



تکالیف خود را تا تاریخ معین شده به آدرس mostafa.mahdjou@gmail.com ارسال نمایید. دقت کنید که عنوان ایمیل خود را با این الگو منطبق کنید: `tmu_ap_[exercise number]` بطور مثال برای تمرین اول می‌بایست این عنوان را استفاده نمایید: `tmu_ap_1`.

همچنین برای ارسال تکالیف، فقط و فقط فایل `.cpp` مرتبط با هر تمرین را و نه فایل پروژه‌ی ساخته شده در Visual Studio را، در یک پوشه قرار داده و آن را بصورت RAR شده ارسال نمایید. نکته‌ی مهم دیگر در ارتباط با نام فایل‌های `.cpp` ارسال شده است. فایل‌های `.cpp` ارسال شده می‌بایست با الگوی `studentNumber_[exerciseNumber]_[arbitraryFileName].cpp` نامگذاری شوند. منظور از `studentNumber` شماره‌ی اختصاصی‌ای است از ۱ تا ۲۸ که پس از حضور در کلاس حل تمرین به هر فرد اختصاص داده خواهد شد. همچنین منظور از `arbitraryFileName` نام دلخواهی است که برای فایل مرتبط با آن تمرین می‌خواهید در نظر بگیرید. برای مثال برای نامگذاری فایل مرتبط با پاسخ سوال اول، تکالیف سری نخست، با فرض اینکه `studentNumber` اختصاصی فرد ارسال کننده ۵ باشد، بدین صورت خواهد شد: `_5_1_somefilename.cpp`.

لازم به ذکر است، محدودیتهای گذاشته شده در خصوص عناوین ایمیل‌ها و همچنین نام فایل‌ها به جهت انجام مرحله‌ی تست نرم افزاری جهت بررسی کدهای کپی برداری شده می‌باشد. همچنین در هر مرتبه، تمرین تعدادی از دانشجویان بصورت تصادفی، علاوه بر ارسال بصورت ایمیل، به شکل حضوری تحویل گرفته خواهد شد. در صورت ناتوانی در پاسخگویی و یا تشخیص کپی برداری به هر نحو توسط تحویل گیرندگان تمرین، عواقب خوبی در انتظار دانشجوی مربوطه نخواهد بود.

توجه: ارسال تمرین با تأخیر مشمول کسر نمره خواهد شد.



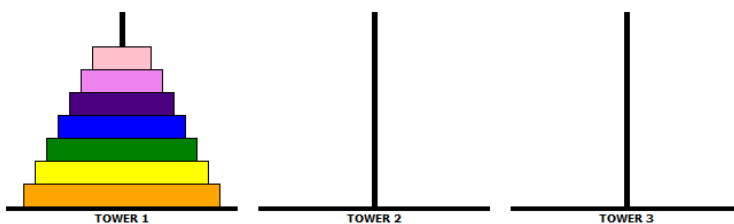
۱. بزرگترین مقسوم علیه مشترک:

برنامه‌ای بنویسید که دو عدد a و b را از ورودی گرفته و بزرگترین مقسوم علیه مشترک آنها را بصورت بازگشتی محاسبه نماید. در این برنامه می‌بایست از این ویژگی کمک بگیرید:

اگر $a > b$ ، آنگاه بزرگترین مقسوم علیه a و b معادل با بزرگترین مقسوم علیه b و باقیمانده‌ی a بر b خواهد بود.

۲. برج هانوی:

فرض کنید سه میله و N دیسک با سایزهای متفاوت داریم. در گام نخست تمامی دیسک‌ها همانند تصویر زیر، به ترتیب از بزرگ به کوچک در یک میله قرار داده شده‌اند.



قصد داریم دیسک‌ها را با حفظ ترتیب اولیه و با رعایت قوانین زیر به یکی از دو ستون دیگر منتقل کنیم.

قوانین:

✓ در هر لحظه فقط حق جابجا کردن یک دیسک را داریم.

✓ مجاز به قرار دادن هیچ دیسکی بر روی دیسک کوچکتر از آن نیستیم.

برنامه‌ی بازگشتی‌ای بنویسید که تعداد دیسک‌ها را از ورودی گرفته و ترتیب جابجایی‌ها را در خروجی چاپ نماید.

۳. کد خاکستری:

آقای ساموئل بکت^۱، نمایشنامه‌نویس، نمایشنامه‌ای را تحت عنوان چهارگانه^۲، نوشته است که دارای ویژگی زیر می‌باشد:

با شروع از صحنه‌ی نمایش خالی، در هر لحظه یکی از شخصیت‌های نمایش وارد صحنه و یا از آن خارج می‌شود. همچنین هر زیر مجموعه از شخصیت‌های نمایش فقط و فقط یکبار در صحنه‌ی نمایش ظاهر می‌شوند. با فرض اینکه این نمایش، همانطور که از نام آن نیز پیداست، چهار شخصیت بیشتر نداشته باشد، ترتیب ورود و خروج شخصیت‌ها به چه ترتیبی خواهد بود؟ اگر تعداد شخصیت‌ها بیشتر از ۴ باشد چطور؟

^۱ Samuel Beckett

^۲ Quad



برنامه‌ی بازگشتی‌ای بنویسید که تعداد شخصیت‌ها را از ورودی گرفته و ترتیب ورود و خروج شخصیت‌ها را در خروجی چاپ کند.

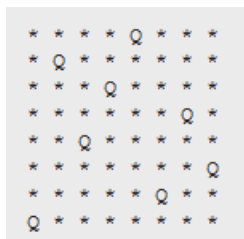
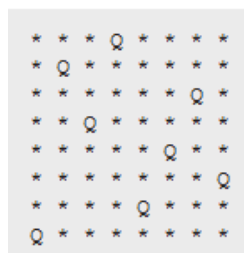
برای حل این سوال می‌بایست از کدهای خاکستری n -بیتی کمک بگیرید که یک لیست با تعداد 2^n سطر و n ستون می‌باشد. هر سطر با سطر قبلی خود فقط در یک بیت متفاوت است. می‌توان شماره‌ی بیتها را معادل با شماره‌ی شخصیت‌ها در نظر گرفته و در صورتی که بیت معادل با یک شخصیت از صفر به یک تغییر کرد، اینطور بیان نمود که شخصیت می‌بایست وارد صحنه شود و در صورتی که این تغییر از یک به صفر باشد، شخصیت می‌بایست از صحنه خارج شود. در تصویر زیر این طرز بیان برای چهار شخصیت نمایش داده شده است:

code	subset	move
0 0 0 0	empty	
0 0 0 1	1	enter 1
0 0 1 1	2 1	enter 2
0 0 1 0	2	exit 1
0 1 1 0	3 2	enter 3
0 1 1 1	3 2 1	enter 1
0 1 0 1	3 1	exit 2
0 1 0 0	3	exit 1
1 1 0 0	4 3	enter 4
1 1 0 1	4 3 1	enter 1
1 1 1 1	4 3 2 1	enter 2
1 1 1 0	4 3 2	exit 1
1 0 1 0	4 2	exit 3
1 0 1 1	4 2 1	enter 1
1 0 0 1	4 1	exit 2
1 0 0 0	4	exit 1

Gray code representations

۴. مسأله‌ی N وزیر:

در این مسأله می‌بایست، N وزیر را در یک صفحه‌ی شطرنجی $N \times N$ طوری قرار دهید که هیچکدام از وزیرها همدیگر را تهدید نکنند. برنامه‌ای بنویسید، که مکان وزیر اول و N را از ورودی خوانده، و تمامی حالت‌های ممکن برای مسأله را مشابه تصویر زیر در خروجی چاپ نماید.



به نام خدا

دانشگاه تربیت معلم - برنامه سازی پیشرفته

#تاریخ تحویل: ۱۰ اسفند ۱۳۹۰ #

تکلیف نخست



۵. مربعهای بازگشتی (نمره‌ی تشویقی +)

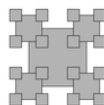
برنامه‌ی بازگشتی‌ای بنویسید عدد N را از ورودی گرفته و الگوهای زیر را در یک Frame تولید کند.



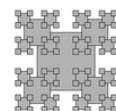
$N = 1$



$N = 2$



$N = 3$



$N = 4$