

تمرین سری پنجم درس برنامه نویسی پیشرفته کامپیوتر

امیر جهانشاهی

۲۶ اردیبهشت ۱۳۹۸

تمرین C++

۱. (آ) یک `vector` به اسم `vec1` و به اندازه ۱۰۰ ایجاد کنید و بدون استفاده از حلقه آن را از ۱ تا ۱۰۰ مقدار دهی کنید. به همین طریق یک `vector` دیگر به اسم `vec2` و به اندازه ۱۰ ایجاد کنید که مقادیر ۱ تا ۱۰ را داشته باشد.
- (ب) اکنون `vec1` را بدون استفاده از حلقه به انتهای `vec2` کپی کنید.
- (ج) یک وکتور به اسم `odd_vec` تعریف کنید و بدون استفاده از حلقه مقادیر فرد `vec1` را در آن کپی کنید.
- (د) یک وکتور به اسم `reverse_vec` تعریف کنید و بدون استفاده از حلقه مقادیر `vec1` را به صورت معکوس (اولین عنصر `vec1` به آخر و برعکس) در آن کپی کنید.
- (ه) یکی از قابلیت‌های اضافه شده در استاندارد ۲۰۱۷، استفاده از پردازش موازی در الگوریتم‌هاست. وکتور `vec2` را در ابتدا به صورت عادی و سپس به صورت موازی `sort` نمایید.
راهنمایی: بدین منظور از `std::execution::par` استفاده کنید.
- نکته:** در هر مرحله بدون استفاده از حلقه خروجی را چاپ کنید.
- راهنمایی:** بدین منظور می‌توانید از `std::copy` استفاده نمایید.

تمرین Python

۲. اکثر توابع در پایتون به این صورت تعریف می‌شوند:

def func(a, b, c, ..., *args, **keywords):

منظور از دو آرگومان آخر چیست؟ با یک مثال نحوه استفاده از آن‌ها را نشان دهید.

۳. کد زیر را در نظر بگیرید.

```
A0 = dict(zip(('a', 'b', 'c', 'd', 'e'), ('1', '2', '3', '4', '5')))
```

```
A1 = range(10)
```

```
A2 = [i for i in A1 if i in A0]
```

```
A3 = sorted(A0[i] for i in A0)
```

```
A4 = [[i, i*i] for i in A1]
```

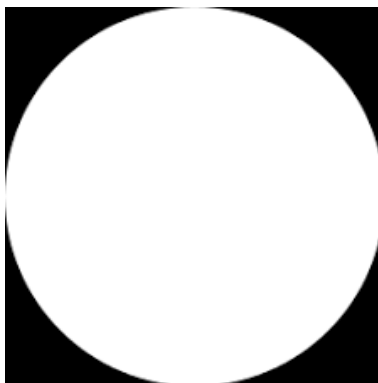
الف) در ابتدا بدون نوشتن کد، خروجی هر یک از عبارات فوق را بنویسید.

ب) کدی بنویسید که در یک حلقه تمام متغیرهای A0 تا A4 را چاپ کند.

۴. می‌خواهیم عدد پی را تخمین بزنیم.

ابتدا یک مربع به ضلع ۱ در نظر بگیرید و یک دایره در آن محاط کنید. به این ترتیب ابتدا یک تابع `IsInCircle(x,y)` تعریف کنید که مشخص کند یک نقطه درون دایره قرار دارد یا خیر. سپس با تولید تعدادی نقاط تصادفی که در داخل مربع قرار دارند، نسبت تعداد نقاط داخل دایره به تعداد کل نقاط را حساب کنید. این نسبت برابر عدد پی است.

بدین ترتیب تابع `Find()` را به گونه‌ای بنویسید که تعیین کند که تعداد نقاط تصادفی باید حداقل چند



تا باشد تا به خطای یک صدم برسیم. (افزودن نقاط تصادفی تا زمانی که به خطای مورد نظر برسیم) سپس یک عدد از ورودی خوانده و به تعداد آن تابع `Find()` را صدا بزنید و میانگین خروجی‌ها را به عنوان نتیجه نهایی تا پنج عدد اعشار چاپ کنید.

۵. می خواهیم برنامه ای بنویسیم که با کمک آن بتوانیم فایل های خود را مدیریت کنیم. قصد داریم برای مدیریت فایل های خود در این برنامه قابلیت های زیر را داشته باشیم:

(آ) ساخت یک پوشه جدید

(ب) ساخت یک فایل جدید

(ج) جستجو کردن بر اساس نام فایل

(د) حذف فایل

در این برنامه، ورودی ها به صورت زیر باید باشند. و هرگاه کاربر دستوری مطابق الگوی ذکر شده وارد کرد، عملیات مذکور انجام شود.

`create_dir(name, address)` یک پوشه جدید با نام `name` در آدرس `address` می سازد. در صورت عدم وجود پوشه باید پوشه ای جدید ساخته شود در غیر این صورت نباید هیچ اتفاقی بیفتد!

`create_file(name, address)` یک فایل جدید با نام `name` در آدرس `address` ایجاد می کند. در صورت عدم وجود فایل باید فایلی جدید ساخته شود در غیر این صورت نباید هیچ اتفاق دیگری بیفتد!

`delete(name, address)` یک فایل با نام `name` در آدرس `address` حذف می کند. در صورت عدم وجود فایل نباید هیچ اتفاقی بیفتد.

`find(name, address)` آدرس تمام فایل های با نام `name` را از آدرس `address` به بعد برمی گرداند. باید یک لیست از تمام آدرس ها برگردانده شود و در صورت عدم وجود فایلی با این نام یک لیست خالی برگردانده شود ترتیب این ادرس ها اهمیتی ندارد.

۶. در این تمرین می خواهیم $\int_a^b f(x)dx$ را با کامپیوتر به روش عددی گاوس حل کنیم. روش گاوس یکی از بهینه ترین روش های حل عددی برای انتگرال های خوش رفتار می باشد. روش کار بدین صورت می باشد که هر انتگرال بین -1 تا 1 را می توانیم توسط فرمول (۱) تقریب بزنیم:

$$\int_{-1}^1 f(x)dx \approx \sum_{i=1}^n w_i f(x_i) \quad (1)$$

توجه کنید که در این فرمول x_i ها صفر های چند جمله ای لژاندر هستند و n مرتبه چند جمله ای لژاندر می باشد. با استفاده از صفر های چند جمله ای لژاندر ضریب های w_i توسط فرمول (۵) قابل محاسبه می باشند. برای محاسبه چند جمله ای های لژاندر و مشتق آنها از به ترتیب از فرمول های (۲) و (۳)

می شود استفاده کرد:

$$P_0(x) = 1, P_1(x) = x, nP_n(x) = (2n-1)xP_{n-1}(x) - (n-1)P_{n-2}(x) \quad (۲)$$

$$P'_n(x) = \frac{n}{x^2-1}(xP_n(x) - P_{n-1}(x)) \quad (۳)$$

برای به دست آوردن صفرهای تابع لژاندر از روش نیوتن-رافسون استفاده کنید. برای حدس اولیه می توانید از فرمول (۴) کمک بگیرید:

$$x_0 = \cos\left(\pi \frac{i - \frac{1}{4}}{n + \frac{1}{2}}\right) \quad (۴)$$

همانطور که گفته شد برای محاسبه w_i ها از فرمول (۵) استفاده می شود:

$$w_i = \frac{2}{(1-x_i^2)[P'_n(x_i)]^2} \quad (۵)$$

راهنمایی: هر انتگرال قابل تغییر متغیر دادن لازم توسط فرمول (۶) به جهت حل توسط روش گاوس می باشد:

$$\int_a^b f(x)dx \approx \frac{b-a}{2} w_i f\left(\frac{b-a}{2}x + \frac{b+a}{2}\right) \quad (۶)$$

• فرض کنید: $f(x) = \frac{x^3}{x+1} \cos(x^2), \quad a=0, \quad b=1$

• برای به دست آوردن چند جمله ای های لژاندر از توابع بازگشتی یا recursive استفاده کنید.

این برنامه با زبان برنامه نویسی C++ به صورت **Object Oriented** نوشته شده و ضمیمه شده است. ورودی این برنامه عدد **n** یعنی درجه ی چند جمله ای لژاندر می باشد. همچنین می توانید جواب خود را با این **لینک** چک کنید.

(آ) این برنامه را به صورت مشابه با استفاده از زبان برنامه نویسی Python بنویسید و زمان اجرا را اندازه بگیرید. دقت کنید که برنامه را حتما به صورت Object Oriented یعنی با استفاده از کلاس بنویسید.

(ب) توسط کتابخانه ی **subprocess** برنامه ی C++ را از طریق برنامه ی Python اجرا کنید و زمان اجرا را اندازه بگیرید. استفاده از دستور subprocess کدی به این صورت دارد. هر چند می توانید در اینترنت هم دستور آن را جستجو کنید.

```
subprocess.call(["IntegrateC++.exe", str(i)])
```

که در آن "IntegrateC++.exe" فایل exe. تولید شده از برنامه ی C++ و pythonstr(i) درجه ی چند جمله ای لژاندر است.

(ج) یک جدول در خروجی چاپ کنید که سطرهای آن درجه چند جمله ای باشد و ستون های آن زمان اجرای کد C++ و Python را نشان بدهد. همچنین این جدول را توسط کتابخانه ی matplotlib رسم کنید. نمودار خروجی که در فایل result.pdf ذخیره شده است را در گزارش خود بیاورید.

۷. به شما یک آرایه از اعداد داده می شود. ما به یک عدد، خاص می گوئیم اگر هم خودش بر ۶ بخش پذیر باشد و هم در حداقل در یکی از جایگاه هایی که در آرایه ظاهر می شود، شماره آن بر ۶ بخش پذیر باشد. توجه کنید که اندیس اولین عضو آرایه از ۱ شروع می شود. حال که فکر می کنید این سوال ساده است، آن را کمی سخت تر می کنیم! باید کد شما حداکثر یک سطر باشد و در آن نمی توانید از: یا exec استفاده کنید.

ورودی

در تنها خط ورودی تعدادی عدد آمده است که آرایه ورودی را نشان می دهد. تعداد اعداد کمتر از ۱۰۰ تاست.

خروجی

در تنها خط خروجی اعداد خاص به ترتیب صعودی چاپ می شوند به صورتی که بین هر دو عدد دقیقاً یک فاصله باشد. توجه کنید که هر عدد خاص دقیقاً یک بار در خروجی ظاهر شود.

ورودی نمونه

3 4 1 37 21 18 23 21 27 22 43 21

خروجی نمونه

18

ورودی نمونه

1 2 3 4 5 6 7 8 9 10 11 12

خروجی نمونه

6 12

جهت تحویل تمرین، هر تمرین را داخل یک فولدر بریزید که با شماره تمرین نام گذاری شده است. Q1, Q2, ... گزارش کار را به صورت PDF در فولدر اصلی تمرین ها قرار دهید. در نتیجه در فولدر اصلی فقط یک فایل گزارش موجود می باشد و تعدادی فولدر که با شماره تمرین ها نام گذاری شده است. اسم فولدر اصلی را به صورت زیر نام گذاری و سپس فشرده سازی و در قالب یک فایل ارسال کنید. توجه نمایید که از قالب فشرده سازی **rar** استفاده نکنید.

پاسخ تمرین های خود را در یکی از سرویس های **github** و یا **gitlab** در یک repository به نام AP-HW5 به صورت Private بارگذاری نمایید. برای این کار باید در قسمت New repository در زمان ساختن repository جدید حالت Private را انتخاب نمایید.

در ادامه تمرینات انجام شده را با فولدر بندی مناسب (سوال ۱ داخل فولدری به همین نام و ...) داخل این پروژه آپلود نمایید. در بخش گزارش فرآیند بارگذاری را شرح دهید و لینک تمرین را داخل گزارش ذکر نمایید.

دقت کنید که با توجه به موارد گفته شده، فایل **gitignore** را به نحوی طراحی کنید که تنها فایل های اصلی و **make file** درون **git** قرار داده شوند.

توجه: به منظور دسترسی به تمرین برای تصحیح، پس از پایان زمان تحویل تمرین پروژه را به حالت Public تغییر دهید.

AP-HW5-شماره دانشجویی.zip

مهلت تحویل: تا ساعت ۲۳، ۵ خرداد ماه ۱۳۹۸