



تمرین کامپیوتری اول مبانی کامپیوتر و برنامه سازی



اساتید درس: دکتر هاشمی، دکتر مرادی

طراحان: آریا عازم، مهدی حاجی، مجید صادقی نژاد

مقدمه

سلام، در تمرین کامپیوتری اول شما با تعدادی سوال ساده اما کاربردی رو به رو می شوید! در این تمرین شما باید بر اساس فلوچارت و الگوریتمی که در تکلیف دستی خود طراحی کرده اید کد مربوط به هر سوال را بنویسید، همچنین لطفا تمامی نکات تمیزی کد، که در کلاس بیان شده است را رعایت کنید. **در انتهای این فایل می توانید نکات مهم مربوط به تحویل تمرین و همچنین نحوه نمره دهی را مشاهده کنید.**



سوال اول: شمارش آرا

اهداف آموزشی سوال

- ☐ آشنایی با حلقه ها و شرط های کنترلی
- ☐ آشنایی با دریافت ورودی و نمایش خروجی
- ☐ آشنایی با تایپ های مختلف متغیرها و انتخاب تایپ صحیح

شرکتی از شما خواسته تا برنامه ای طراحی کنید که تعداد آرای موافق و مخالف اعضای هیئت مدیره و درصد آن‌ها را محاسبه کند، تعداد غایبان را بشمارد و نتیجه نهایی رای‌گیری را اعلام کند. جزئیات رای اعضا در فرمت زیر به شما داده خواهد شد:

نمونه ورودی
3 8 PNAPPPNA NNNAPPPN AAAPPPPA

در سطر اول دو عدد با یک فاصله به شما داده می‌شود که اول تعداد ردیف صندلی‌ها و دومی تعداد صندلی‌ها در هر ردیف است. خطوط بعدی در هر خط یک ردیف صندلی به شما داده می‌شود که هر حرف انگلیسی بزرگ به معنای صندلی یک عضو است که

A: به معنای غایب بودن آن عضو

P: به معنای رای موافق آن عضو

N: به معنای رای مخالف آن عضو

می‌باشد. توجه کنید که در هنگام دریافت ورودی برنامه شما نباید هیچ کاراکتر یا نوشته‌ای را در محیط ترمینال چاپ کند. نحوه خروجی شما باید به صورت زیر باشد:

نمونه خروجی
Positive: 11 Negative: 6 Absent: 7 P: 64.71 % N: 35.29 % Result: P

در خط اول تعداد آرای موافق، در خط دوم تعداد آرای مخالف، در خط سوم تعداد افراد غایب، در خط چهارم درصد آرای موافق تا دو رقم اعشار، در خط پنجم درصد آرای مخالف تا دو رقم اعشار و در خط ششم در صورتی که نتیجه ی نهایی مثبت بود کاراکتر P، در صورتی که نتیجه ی نهایی منفی بود کاراکتر N و در صورتی که نتیجه نهایی ممتنع بود کاراکتر A چاپ شود. توجه کنید که درصد موافقان و مخالفان از بدون در نظر گرفتن غایبان محاسبه شده و مخرج کسر در این مثال برابر ۱۷ می‌باشد همچنین لازم به ذکر است که کلماتی همچون Positive, Negative, P, N و همچنین دو نقطه و فاصله جلوی آن ها باید در خروجی کد شما به صورت دقیق و منطبق با نمونه خروجی نمایش داده شوند.

نکته ی جدید 📌
برای مشخص کردن تعداد ارقام پس از اعشار در متغیر float و به هنگام استفاده از printf باید از چنین فرمتی استفاده کنید: %0.5f، که در اینجا عدد پنج نشون دهنده ی ۵ رقم پس از اعشاره!!
مثال: <pre>printf("number is: %0.2f \n", number);</pre>

نکته ی جدید 📌
برای نوشتن کاراکتر % در printf کافیه که دوبار کاراکتر % رو کنار هم بنویسید اینجوری در ترمینال یک کاراکتر درصد چاپ میشه و printf این کاراکتر رو با %d یا %c اشتباه نمی گیره برای مثال قطعه کد زیر سه کاراکتر درصد رو با یک فاصله بین هر کدوم چاپ می کنه !!
مثال: <pre>printf("%% %% %%");</pre>



سوال دوم: رشد باکتری‌ها

اهداف آموزشی سوال

- ☐ آشنایی با حلقه ها و شرط های کنترلی
- ☐ آشنایی با دریافت ورودی و نمایش خروجی
- ☐ آشنایی با تایپ های مختلف متغیرها و انتخاب تایپ صحیح

دو باکتری با وزن‌های اولیه a و b در یک ظرف زندگی می‌کنند. هر باکتری یک ضریب رشد مخصوص به خود دارد و در هر ساعت به نسبت آن رشد می‌کند. برای مثال اگر ضریب رشد باکتری با وزن a برابر ۲ باشد، بعد از گذشت یک ساعت، وزن آن $2a$ و بعد دو ساعت وزن آن $4a$ و ... خواهد بود. برنامه‌ای بنویسید که با دریافت وزن دو باکتری و ضریب رشد آن‌ها، اعلام کند که بعد از گذشت چند ساعت وزن باکتری اول اکیدا از وزن باکتری دوم بیشتر خواهد شد. ممکن است در ابتدا، وزن و ضریب باکتری دوم از وزن و ضریب باکتری اول بزرگ‌تر یا مساوی باشد. در این صورت هیچ‌گاه وزن باکتری اول از دومی بزرگ‌تر نخواهد شد. خروجی برنامه برای چنین حالاتی باید ۱- باشد.

نمونه ورودی
4 7 1.7 1.5

نمونه خروجی
5

در خط اول، وزن اولیه باکتری‌ها و در خط دوم ضریب رشد آن‌ها به ترتیب داده می‌شود. خروجی برنامه یک عدد است که بیانگر مدت زمان مورد نیاز برای سنگین‌تر شدن باکتری اول از باکتری دوم است. توجه کنید که اعداد ورودی لزوماً صحیح نمی‌باشند و ممکن است اعشاری باشند.

نمونه ورودی
2 4 3 1.5

نمونه خروجی

2

در نمونه بالا، بعد از گذشت یک ساعت، وزن هر دو باکتری ۶ خواهد شد. بعد از گذشت دو ساعت، وزن باکتری اول ۱۸ و دومی ۹ خواهد بود. پس بعد از ۲ ساعت، وزن باکتری اول از دومی اکیدا بزرگتر خواهد بود.

نمونه ورودی

3 3

7 8

نمونه خروجی

-1



سوال سوم: ماشین حساب ساده!

اهداف آموزشی سوال

- ☐ آشنایی با حلقه ها و شرط های کنترلی
- ☐ آشنایی با دریافت ورودی و نمایش خروجی
- ☐ آشنایی با تایپ های مختلف متغیرها و انتخاب تایپ صحیح

می‌خواهیم یک ماشین حساب ساده طراحی کنیم که قابلیت جمع و ضرب n عدد صحیح را داشته باشد. برنامه‌ای بنویسید که در سطر اول فرمان جمع یا ضرب اعداد و تعداد اعداد را دریافت کند. سپس در n خط بعدی، اعداد را دریافت کند و در آخر حاصل خواسته شده را نمایش دهد. در خط اول علامت '+' به معنای فرمان جمع و '*' به معنای فرمان ضرب است. اگر کاراکتری غیر از این دو دریافت شد، پیغام "Wrong symbol" را نمایش دهید.

نمونه ورودی

+ 4

1

2

3

4

نمونه خروجی
10

در این نمونه جمع چهار عدد ۱ تا ۴ انجام می‌شود و حاصل که ۱۰ است نمایش داده می‌شود.

نمونه ورودی
* 3 2 3 1

نمونه خروجی
6

ضرب سه عدد ۲ و ۳ و ۱ انجام می‌شود و حاصل ۶ است.

نمونه ورودی
- 2

نمونه خروجی
Wrong Symbol

چون کاراکتری به غیر از '+' و '*' وارد شده، پیغام "Wrong Symbol" را نمایش می‌دهیم.

نکات مهم 😊

- از چاپ هر گونه کاراکتر یا متن، غیر از موارد خواسته شده پرهیز کرده و فرمت خروجی و ورودی را به طور دقیق پیاده سازی کنید.
- در صورت وجود هرگونه سوالی، برای ارتباط با طراحان این تمرین می توانید بر روی نام آن ها در ابتدای این فایل کلیک کنید.
- فایل های سوالات تمرین خود را در یک فایل فشرده با فرمت "zip" و با نام "CA1-SID.zip" قرار دهید که SID همان شماره ی دانشجویی شماست. برای مثال اگر شماره ی دانشجویی شما 810103555 باشد باید نام فایل خود را "CA1-810103555.zip" قرار دهید و آن را در قسمت در نظر گرفته شده در صفحه درس در سامانه ایلرن آپلود نمایید.
- فایل zip تمرین شما باید شامل سه فایل C باشد، با نام های:
"q1.c", "q2.c", "q3.c"
- که به ترتیب شامل برنامه های سوال های ۱، ۲، ۳ می باشند.
- برنامه های شما باید با زبان برنامه نویسی C نوشته شود و استفاده از دیگر زبان های برنامه نویسی مجاز نیست.
- **مهلت تحویل تمرین تا ساعت ۲۳:۵۵ روز پنجم آذر است.**
- شما در طول این ترم چهار روز مهلت اضافه برای تمرین کامپیوتری دارید که می توانید آن ها را بین تمرین کامپیوتری به انتخاب خودتان تقسیم کنید در صورتی که از مهلت اضافی خود برای یک تمرین استفاده کنید می توانید آن تمرین را بدون کسر نمره، با تاخیر تحویل دهید. همچنین توجه داشته باشید که تمرین کامپیوتری با بیش از دو روز تاخیر تحویل گرفته نمی شوند.
- تمرین کامپیوتری برای یادگیری برنامه نویسی و مباحث مطرح شده در کلاس طراحی می شوند و انجام آن ها به صورت انفرادی خواهد بود. همچنین در صورت شباهت میان دو پروژه (که به وسیله ی نرم افزارهای مربوطه چک می شود.) برای هر دو نفر نمره صفر در نظر گرفته خواهد شد.

نحوه ی نمره دهی 😊

- ☐ نام گذاری مناسب متغیر ها و توابع در کد (۱۰ نمره)
- ☐ مرتب بودن و تمیزی کد (۲۴ نمره)
- ☐ عدم وجود خطای کامپایل، اخطار (۳۰ نمره)
- ☐ عملکرد صحیح کد در تست ها (۳۶ نمره)

😊 موفق باشید