

به نام خدا

دستور کار آزمایشگاه شماره ۲

آشنایی مقدماتی با C، متغیرها و ورودی و خروجی

در این جلسه قرار است که شما با محاسبات ریاضی در زبان C و محیط Visual Studio 2010 آشنا شوید.

پیش نیاز: مطالعه و انجام خودآموز Visual Studio 2010.

انجام دهید:

۱. یک پروژه ی جدید بسازید و برنامه ای بنویسید که خروجی زیر را چاپ کند:

Hello World!

I want to be a master in C.

۲. راهنمایی:

```
#include <stdio.h>
int main() {
    printf("Hello World!\n");
    //Some other codes you should add here.
}
```

نتایج را به دستیاران آموزشی نشان دهید.

۳. توجه: '\n' یک کاراکتر اسکی^۱ محسوب می شود و یکی از علائم به خصوصی است که می توان در تابع printf از آن استفاده کرد. از جمله علائم مشابه می توان به '\r' اشاره کرد که یک کاراکتر اسکی^۲ است و تاریخچه ی جالبی دارد!

^۱ new line character (10_{Dec})

^۲ carriage return (13_{Dec})

توضیح این که کاراکتر 'r' به هنگام پرینت کردن با پرینترهای قدیمی مورد استفاده قرار می گرفت. بدین گونه که head پرینتر را به ابتدای خط باز می گرداند. این کاراکتر امروزه نیز استفاده می شود و کاربرد آن در سیستم عامل های مختلف کمی با هم متفاوت است.

با مراجعه به لینک زیر در این مورد تحقیق کنید و نتیجه را بر روی یک کاغذ بنویسید و به دستیاران آموزشی تحویل دهید. (این قسمت بخشی از نمره ی شما را تشکیل می دهد).

<http://stackoverflow.com/questions/1279779/what-is-the-difference-between-r-and-n>

عملیات خواندن از ورودی و نوشتن در خروجی توسط دو تابع Printf و Scanf :

توابع scanf و printf به ترتیب توابع ورودی و خروجی استاندارد فرمت دار هستند. یعنی شما می توانید فرمت داده ای که می خواهید بخوانید و یا بنویسید را تعیین کنید.

۱. فرمت های مختلف در جدول زیر آمده است:

Format	Format Specifier
int	%d or %i
char	%c
float	%f
double	%lf
string	%s

فرمت های دیگری نیز وجود دارند که اهمیت کمتری دارند. مثلاً %x برای اعداد hex و ...

انجام دهید:

۱. حال این قطعه کد را در یک پروژه ی جدید اجرا کنید تا توضیحات بالا بیش تر برایتان جا بیفتد.

```
#include <stdio.h>
int main() {
```

```

int x, y;
printf("Enter the first number:\n");
scanf("%d", &x);
printf("Enter the second number:\n");
scanf("%d", &y);
printf("The result is: %d\n", x + y);
}

```

۲. سعی کنید قطعه کد بالا را جوری تغییر دهید که کاربر بتواند دو عدد ورودی را در یک خط و با یک کاراکتر `space` بین آن دو وارد کند. توجه کنید که باید پیغام مناسبی نیز به کاربر داده شود تا وی متوجه شود دو عدد را به چه فرمتی باید وارد کند. مثلاً عبارت زیر برای دادن پیغام مناسب به کاربر، گویا خواهد بود:

```
printf("Please enter two numbers separated by a single space.\n");
```

حال نتایج را به دستیاران آموزشی نشان دهید.

۳. در تابع `scanf`، `\n` در انتهای فرمت قرار ندهید. اگر قرار دهید کاربر باید یک `enter` اضافه وارد کند.

۴. در دستور `scanf`، علامت `&` که قبل از `x` گذاشته شده را بردارید و مجدداً برنامه را کامپایل و اجرا نمایید. حال یک عدد را به عنوان ورودی وارد کنید. چه اتفاقی افتاد؟ چرا برنامه از کار افتاد؟ در این مورد به دو نکته ی زیر توجه کنید:

۱. شما علت دقیق بروز این خطا را بعد از آشنایی با مفاهیم تابع^۳ و نحوه ی آرگمان دهی^۴ به آن، اشاره گر ها^۵ و مرجع ها^۶ در همین درس خواهید آموخت، ولی در حال حاضر همین قدر بدانید که اگر در تابع `scanf` علامت `&` قبل از `x` را فراموش کنید بگذارید، برنامه ی شما می خواهد به خانه ای از حافظه دسترسی داشته باشد که مال خودش نیست! برای همین سیستم عامل جلوی اجرای برنامه را می گیرد.

۲. خطایی که با آن مواجه شدید خطایی بود که به هنگام اجرا رخ داد^۷ و خطای زمان کامپایل^۸ نبود. (زیرا برنامه شما به درستی و بدون خطا کامپایل شد.)

۵. خط اول برنامه فوق (یعنی `#include <stdio.h>`) را حذف کنید و مجدداً برنامه را کامپایل نمایید. چه اتفاقی می افتد؟ پیغام خطایی که کامپایلر به شما می دهد به چه معنا است؟

³ Function

⁴ Argument Passing

⁵ Pointers

⁶ References

⁷ Runtime Error

⁸ Compile Error

علت را برای دستیاران آموزشی توضیح دهید.

انجام دهید:

متغیرها در کامپیوتر به روش‌های مختلفی ذخیره می‌شوند. از این روش‌ها می‌توان روش ASCII برای متغیر از نوع char و یا سیستم نمایش Floating Point را برای اعداد اعشاری یا متغیر float نام برد.

۱. قطعه کد زیر را در یک پروژه جدید اجرا کنید. سپس به عنوان ورودی کاراکتر S را وارد نمایید.

```
#include <stdio.h>
int main() {
    char x;
    printf("Enter a character:\n");
    scanf("%c", &x);
    printf("%d\n", x);
}
```

عدد مشاهده شده در خروجی نمایانگر چه مقداری است؟ علت را برای دستیاران آموزشی توضیح دهید.

۲. قطعه کد زیر را در یک پروژه جدید اجرا کنید. سپس به عنوان ورودی یک بار عدد ۵ و یک بار عدد ۱۰۹۲۰۹۱۹۰۴ را وارد کنید.

```
#include <stdio.h>
int main() {
    float x;
    printf("Enter a decimal number:\n");
    scanf("%d", &x);
    printf("%f\n", x);
}
```

عدد مشاهده شده در خروجی نمایانگر چه مقداری است؟ مقدار عدد مشاهده شده در خروجی به ازای ورودی ۱۰۹۲۰۹۱۹۰۴ را در سیستم Floating Point محاسبه کنید. سپس با استفاده از مبدل اعداد binary به decimal در لینک زیر مقدار عدد را در مبنای ۱۰ محاسبه کنید. چه نتیجه‌ای می‌گیرید؟

<http://www.binaryhexconverter.com/binary-to-decimal-converter>

نتیجه را برای دستیاران آموزشی توضیح دهید.

انجام دهید:

۱. برنامه‌ای بنویسید که عدد اعشاری ۲ را از کاربر بگیرد و مساحت دایره‌ای به شعاع ۲ را حساب کند و نتیجه را تا ۳ رقم اعشار نمایش دهد. مثلاً برای حالت $r = 10$ خروجی به صورت زیر نمایش است:

The result is: 314.160

۲. یادآوری(!): مساحت دایره از فرمول πr^2 محاسبه می شود.

۳. راهنمایی ۱:

```
#include <stdio.h>
#define PI 3.1416

int main() {
    double r, result;
    // get r from input.
    Your code goes here.
    // calculate the result
    Your code goes here.
    // print the result to the console
    Your code goes here.
}
```

۴. راهنمایی ۲: برای نمایش یک عدد اعشاری تا ۳ رقم اعشار می توانید از یکی دیگر از قابلیت های تابع printf استفاده کنید! به کد زیر توجه کنید:

```
float r;
printf("%.3f\n", r);
```

با این کار می توانید عدد اعشاری ۲ را تا ۳ رقم اعشار چاپ کنید.

فرمت ها در دستور printf بسیار جذاب و پرکاربرد هستند و خروجی شما را می توانند خوانا و زیبا کنند. برای آشنایی بیش تر با فرمت ها می توانید به لینک زیر مراجعه کنید:

<http://www.codingunit.com/printf-format-specifiers-format-conversions-and-formatted-output>

۵. توجه ۱: یکی از ویژگی های یک برنامه ی خوب کاربر پسند بودن آن است. برای رعایت این نکته پیش از خواندن ورودی ابتدا باید به کاربر پیغام مناسب بدهید. (مانند قطعه کد صفحه ی قبل)

۶. توجه ۲: به عبارت `#define PI 3.1416` که در ابتدای قطعه کد بالا نوشته شده است توجه کنید با این تعریف در هر جای کدتان می توانید به جای عدد ۳,۱۴۱۶ از label ای به نام PI استفاده کنید!

حال نتایج را به دستیاران آموزشی نشان دهید.

⁹ User friendly

انجام دهید(امتیازی):

۱. قطعه کد زیر را در یک پروژه جدید اجرا کنید. سپس به عنوان ورودی عدد ۱۱۵ را وارد کنید.

```
2. #include <stdio.h>
3. int main() {
4.     char x;
5.     printf("Enter a number:\n");
6.     scanf("%d", &x);
7.     printf("%c\n", x);
8. }
```

مشاهده می کنید که برنامه پس از اجرا با خطای runtime مواجه می شود. علت این خطا چیست؟
نتیجه را با دستیاران آموزشی مطرح کنید.