

7

گزارش سری اول تمرین برنامه نویسی
زمستان ۹۶

نام و نام خانوادگی:

محمد رضی

شماره دانشجویی:

۹۴۲۳۰۵۲

تعداد سوالات: ۳

تاریخ تألیف: ۲۵ بهمن ۹۶

فهرست گزارش

سوال ۱	۳
کد نهایی سوال:	۳
توضیحات کد:	۴
اسکرین شات از عمل کردن کد	۶
سوال دوم	۸
کد نهایی سوال:	۸
توضیحات کد:	۱۰
اسکرین شات از عمل کردن کد	۱۲
سوال سوم	۱۴
کد نهایی سوال	۱۴
توضیحات کد	۱۵
اسکرین شات از عمل کردن کد	۱۷

کد نهایی سوال:

صفحه ۲

توضیحات کد:

در خط اول عبارت `bin/sh/#!` که با علامت `!` شروع میشود بیان میکند با چه چیزی اجرا شود.

در خطوط بعدی کد پس از نمایش لوگوی نشان داده شده، `username` و `password` را از طرف مقابل دریافت میکند.

- در این سوال username و password صحیح عبارت APNew فرض شده است.

echo '

```

AAA
A:::A
A::::A
A:::::A
A:::::A
A:::::A:::A
A:::::A A:::::A
A:::::A A:::::A
A:::::A A:::::A
A:::::AAAAAAAAA:::A
A:::::AAAAAAAAA:::A
A:::::AAAAAAAAAAAAA:::A
A:::::A A:::::A
A:::::A A:::::A
A:::::A A:::::A
AAAAAAAA
AAAAAAAAAPPPPPPPPP

```

By @MohammadRaziei

```
read -p "Please Enter Your UserName : " username
read -p "Please Enter Your Password : " password
```

در قسمت بعد مادامی که عبارت صحیح را وارد نکنیم توسط while زیر بطور متوالی از ما میخواد که عبارات صحیح را وارد کنیم.

```
while [ "$username" != "APNew" -o "$password" != "APNew" ]
do
    echo 'Sorry! Check your information and retry.'
    read -p "Please Enter Your UserName : " username
    read -p "Please Enter Your PassWord : " password
    echo
done
```

- در این قسمت نسبت به خالی گذاشتن جواب نیز حساس است و برخورد مشابهی انجام میدهد

پس از ورود عبارتی راهنما چاپ میشود و از ما میخواهد که یکی از دو گزینه ی معرفی شده را انتخاب کنیم.

```
echo 'Signing successful'
```

NOTICE:

```
1 >> to create '$PWD'/AP2018/Hello.cpp" and then exit
2 >> to exit now
'
```

read -p 'Enter 1 or 2 : ' key

سپس توسط حلقه ای دیگر دوباره چک میشود که عدد وارد شده ی ما صحیح می باشد یا خیر.

```
while [ "$key" != "1" -a "$key" != "2" ]
do
    echo 'Invalid Key'
    read -p 'Try to enter 1 or 2 : ' key
done
```

- این قسمت نیز نسبت به خالی گذاشتن عبارت خواسته شده حساس است و رفتار مشابهی نشان میدهد.

در قسمت بعد با یک شرط (پس از مطمئن شدن از محتوای خواسته شده) اعمال خواسته شده را انجام میدهد.

```
if [ $key = 1 ];then
    mkdir AP2018
    touch AP2018/Hello.cpp
fi
exit
```

عبارت exit خارج از شرط آمده است زیرا در هر حال در هر کدام از حالات باید انجام شود.

و کد بدین صورت، درست عمل خواهد کرد.

- برای اجرا این فایل اگر در directory آن فایل هستیم، عبارت ./Test.sh را وارد میکنیم.

```

#./bin/sh
#####
MohammadRaziei
#####
echo "
          AAA                PPPPPPPPPPPPPPP
        A::A              P::::::::::::P
       A::::A            P::::::PPPPPP::P
      A:::::A           PP::::P    P::::P
     A:::::A          P::::P    P::::P
    A:::::A:A::::A   P::::P    P::::P
   A:::::A A:::::A  P::::PPPPPP::P
  A:::::A   A:::::A P::::::::::::PP
 A:::::A   A:::::A P::::PPPPPPPP
A:::::AAAAAAA:::A  P::::P
A:::::AAAAAAAA:::A P::::P
A:::::AAAAAAAAAA:::A P::::P
A:::::A          A:::::A PP:::::PP
A:::::A          A:::::A P:::::P
A:::::A          A:::::A P:::::P
AAAAAA          AAAAAAPPPPPPPP

By @MohammadRaziei

read -p "Please Enter Your UserName : " username
read -p "Please Enter Your PassWord : " password

while [ "$username" != "APNew" -o "$password" != "APNew" ]
do
    echo 'Sorry! Check your information and retry.'
    read -p "Please Enter Your UserName : " username
    read -p "Please Enter Your PassWord : " password
    echo
done
echo 'Signing succesfull'

NOTICE:
l >> to create "'$PWD'/AP2018/Hello.cpp" and then exit
"Q1/Test.sh" 63L, 2682C

```

```
HW01 Q1/Test.sh

      AAA                      PPPPPPPPPPPPPPPPP
      A:::A                    P:::::::::::::P
      A::::A                    P:::::PPPPPP:::::P
      A:::::A                    PP:::::P    P:::::P
      A:::::A                    P:::P      P:::::P
      A:::::A A:::::A            P:::P      P:::::P
      A:::::A A:::::A            P:::PPPPPP:::::P
      A:::::A A:::::A            P:::::::::::::PP
      A:::::A A:::::A            P:::PPPPPPPPPP
      A:::::A A:::::A            P:::P
      A:::::A A:::::A            P:::P
      A:::::A A:::::A            P:::P
      A:::::A A:::::A            PP:::::PP
      A:::::A A:::::A            P:::::P
      A:::::A A:::::A            P:::::P
      AAAAAA                      AAAAAAPPPPPPPPPPP

By @MohammadRaziei

Please Enter Your UserName :      Mohammad Raziei
Please Enter Your PassWord :      9423052
Sorry! Check your information and retry.
Please Enter Your UserName :
Please Enter Your PassWord :

Sorry! Check your information and retry.
Please Enter Your UserName :      APNew
Please Enter Your PassWord :      APNew

Signing succesfull

NOTICE:
1 >> to create "/mnt/d/books/academic books/term 6/Advance Programming/tests/HWs_Attempt/HW01/AP2018/Hello.cpp" and then exit
2 >> to exit now

Enter 1 or 2 :      5
Invalid Key
Try to enter 1 or 2 :
Invalid Key
Try to enter 1 or 2 :      1_
```

```
Q1 tree
├── AP2018
│   ├── Hello.cpp
│   └── Test.sh
└── 1 directory, 2 files
```

سوال دوم

کد نهایی سوال:

```
Q2/main.cpp
#include <iostream>
#include <cmath>
#include <iomanip>

double scant( double (*func)(double), double, double );
double poly(double);
double fun(double);
void choose(bool);

int main()
{
    bool a{};
    std::cout << "please enter 0 or 1...\n 0 : for 'coth(x)-log(x)'\n 1 : for 'x^3 + x^2 + x + 1' \n\n";
    std::cin >> a;
    choose(a) ;
    return 0;
}

double scant( double (*func)(double), double old, double older)
{
    static size_t CNT{};
    double New{};
    if ( std::isinf(func(old)) )
        New = older;
    else if ( std::isinf(func(older)) )
        New = old;
    else
        New = (older*func(old) - old*func(older))/(func(old) - func(older));

    double err{(New - old)* old > 0 ? (New - old)/old : (old - New)/old };

    std::cout << CNT++ << "\t:" << std::fixed << std::setprecision( 8 )
        << std::setw(20) << older << std::fixed << std::setprecision( 8 )
        << std::setw(20) << old << std::fixed << std::setprecision( 8 )
        << std::setw(20) << New << std::fixed << std::setprecision( 8 )
        << std::setw(20) << err << std::endl;
    std::cout.unsetf( std::ios::fixed );
    if( err > 0.0001 || std::isinf(err) )
    {
        New = scant(func,New, old);
    }
    return New;
}

double poly(double x)
{
    return pow(x, 3) + pow(x, 2) + x + 1;
}

double fun(double x)
{
    double exp2x{exp(2*x)};
    return (exp2x + 1)/(exp2x - 1) - log(x);
}

void choose(bool x)
{
    double (*pf)(double){x?poly:fun};
```



```

std::cout << std::endl << (x?"For 'x^3 + x^2 + x + 1' ":"For 'coth(x)-log(x)' ")
<< std::endl;
std::cout << "stage" << "\t:"<< std::fixed << std::setprecision( 8 )
    << std::setw(20) << "Older num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "Old num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "New num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "error value" << std::endl;
std::cout.unsetf( std::ios::fixed );
double root{scant( pf, 0, 1)};
std::cout << std::endl << "#####\n"
    << "Done :)\n Root is  " << root << std::endl;
}

```

Q2/Makefile

```

CXX = g++
CXXFLAGS = -std=c++17 -Wall -c
LXXFLAGS = -std=c++17
OBJECTS = main.o
TARGET = main

$(TARGET): $(OBJECTS)
    $(CXX) $(LXXFLAGS) $(OBJECTS) -o $(TARGET)
main.o: main.cpp
    $(CXX) $(CXXFLAGS) main.cpp
clean:
    rm -f $(TARGET) $(OBJECTS)

```

توضیحات کد:

در این سوال تابع زیر را می‌خواهیم بسازیم.

```
double scant( double (*func)(double), double, double );
```

باتوجه به رابطه ای که داریم:

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

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

تابع را بصورت زیر می‌سازیم:

```
double scant( double (*func)(double), double old, double older)
{
    static size_t CNT{};
    double New{};
    if ( std::isinf(func(old)) )
        New = older;
    else if ( std::isinf(func(older)) )
        New = old;
    else
        New = (older*func(old) - old*func(older))/(func(old) - func(older));

    double err{(New - old)* old > 0 ? (New - old)/old : (old - New)/old };

    std::cout << CNT++ << "\t:" << std::fixed << std::setprecision( 8 )
        << std::setw(20) << older << std::fixed << std::setprecision( 8 )
        << std::setw(20) << old << std::fixed << std::setprecision( 8 )
        << std::setw(20) << New << std::fixed << std::setprecision( 8 )
        << std::setw(20) << err << std::endl;
    std::cout.unsetf( std::ios::fixed );
    if( err > 0.0001 || std::isinf(err) )
    {
        New = scant(func,New, old);
    }
    return New;
}
```

قسمت ۱ : برای شمردن مراحل یک متغیر استاتیک تعریف میکنیم.

قسمت ۲ : با توجه به رابطه ی داده شده متغیر New را از روی دو متغیر old, older می‌سازیم.

این قسمت به صورت شرطی بیان شده است. زیرا اگر هریک از func(old), func(older) مقدار بینهایت شود. مقدار New برابر NaN میشود. پس بجای آن عبارت، حد آنها را در بینهایت گذاشته شده است.

پس از آن مقدار `err` را با توجه به رابطه ی قدر مطلقى وابسته به `old,new` تعريف شده است.

قسمت ۳ : این قسمت برای نمایش مراحل انجام داده شده می باشد.

قسمت ۴ : این قسمت شرط پایان یافتن تابع بازگشتی مان چک میشود.

توجه داریم که در این حالت یک شرط اضافه تر نیز اضافه شده است.

```
if( err > 0.0001 || std::isinf(err) )
```

این شرط بخاطر این است که گرچه مقدار `err` مثبت است ولی به دلایل حافظه ای در هنگام `inf` شدن مقدار آن میتواند مقدار منفی `-inf` را نیز بپذیرد. که به اشتباه از حلقه شرطی خارج میشود.

مقدار خروجی این تابع همان ریشه ی تابع است.

تابع بعدی که به صورت زیر است از `main` از ما خواسته میشود که انتخاب کنیم ریشه ی کدام معادله را حل کند و پس انتخاب این تابع اجرا میشود و جواب را به شکل شایسته ای نمایش میدهد.

```
void choose(bool x)
{
    double (*pf)(double){x?poly:fun};
    std::cout << std::endl << (x?"For 'x^3 + x^2 + x + 1' ":"For 'coth(x)-log(x)' :") <<
    std::endl;
    std::cout << "stage" << "\t:"<< std::fixed << std::setprecision( 8 )
    << std::setw(20) << "Older num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "Old num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "New num. " << std::fixed << std::setprecision( 8 )
    << std::setw(20) << "error value" << std::endl;
    std::cout.unsetf( std::ios::fixed );
    double root{scant( pf, 0, 1)};
    std::cout << std::endl << "#####\n"
    << "Done :)\n Root is  " << root << std::endl;
}
```

این کد به درستی عمل میکند.

نکته ی دیگری که لازم است به آن توجه کنیم این است که چون تابع چند جمله ای داده شده با این الگوریتم و مقادیر اولیه آن به ریشه ی آن همگرا میشود. ولی در مورد دوم با توجه به الگوریتم و مقدار اولیه آن و همچنین تابع آن که در صفر، مقدار آن نامحدود میباشد؛ این حلقه در نقطه ای غیر از ریشه قفل میشود و این مشکل کد نیست بلکه خاصیت مسیله اینطور ایجاب میکند.

```

#include <iostream>
#include <cmath>
#include <iomanip>

double scant( double (*func)(double), double, double );
double poly(double);
double fun(double);
void choose(bool);

int main()
{
    bool a{};
    std::cout << "please enter 0 or 1...\n 0 : for 'coth(x)-log(x)'\n 1 : for 'x^3 + x^2 + x + 1' \n\n";
    std::cin >> a;
    choose(a) ;
    return 0;
}

double scant( double (*func)(double), double old, double older)
{
    static size_t CNT{};
    double New{};
    if ( std::isinf(func(old)) )
        New = older;
    else if ( std::isinf(func(older)) )
        New = old;
    else
        New = (older*func(old) - old*func(older))/(func(old) - func(older));

    double err{(New - old)* old > 0 ? (New - old)/old : (old - New)/old };

    std::cout << CNT++ << "\t:" << std::fixed << std::setprecision( 8 )
        << std::setw(20) << older << std::fixed << std::setprecision( 8 )
        << std::setw(20) << old << std::fixed << std::setprecision( 8 )
        << std::setw(20) << New << std::fixed << std::setprecision( 8 )
        << std::setw(20) << err << std::endl;
    std::cout.unsetf( std::ios::fixed );
    if( err > 0.0001 || std::isinf(err) )
    {
        New = scant(func,New, old);
    }
    return New;
}

double poly(double x)
{
    return pow(x, 3) + pow(x, 2) + x + 1;
}

double fun(double x)

```

"main.cpp" 67L, 2080C

```

please enter 0 or 1...
0 : for 'coth(x)-log(x)'
1 : for 'x^3 + x^2 + x + 1'

1

For 'x^3 + x^2 + x + 1' :
stage :      Older num.      Old num.      New num.      error value
0      :      1.00000000      0.00000000      -0.33333333      inf
1      :      0.00000000      -0.33333333      -1.28571429      2.85714286
2      :      -0.33333333      -1.28571429      -0.80403458      0.37463977
3      :      -1.28571429      -0.80403458      -0.94784798      0.17886469
4      :      -0.80403458      -0.94784798      -1.01151318      0.06716815
5      :      -0.94784798      -1.01151318      -0.99938757      0.01198760
6      :      -1.01151318      -0.99938757      -0.99999299      0.00060579
7      :      -0.99938757      -0.99999299      -1.00000000      0.00000702

#####
Done :)
Root is -1

Press ENTER or type command to continue

```

```

please enter 0 or 1...
0 : for 'coth(x)-log(x)'
1 : for 'x^3 + x^2 + x + 1'

0

For 'coth(x)-log(x)' :
stage :      Older num.      Old num.      New num.      error value
0      :      1.00000000      0.00000000      1.00000000      -inf
1      :      0.00000000      1.00000000      1.00000000      0.00000000

#####
Done :)
Root is 1

Press ENTER or type command to continue

```

سوال سوم

کد نهایی سوال

Q3/main.cpp

```
#include <iostream>

double flr(double*, int l ,int r);
double print_calc(double* ,int);
double print(double*, int l, int r);
int main()
{
    size_t n{};
    std::cout << "Please enter n : \t";
    std::cin >> n;
    std::cout << "Now enter n numbers: " << std::endl;
    double* arr{new double [n]};
    for(size_t i{}; i < n ;i++)
        std::cin >> arr[i];

    print_calc(arr,n);

    delete[] arr;
    return 0;
}

double flr(double* arr, int l, int r)
{
    double sum{};
    for(int i{l-1}; i < r ; i++)
        sum += arr[i];
    return sum;
}

double print_calc(double* arr, int n)
{
    double ans{};
    for(int r{1}; r <= n; r++) // for 1<=r<=n
        for(int l{1}; l <= r; l++) // for 1<=l<=r
        {
            ans += print(arr,l,r);
            std::cout << ", ";
        }
    std::cout << "\b\b . \n\nans = " << ans << std::endl;
    return ans;
}

double print(double* arr,int l, int r)
{
    std::cout << "f(" << l << ", " << r << ") = " << flr(arr,l,r) ;
    return flr(arr,l,r);
}
```

Q3/Makefile

```
CXX = g++
CXXFLAGS = -std=c++17 -Wall -c
LXXFLAGS = -std=c++17
OBJECTS = main.o
TARGET = main

$(TARGET): $(OBJECTS)
    $(CXX) $(LXXFLAGS) $(OBJECTS) -o $(TARGET)
```

```
main.o: main.cpp
$(CXX) $(CXXFLAGS) main.cpp
clean:
rm -f $(TARGET) $(OBJECTS)
```

توضیحات کد

در این نیز مانند سوال قبل باید تابعی را باتوجه به روابط داده شده بسازیم.

```
double print_calc(double* arr, int n)
```

این روابط به صورت زیر بیان شده است.

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

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

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

در این حالت ابتدا یک تابع $flr(,)$ را بصورت زیر تعریف میکنیم.

```
double flr(double* arr, int l, int r)
{
    double sum{};
    for(int i{l-1}; i < r ; i++)
        sum += arr[i];
    return sum;
}
```

این تابع دقیقاً متناظر با همان رابطه است.

طول آرایه و آرایه ی نظیر آن در تابع main از کاربر گرفته میشود. چون طول آرایه مشخص نیست، آرایه بصورت داینامیک ایجاد میکنیم و نهایت آنرا حذف میکنیم. در این بین با تابع معرفی شده نخست آن را پاسخ را محاسبه و نمایش می دهیم.

```
size_t n{};
std::cout << "Please enter n : \t";
std::cin >> n;
std::cout << "Now enter n numbers: " << std::endl;
double* arr{new double [n]};
for(size_t i{}; i < n ;i++)
    std::cin >> arr[i];

print_calc(arr,n);

delete[] arr;
return 0;
```

پس تابع اول به صورت زیر خواهد بود.

```
double print_calc(double* arr, int n)
{
    double ans{};
    for(int r{1}; r <= n; r++) // for 1<=r<=n
        for(int l{1}; l <= r; l++) // for 1<=l<=r
        {
            ans += print(arr,l,r);
            std::cout << ", ";
        }
    std::cout << "\b\b . \n\nans = " << ans << std::endl;
    return ans;
}
double print(double* arr,int l, int r)
{
    std::cout << "f(" << l << ", " << r << ") = " << flr(arr,l,r) ;
    return flr(arr,l,r);
}
```

تابع زیرین آن تابعیست کمکی جهت محاسبه و نمایش پاسخ برای خوانایی بیشتر کد.

در این تابع با توجه به رابطه ی اول سوال نوشته شده است که تناظر یک به یک آن نیاز به توضیح آن را مرتفع میکند.

این کد به درستی عمل میکند.


```

#include <iostream>

double flr(double*, int l, int r);
double print_calc(double*, int);
double print(double*, int l, int r);
int main()
{
    size_t n{};
    std::cout << "Please enter n : \t";
    std::cin >> n;
    std::cout << "Now enter n numbers: " << std::endl;
    double* arr{new double [n]};
    for(size_t i{}; i < n ;i++)
        std::cin >> arr[i];

    print_calc(arr,n);

    delete[] arr;
    return 0;
}

double flr(double* arr, int l, int r)
{
    double sum{};
    for(int i{l-1}; i < r ; i++)
        sum += arr[i];
    return sum;
}

double print_calc(double* arr, int n)
{
    double ans{};
    for(int r{1}; r <= n; r++)           // for l<=r<=n
        for(int l{1}; l <= r; l++)       // for l<=l<=r
        {
            ans += print(arr,l,r);
            std::cout << ", ";
        }
    std::cout << "\b\b . \n\nans = " << ans << std::endl;
    return ans;
}

double print(double* arr,int l, int r)
{
    std::cout << "f(" << l << ", " << r << ") = " << flr(arr,l,r) ;
    return flr(arr,l,r);
}

```

"main.cpp" 47L, 910C 1,1 All

```

Please enter n :      3
Now enter n numbers:
1 2 3
f(1,1) = 1, f(1,2) = 3, f(2,2) = 2, f(1,3) = 6, f(2,3) = 5, f(3,3) = 3 .

ans = 20

Press ENTER or type command to continue

```