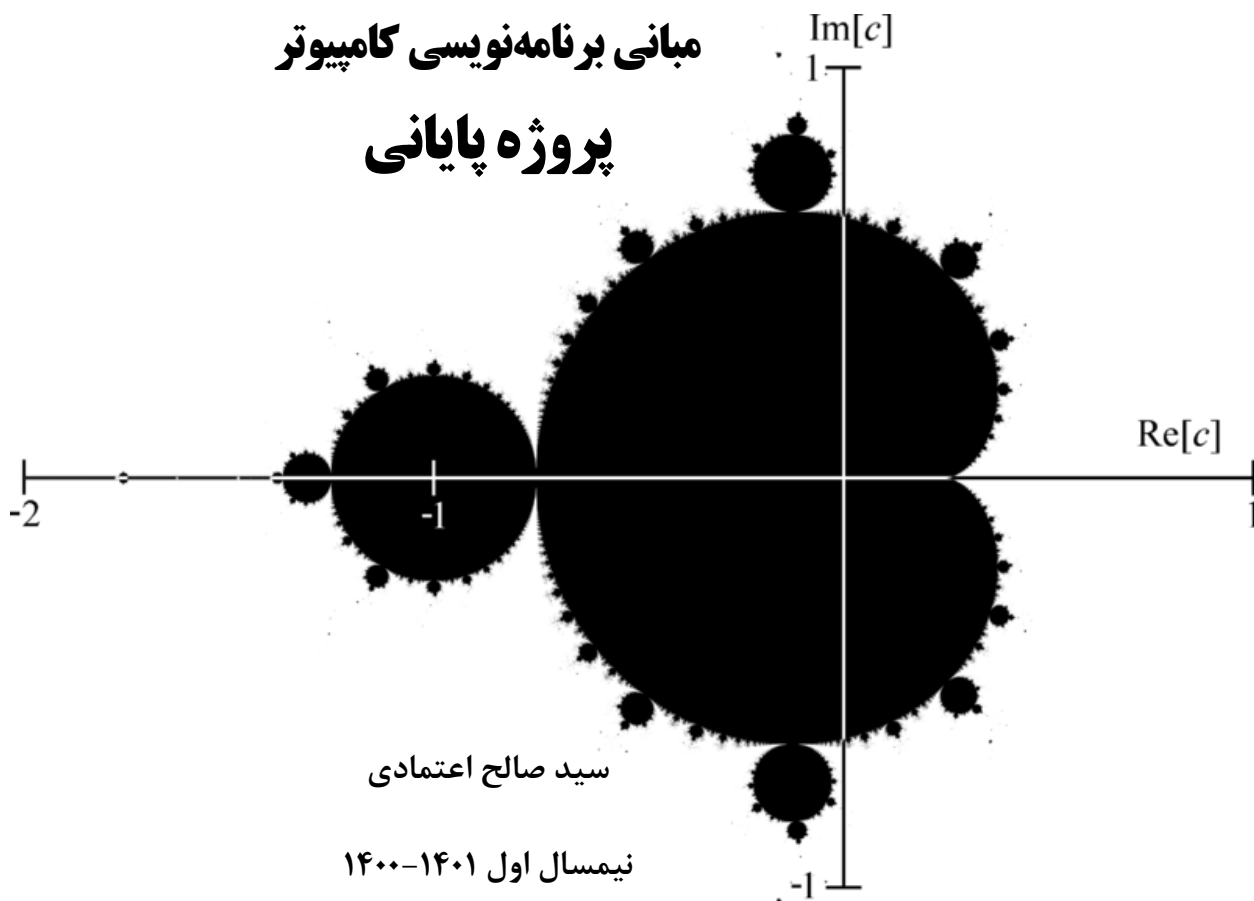




دانشکده مهندسی کامپیوتر

مبانی برنامه نویسی کامپیوتر  
پروژه پایانی



سید صالح اعتمادی

نیمسال اول ۱۴۰۱-۱۴۰۰

مهلت تحویل: ۲ بهمن

زمان ارائه: ۵ بهمن ماه ساعت ۱۴ تا ۱۶

## ۱ مقدمه

انسان در طول قرن‌ها ابزارهای زیادی مثل بیل و کلنگ و کاغذ و قلم برای کشاورزی و توسعه علم و غیره ساخته است. اما بدون شک کامپیوتر قوی‌ترین و دقیق‌ترین ابزار ساخته بشر تاکنون است. به همین دلیل امروزه تحقیق و توسعه تقریباً در هیچ زمینه‌ای بدون کامپیوتر امکان‌پذیر نیست. برنامه‌نویسی کامپیوتر جامع‌ترین روش برای بکارگیری این ابزار است. بعد از آشنایی با اصول اولیه این ابزار بهترین راه کسب مهارت، تجربه استفاده از ابزار است. مانند نوازنده یک ساز موسیقی که بعد از سال‌ها استفاده از ساز کم‌کم با ساز یکی می‌شود و حس و حال خود را می‌تواند با موسیقی بیان کند. اما در ابتدا بخش‌های مختلف ساز را بصدا در می‌آورد و صدای آن را می‌شنود. کم‌کم نوت‌های مختلف را ترکیب می‌کند تا ببیند چه صدایی دلنشین می‌شود و چه صدایی دلخراش.

علاوه بر اینکه برنامه‌نویسی خود یک هنر است، اما به عقیده نویسنده قوی‌ترین و دقیق‌ترین ابزار تولید هنر نیز می‌باشد زیرا هیچ محدودیتی در تولید انواع هنرهای تجسمی یا شنیداری ندارد. اما این ابزار تولید هنر برای اکثر افراد «کوک» نشده است و روش، ایده‌ها و الگوهای تولید صدا و تصویر دلنشین را نمی‌دانند.

موضوع این پروژه بکارگیری برنامه‌نویسی برای تولید یک اثر هنری صوتی/تصویری است. هدف ما تکرار فراوان چرخه تصور-پیاده‌سازی-ارزیابی است. به این شکل که شما یک تصویر و آهنگ تصور کنید. سپس آنرا پیاده‌سازی کنید، بعد به سلیقه خود ارزیابی کنید که آیا خوب است یا نه، و نقص آنرا رفع کنید. یکی از مزایای این روش این است که خروجی نهایی درست و غلط ندارد. سلیقه شماست. همچنین هر چقدر روی آن وقت بگذارید، قطعاً می‌توانید ایده‌های جدیدتری پیاده‌سازی کنید و خروجی نهایی را بهتر کنید. تکرار فرایند تصور-پیاده‌سازی-ارزیابی است که شما را به کامپیوتر نزدیک‌تر کرده و مهارت برنامه‌نویسی کامپیوتر شما را افزایش می‌دهد. با توجه با اینکه این فرآیند در جهت ذوق و سلیقه شما پیش می‌رود، امیدواریم فرآیند رضایت‌بخش و دلپذیری برای شما باشد.

هدف اصلی پروژه خلق یک اثر سمعی/بصری هنری توسط کامپیوتر است. برخی آثار هنری برنامه‌نویسی را می‌توانید در [این آدرس](#) ببینید. همچنین هر ساله در ماه ژانویه وبسایت <https://genuary.art> حرکتی در جهت تولید هنر توسط کامپیوتر به اجرا می‌گذارد و با هشت تگ #GENUARY در شبکه‌های اجتماعی منتشر می‌شود. اما برای اینکه نقطه شروعی برای همه وجود داشته باشد و در مدت کوتاه قابل انجام باشد، نمودار Mandelbrot-Set را در نظر گرفتیم. در این پروژه شما یک ویدیوی یک تا سه دقیقه‌ای از سیر خود در این مجموعه تولید خواهید کرد. تصاویر و موسیقی این ویدیو بصورت کامل و از صفر توسط برنامه کامپیوتری شما تولید خواهند شد. [این صفحه وب](#) و [این صفحه وب](#) توضیح و انیمیشن خوبی برای فهم مدارها و شکل مندلبرو دارند. از [این ابزار](#) آنلاین برای زوم کردن روی بخش‌های مختلف این شکل می‌توانید استفاده کنید. ابزارهای موجود در [این صفحه](#) نیز انواع روش‌های رنگ‌بندی و زوم کردن و ... را پیاده‌سازی کرده است.

## ۲ بخش‌های مختلف پروژه

در پیاده‌سازی این اثر هنری از ابزارهای زیر استفاده خواهید کرد.

۱. زبان برنامه‌نویسی پایتون برای پیاده‌سازی اولیه و ایده‌پردازی
۲. زبان برنامه‌نویسی سی برای تولید تصاویر لازم برای نسخه نهایی ویدیو
۳. زبان برنامه‌نویسی SonicPi برای تولید موسیقی متن
۴. ابزار ffmpeg برای ترکیب تصاویر و موسیقی، و تولید ویدیوی نهایی

## ۳ قابلیت‌های برنامه به زبان سی

توصیه می‌شود برای پیاده‌سازی حتما ابتدا از زبان پایتون شروع کنید. اگر نه در پیچ و خم‌ها و خطاهای زبان سی گم می‌شوید و نمی‌دانید اشکال از منطق برنامه شماست یا اشکال برنامه‌نویسی دارید. برنامه‌هایی که به زبان پایتون می‌نویسید را نیز نگه داشته و در انتها تحویل دهید. اما این کدها فقط برای سعی و خطا و ایده‌پردازی است و ورودی/خروجی مشخص برای آنها انتظار نداریم. ولی برنامه‌ای که به زبان سی می‌نویسید لازم است که بر اساس اطلاعات ورودی تعدادی تصویر به ترتیب تولید کند.

### ۱-۳ تولید تصویر اولیه مجموعه مندلبرو

برنامه شما باید بتواند با اطلاعات اولیه زیر تصویر شماره یک را بسازد. برای جزئیات بیشتر در مورد مندلبرو به [این لینک](#) مراجعه کنید. همچنین برای روش‌های ترسیم و رنگ‌بندی به [این لینک](#) مراجعه کنید.

- ابعاد تصویر به پیکسل
- بازه افقی و عمودی محور مختصات در تصویر

### ۲-۳ بزرگ‌نمایی

پس از تولید تصویر اولیه (یا اتمام یکی از عملیات‌های دیگر)، برنامه شما باید بتواند با گرفتن اطلاعات زیر تصویر را به تدریج بزرگ‌نمایی کند و مراحل بزرگ‌نمایی را بصورت تصاویر جداگانه یکی پس از دیگری تولید کند.

- میزان بزرگ‌نمایی
- تعداد مراحل میانی

### ۳-۳ چرخش

پس از تولید تصویر اولیه (یا اتمام یکی از عملیات‌های دیگر)، برنامه شما باید بتواند تصویر را به میزان دلخواه و مراحل خواسته شده با اطلاعات زیر بصورت تصاویر جداگانه تولید کند. برای [فرمول چرخش](#) اینجا را ببینید.

- میزان چرخش
- تعداد مراحل میانی

### ۴-۳ تغییر مرکز

پس از تولید تصویر اولیه (یا اتمام یکی از عملیات‌های دیگر)، برنامه شما باید بتواند با اطلاعات زیر مرکز تصویر را به تدریج از مختصات فعلی به مختصات خواسته شده تغییر دهد.

- مختصات جدید مرکز (عمودی و افقی)
- تعداد مراحل میانی

### ۵-۳ توقف

پس از تولید تصویر اولیه (یا اتمام یکی از عملیات‌های دیگر)، برنامه شما باید بتواند بدون تغییر تعدادی تصویر یکسان با تعداد خواسته شده تولید کند. این قابلیت برای توقف فیلم در یک مکان استفاده خواهد شد.

## ۴ ورودی/خروجی برنامه سی

برنامه سی شما نام یک فایل ورودی را به عنوان پارامتر دریافت می کند و بر اساس محتویات فایل تعدادی عکس تولید می کند. قالب فایل ورودی به شکل زیر است.

```
width*height
minx,maxx,miny,maxy
zoom zoom:steps
center newx,newy:steps
rotate angle:steps
hold steps
```

شکل ۱ - قالب فایل ورودی

به عنوان یک نمونه فایل به محتوای زیر توجه کنید. در این فایل پس از رسم تصویر اولیه با رزولوشن ۱۸۰۰ در ۱۲۰۰، طی ۲۰ مرحله تصویر را به اندازه ۲ بزرگ نمایی می کند. سپس مرکز تصویر را به مختصات داده شده طی ۲۰ مرحله تغییر می دهد. پس از آن تصویر را طی ۱۰ مرحله به اندازه ۹۰ درجه می چرخاند. پس از آن ۱۰ تصویر یکسان تولید کرده و به همین ترتیب تا آخر فایل ۱۵۱ تصویر تولید می کند.

```
1800*1200
-1.5,1.5,-1,1
zoom 2:20
center -0.7746806106269039,-0.1374168856037867:20
rotate 90:10
hold 10
zoom 2:20
rotate 90:10
hold 10
center -0.79,-0.1374168856037867:50
```

شکل ۲ - نمونه فایل ورودی

برنامه شما نام این فایل ورودی را به عنوان پارامتر دریافت کرده و طبق دستورات آن فایل های خروجی تصویری با نام های ترتیبی 00000.bmp, 00001.bmp, 00002.bmp, ... تولید می کند. همچنین یک فایل به نام audio.txt تولید کرده که محتوای آن به اختیار شماست و باید برای تولید موسیقی مناسب با SonicPi استفاده شود. مثلاً اگر اسم فایل ورودی play.cfg باشد و نام برنامه سی شما mbs\_project.exe باشد، استفاده از دستور زیر باید عکس ها و فایل audio.txt را تولید کند.

msb\_project.exe play.cfg

## ۵ تولید ویدیوی تصویری

با استفاده از دستور ffmpeg مشابه آنچه در کلاس نشان دادیم فایل های تولید شده توسط برنامه سی را می توانید به یک ویدیو تبدیل کنید. با توجه به تعداد فریم بر ثانیه، طول ویدیوی شما به ثانیه مشخص خواهد شد. مثلاً اگر ۱۰ فریم در ثانیه استفاده کنید، ۱۵۰ عکس، ۱۵ ثانیه زمان خواهد برد. به عنوان مثال دستور فایل های تصویر را به ترتیب عددی با هم ترکیب کرده (هر ۵ تصویر یک ثانیه) و یک فیلم به نام out.mp4 تولید می کند.

```
ffmpeg -r 5 -i %05d.bmp -c:v libx264 -vf fps=25 -pix_fmt yuv420p out.mp4
```

## ۶ تولید موسیقی

بعد از تولید ویدیو و نوبت به تولید موسیقی متن می‌رسید. با استفاده از audio.txt در محیط SonicPi یک موسیقی مناسب تولید کرده و آنرا بصورت یک فایل صوتی استخراج کنید. برای آشنایی بیشتر به اسلایدها و ویدیوی کارگاه آموزشی SonicPi که برگزار شد، مراجعه کنید. فایل اسلاید در [گروه](#) به اشتراک گذاشته شده است. همچنین ویدیوی کارگاه هم در بخش Recordings گروه مبنای در تیمز موجود است.

## ۷ خلق اثر هنری

در نهایت پس از آماده شدن موسیقی و مجدداً با استفاده از برنامه ffmpeg همه عکس‌ها و فایل موسیقی را با هم ترکیب کرده و نسخه نهایی ویدیو را تولید کنید. پس پیاده‌سازی موفقیت‌آمیز قابلیت‌های برنامه، لازم است سناریوهای مختلف زوم، چرخش، تغییر مرکز، ... را با ترتیب‌های مختلف و موسیقی متن مختلف تجربه کنید تا اثر هنری دلخواه شما تولید شود. برای ترکیب فایل‌های تصویر و فایل صوتی می‌توانید از دستور زیر استفاده کنید.

```
ffmpeg -r 5 -i %05d.bmp -i music.wav -c:v libx264 -vf fps=25 -pix_fmt yuv420p out.mp4
```

## ۸ از کجا شروع کنیم؟

برای آمادگی پروژه من بجز به بخش‌های جزئی همه کد لازم را زدم. واقعا خیلی زیاد و سخت نیست. ولی خوب برای اینکه کد کار بکند باید دقیق زده بشه. تعریف اکثر structها، توابع و پیاده‌سازی بعضی از آنها را نگه داشتیم و بقیه را حذف کردم. دوتا چیز جدید استفاده کردم که درس نداده بودم. ولی چیزهای مهم، ضروری یا پیچیده‌ای نیستند. یکی enum و یکی هم دو خط اول فایل‌های H..اولی که بجاش از int هم می‌توانید استفاده کنید و دومی هم برای اینه که به فایل H. را بتوان در چند فایل include کرد. در هر حال چیز مهمی نیستند و می‌توانید جایگزین کنید. اگر کد main.c را مطالعه کنید متوجه ساختار آن می‌شوید. دو struct تعریف کردم. یکی Config و یکی ImageState. اولی قرار است محتوای فایل دستورهای ورودی را در خود جای دهد. دومی هم آخرین وضعیت عکس را نگه دارد.

لازم است که تابع ProcessArgs را بگونه‌ای پیاده‌سازی کنید که محتوای فایل ورودی که در argv موجود است را بخواند و ImageState و Config را پر کند. سپس بر اساس اطلاعات Config عکس اول را تولید کرده و ImageState را بروز رسانی کند. سپس دستورهای موجود در Config را یکی-یکی اجرا کند و ImageState را بروز رسانی کند. بطوریکه هر دستور که اجرا می‌شود از آخرین وضعیت دستور قبل شروع کند. به این معنی که اگر دستور قبلی بزرگ‌نمایی به اندازه ۴ بوده و دستور جدید چرخش است. چرخش را روی تصویری که ۴ بار بزرگ‌نمایی شده انجام دهد. برای اینکار کافی است توابع ChangeZoom، ChangeCenter، ChangeRotation و Hold را بدرستی پیاده‌سازی کنید. در نهایت لازم است یک اطلاعاتی در مورد کارهای انجام شده و وضعیت عکس در هر مرحله در audio.txt بنویسید. مثلاً حداقلش این است که بنویسید از فریم شماره چند تا چند چه کاری انجام شده. بعد برای تولید آهنگ مثلاً هنگام چرخش یک صدا تولید کنید، هنگام زوم کردن یک صدا و مثلاً هنگام Hold به صدای دیگر، یا سکوت. از آن بهتر که مثلاً زوم و چرخش با سرعت‌های مختلف انجام شود و صدای هر کدام به تناسب تغییر کند. از آن بهتر که برای مثلاً میزان شلوغی تصاویر یا رنگ‌های غالب در تصویر آهنگ‌های متفاوتی انتخاب کنید.

بستگی به حوصله و علاقه‌تان می‌توانید ابتدا از پیاده‌سازی تابع ProcessArgs شروع کنید. یا اینکه فعلاً در struct بصورت دستی درون کد مقدار دهی شوند که مثلاً چه دستورهایی پشت سر هم اجرا شوند و از پیاده‌سازی UpdateImageData که بر اساس وضعیت state عکس اولیه را در همان state بروز رسانی می‌کند، شروع کنید. پیاده‌سازی تابع WriteBitmapFile هم که موجود است. لذا واقعا مقدار کدی که شما باید بنویسید خیلی نیست. فقط باید با دقت انجام بدهید و قطعاً یک مقداری سعی و خطا و دیباگ کردن هم لازم خواهد داشت.

## ۹ ارسال

کدهای مربوط به پروژه باید در شاخه master و در پوشه FinalProject در آژور تا ۲ بهمن فرستاده شوند. ساختار این پوشه به این شکل است:

- python: کدهای پایتون
- C: کدهای زبان سی
- SonicPi: کدهای تولید موسیقی
- tests
  - zoom: نمونه فایل ورودی که فقط بزرگ‌نمایی می‌کند. به همراه خروجی عکس و audio.txt از برنامه شما با ورودی مشخص شده.
  - rotate: نمونه فایل ورودی که فقط چرخش می‌کند. به همراه خروجی عکس و audio.txt از برنامه شما با ورودی مشخص شده.
  - center: نمونه فایل ورودی که فقط تغییر مرکز می‌دهد. به همراه خروجی عکس و audio.txt از برنامه شما با ورودی مشخص شده.
  - mix: نمونه فایل ورودی که از همه دستورهای بالا فقط یک مورد را دارد. به همراه خروجی عکس و audio.txt از برنامه شما با ورودی مشخص شده.

## ۱۰ ارائه

ارائه پروژه ۵ بهمن ماه ساعت ۱۴ تا ۱۶ خواهد بود. برای ارائه از قابلیت‌هایی که پیاده‌سازی کرده‌اید برای خلق اثر هنری مورد علاقه‌تون و ارائه آن در دانشکده استفاده کنید. ارائه برای همه اعضاء دانشکده خواهد بود و ورود برای عموم آزاد خواهد بود. برای کسانی که بهترین آثار هنری را خلق کنند نمره مثبت قابل توجهی در نظر خواهیم گرفت.

**موفق باشید**