

## سری اول تمرین های درس برنامه نویسی پیشرفته کامپیوتر

امیر جهانشاهی

۲۱ بهمن ۱۳۹۶

۱. هدف این سوال آشنایی با Shell Script می باشد. Shell Script ها فایل هایی می باشند که همانند نمایشنامه های تئاتر می باشند، آن ها به شما می گویند که چه بگویید و چه کاری انجام دهید. این کار توسط کامپیوتر بر روی این فایل ها انجام می شود. Shell Script به ما این اجازه را می دهند که یک مجموعه عمل را بدون آن که لازم باشد دستورات آن را وارد نماییم، توسط کامپیوتر اجرا کنیم. هنگامی که اعمالی استفاده می مداوم یا تکراری دارند، Shell Script ابزار مفیدی می باشد.

یک Shell Script فایلی با پسوند sh. بوده که در terminal لینوکس اجرا می شود. برای ایجاد یک Shell Script کافی است در محیط لینوکس یک فایل متنی با پسوند sh مثلاً (Test.sh) ساخته و دستورات مطلوب را در آن بنویسیم.

برای اجرای این فایل ساخته شده می بایست ابتدا در ترمینال لینوکس با استفاده از دستور cd به آدرس مورد نظر برویم. در مرحله بعد می بایست با استفاده از دستور chmod +x تنها یک بار برای هر فایل، اجازه دسترسی فایل را به گونه ای تغییر دهیم که قابل اجرا شدن باشد. سپس فایل را اجرا می نماییم:

```
۱ sudo chmod +x Test.sh
۲ ./Test.sh
```

همچنین اول هر فایل نیز خط زیر را اضافه می نماییم:

```
۱ #!/bin/sh
```

در این تمرین یک رابط کاربری خیلی ساده خواهیم ساخت. ابتدا از کاربر نام و گذرواژه مورد نظر که APNew می باشد را با پیام های مناسب دریافت می نماییم و تا زمانی که کلمه عبور مناسب دریافت نشده است این پیام تکرار می شود. سپس پس از نمایش پیام موفقیت از کاربر درخواست می شود که عدد ۱ یا ۲ را وارد نماید. در صورت ورود عدد ۱، یک پوشه به نام AP2018 ساخته می شود و در آن فایل Hello.cpp ایجاد شود و سپس ترمینال بسته شود. در صورت ورود عدد ۲، ترمینال بسته می شود.

راهنمایی و نکات :

- Shell Script یک محیط کدنویسی کامل را در اختیار شما می گذارد، به طور مثال قابلیت تعریف متغیر، گذاره های شرطی، حلقه و غیره.

- تعریف متغیر به صورت زیر می باشد:

```
۱ foo='Test'
۲ echo \$foo
```

دقت داشته باشید که در تعریف متغیر از space استفاده نشده است. از دستور echo برای چاپ هرگونه پیام یا متغیر در صفحه استفاده می شود. همچنین برای استفاده از متغیر ها نیز از یک \$ قبل از نام متغیر استفاده می شود.

- برای دریافت اطلاعات از کاربر از دستور read استفاده می شود و برای اختصاص یک پیام به آن (مثلا Username) از مشخصه ی -p استفاده می شود. مثال:

```
۱ read -p "Username: " varname
```

```

marshalamín@marshalamín: ~/Desktop/Bash
File Edit View Search Terminal Help
marshalamín@marshalamín:~/Desktop/Bash$ ./bash.sh
      *
    * *
  *   *
*     *
*****
*     *
*   *
* *
*
*   *
*
*   *
*
Welcome Everyone to AP2018!
Username:

```

شکل ۱: نمونه ای از یک Shell Script

۲. هدف در این سوال پیدا کردن ریشه های یک معادله به روش **وتری (secant)** می باشد. همانطوری که می دانید ریشه ها از دنباله تکراری با رابطه

$$x_{n+1} = x_n - \frac{f(x_n)(x_n - x_{n-1})}{f(x_n) - f(x_{n-1})} = \frac{x_{n-1}f(x_n) - x_nf(x_{n-1})}{f(x_n) - f(x_{n-1})}$$

محاسبه می شود. بنابراین باید دو مقدار اولیه (مانند ۱ و ۰) به متغیر  $x$  بدهید و سپس تا زمانی که خطای آن ( $\epsilon$ ) از حدی کمتر بشود، مقدار  $x$  را از رابطه ذکر شده به روز رسانی کنید.

$$\epsilon = \left| \frac{x^{new} - x^{old}}{x^{old}} \right|$$

با استفاده از برنامه خود ریشه معادله  $x^3 + x^2 + x + 1$  را با دقت  $\epsilon < 1 \times 10^{-4}$  حساب کنید. توجه نمایید که جواب صحیح ۱- می باشد. در صورت درست کارکردن برنامه، حال معادله  $\coth(x) - \log(x)$  را حل کنید.

**راهنمایی:** برای آشنایی با توابع خواسته شده بخش Mathematical Functions از فصل دوم کتاب جداول 2.8 و 2.9 را مطالعه کنید.

در صفحه خروجی بایستی شماره تکرار، مقدار قدیم، مقدار جدید و همچنین مقدار تابع به همراه خطا در هر تکرار به صورت مرتب نشان داده شود.

**راهنمایی:** برای نمایش مرتب بخش Formatting Stream Output از فصل دوم کتاب را مطالعه کنید.

۳. برنامه ای بنویسید که عدد  $n$  و سپس یک دنباله  $n$ -تایی  $a_1, a_2, a_3, \dots, a_n$  را از ورودی بخواند و سپس مقدار زیر را چاپ کند:

$$\sum_{1 \leq l \leq r \leq n} f(l, r)$$

که  $f(l, r)$  را این گونه تعریف می کنیم:

$$f(l, r) = \sum_{i=l}^r a_i$$

در سطر اول ورودی یک عدد  $n$  آمده است و در سطر دوم  $n$  عدد طبیعی آمده است که عدد  $i$ -ام نمایانگر  $a_i$  است. مثال:  
ورودی:

3

3 2 1

خروجی: 20

$$f(1, 1) = 1, f(1, 2) = 3, f(1, 3) = 6, f(2, 2) = 2, f(2, 3) = 5, f(3, 3) = 3$$

$$\text{ans} = 1 + 3 + 6 + 2 + 5 + 3 = 20$$

جهت تحویل تمارین، هر تمرین را داخل یک فولدر بریزید که با شماره تمرین نام گذاری شده است. Q1, Q2, ... و هر تمرین MakeFile مخصوص به خود را دارد. 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.

AP-HW1-شماره دانشجویی.zip

مهلت تحویل: تا ساعت ۲۳ یکشنبه ۲۹ بهمن ماه ۱۳۹۶

برای عضویت در روایات تلگرام برنامه نویسی پیشرفته اینجا را کلیک کنید.

برای عضویت در کانال تلگرام برنامه نویسی پیشرفته اینجا را کلیک کنید.