



اساتید:  
دکترمرادی، دکتر هاشمی

عنوان:  
توابع

نیمسال اول  
1402-03

## ۱- انجام دهید!



در جلسات گذشته، با برنامه‌ای که عدد  $n$  را از ورودی گرفته و سپس حاصل  $n!$  را در خروجی نمایش می‌داد آشنا شدید. یک نمونه از این برنامه در زیر داده شده است:

```
#include <stdio.h>
int main(){
    int n , result =1;
    scanf("%d",&n);
    for (int i=1; i<=n;i++){
        result=result*i;
    }
    printf("%d\n", result);
    return 0;
}
```

حال می‌خواهیم این برنامه را به صورت یک تابع به نام `fact` در بیاوریم. این تابع عدد صحیح  $n$  را به عنوان آرگومان<sup>۱</sup> گرفته، سپس حاصل  $n!$  را حساب کرده و به عنوان خروجی باز می‌گرداند. برای این کار، در قطعه کد زیر، جاهای خالی را تکمیل نمایید.

```
#include <stdio.h>
... fact (... ){
    //...
    return ...;
}
int main(){
    int n;
    scanf("%d",&n);
    printf("%d\n", ... );
    return 0;
}
```

قسمت ۱: نتیجه را به دستیاران آموزشی نشان دهید.



حال اگر بخواهیم از عبارت `printf` در `main` استفاده نکنیم، چه تغییراتی را باید در برنامه قسمت قبل اعمال کنیم؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۲)

## ۲- فکر کنید!



<sup>1</sup> Argument

برنامه زیر برای بررسی زوج یا فرد بودن یک عدد که از ورودی گرفته می‌شود، نوشته شده است. این برنامه را در یک پروژه جدید نوشته و درمورد عملکرد آن فکر کنید. سپس به سوالات زیر در مورد آن پاسخ دهید.

```
#include <stdio.h>
```

```
int function (int n){
    if(n%2 == 0){
        return 1;
    }
    printf("ODD (FARD)!\n");
}
int main(){
    int a;
    scanf("%d",&a);
    if (function(a)){
        printf("EVEN (ZOJ)!\n");
    }
    return 0;
}
```

همانطور که می‌بینید، در تابع `function` متغیر `n` به عنوان آرگومان تابع در نظر گرفته شده است. ولی در `main` متغیر `a` به عنوان ورودی به تابع `function` داده شده است. با وجود این مغایرت، چرا هیچ خطایی دریافت نمی‌کنیم؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۳)

به شرط `if` دقت کنید. چرا قرار دادن نام تابع به عنوان شرط در `if`، منعی ندارد؟ به عبارت دیگر، کامپایلر، چگونه با نام تابع در `main` رفتار می‌کند؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۴)

اگر عددی زوج را به عنوان ورودی وارد کنیم، چرا عبارت داخل `if` موجود در `main` اجرا می‌شود؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۵)

با وارد کردن عددی زوج به عنوان ورودی، چرا عبارت `printf` موجود در `function`، اجرا نمی‌شود؟ با وجود اینکه بعد از `if` هیچ `else` یا چیز دیگری وجود ندارد!!! علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۶)

۳- انجام دهید!



در این قسمت می‌خواهیم برنامه‌ای بنویسیم که در آن، میانگین (وزن دار) نمرات یک دانشجو توسط یک تابع محاسبه و اعلام گردد. با این فرض که این دانشجو یک درس ۳ واحدی ریاضی، یک درس ۲ واحدی فیزیک و یک آزمایشگاه ۱ واحدی دارد. حال یک پروژه جدید ایجاد کرده و کد زیر را کامپایل نمایید.

```
#include <stdio.h>
int function(float cal, float phy, float lab){
    float sum= (cal*3)+(phy*2)+(lab*1);
    float average = sum/6;
    return average;
}
int main(){
    float cal=18 , phy=17 , lab=19;
    printf("average= %f\n",function(cal , phy ));
    return 0;
}
```

با چه خطایی مواجه شدید؟ آن را اصلاح کرده و علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۷)

حال این برنامه را دوباره اجرا کنید. آیا نتیجه اعلام شده صحیح است؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۸)

حال ترتیب ورودی‌های تابع را در printf تغییر دهید. آیا نتیجه تغییر می‌کند؟ چرا؟ علت را برای دستیاران آموزشی توضیح دهید. (قسمت ۹)

4- انجام دهید!



قسمت ۱۰: برنامه زیر را اجرا کنید و سپس مقدار چاپ شده را توجیه کنید.

```
#include <stdio.h>
int g(int y){
    int j = 2;
    y = j * 3;
    return y;
}
int f(int x){
    return g(x);
}
int main() {
```

```
int a = 10;
printf("%d", f(a));
return 0;}
```

## ۵- انجام دهید!



یک فایل جدید ایجاد کرده و برنامه زیر را در آن بنویسید و سعی کنید آن را کامپایل نمایید.

```
#include <stdio.h>
int main() {
    int z = 4;
    if (z - 4) {
        int i = 1;
        z = z + i;
        printf("i=%d , z=%d\n", i,z);
    }else {
        int i = 0;
        z = i + 1;
        {
            int i = 1;
            z = i + 1;
            printf("i=%d , z=%d\n", i,z);
        }
        printf("i=%d , z=%d\n", i,z);
    }
    return 0;}
```

قسمت ۱۱: مقادیر نمایش داده شده را توجیه نمایید.



## ۶- انجام دهید!



اکنون اظهار<sup>۲</sup> تابع fact را که در قسمت اول این آزمایش آنرا پیاده سازی کردید، در یک header file به نام fact.h قرار دهید و بدنه‌ی آن را درون یک فایل c با نام fact.c قرار دهید. دقت کنید که fact.h را در ابتدای fact.c ، include کرده باشید. محتوای دو فایل fact.h و fact.c در زیر آمده است:

```
fact.h
int fact(int x);
```

```
fact.c
#include "fact.h"
int fact(int x) {
    /* Your Code Goes here. */
}
```

حال با توضیحات بالا، قسمت اول را دوباره انجام دهید.

<sup>2</sup> Declaration

```
main.c
#include "fact.h"
int main() {
    /* Your Code Goes here. */
}
```

✓ قسمت ۱۲: نتیجه را به دستیاران آموزشی نشان دهید.

## ۷- انجام دهید! (توابع بازگشتی<sup>۳</sup>) (امتیازی)

در این قسمت قرار است برنامه‌ای بنویسیم که عددی طبیعی مانند  $n$  را از ورودی خوانده و با استفاده از تابعی بازگشتی، حاصل جمع ارقام آن را در خروجی نمایش دهد. به برنامه زیر توجه کنید:

```
#include<stdio.h>
int sum=0; //global variable
int DigitSum(int n){
    if(n!=0){
        sum += ...;
        DigitSum(...);
    }
    return sum;
}
int main(){
    int n;
    scanf("%d", &n);
    printf("digits sum = %d\n", DigitSum(n));
    return 0;
}
```

✓ جاهای خالی را تکمیل نموده و روند عملکرد این برنامه را برای دستیاران آموزشی توضیح دهید. (قسمت ۱۳)

✚ حال به قطعه کد زیر نگاه کنید. به نظر شما بعد از اجرای برنامه چه اتفاقی می‌افتد؟

```
long fib(long n){
    if (n <= 2)
        return 1;
    return fib(n - 1) + fib (n - 2);
}
```

✓ قسمت ۱۴: علت را برای دستیاران آموزشی توضیح دهید.

برای کسب اطلاعات بیشتر در مورد توابع بازگشتی می‌توانید به این [لینک](#) و یا این [لینک](#) مراجعه کنید.

<sup>3</sup> Recursive functions

موفق باشید.

تهیه و تنظیم: امیر مرتضی رضائی