مبانی کامپیوتر و برنامه سازی

دانشگاه آل طه

رشته کامپیوتر

درس تخصصی

پاییز 1403

رئوس مطالب

- مفاهیم اولیه
- محاسبات در کامپیوتر
- آشنایی با الگوریتم و فلوچارت
- مقدمات برنامه سازی
- فرمت بندی ورودی/خروجی
- دستورها
- توابع
- آزمون و خطایابی برنامه

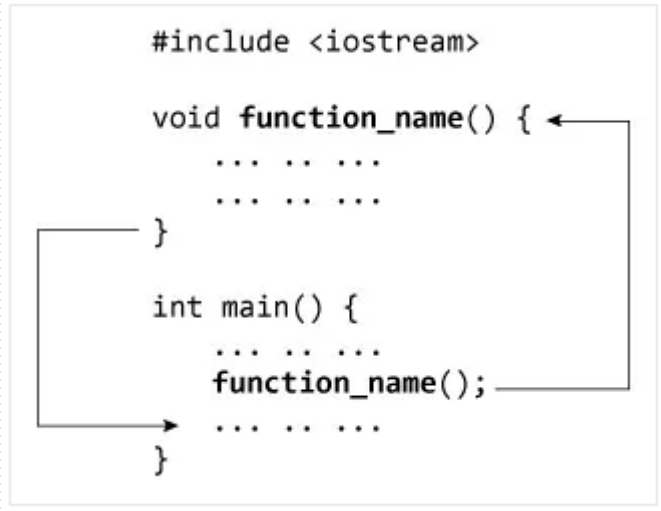
- آرایه ها
- اشاره گرها
- کاراکترها و رشته ها
- ساختارها
- آشنایی با c++
- آشنایی با رده ها
- حوزه های private , public
- پرونده ها

دلایل استفاده از تابع

- تقسیم‌بندی کد به بخش‌های کوچک‌تر و قابل مدیریت‌تر
- تسهیل در همکاری گروهی
- سهولت در عیب‌یابی
- صرفه جویی در وقت
- افزایش قابلیت نگهداری
- استفاده مجدد از کد

نوشتن تابع

- برای نوشتن تابع باید اهداف تابع مشخص باشد
- هر تابع دارای سه جنبه است:
  - تعریف
  - فراخوانی
  - اعلان – الگوی تابع



تعریف تابع

(لیست پارامترها) نام تابع <نوع تابع>

{

دستورات;

}

نکات نوشتن تابع

- نوع تابع مقداری که تابع برمیگرداند را مشخص میکند.
- اگر تابع هیچ مقداری به تابع فراخواننده برنگرداند نوع آن void است.
- پارامترها اطلاعاتی هستند که هنگام فراخوانی تابع، از برنامه فرخواننده به آن ارسال میشوند.
- پارامترها وسیله ای برای تبادل اطلاعات بین تابع فراخواننده و فراخوانی شونده هستند.
- پارامترها با کاما , از یکدیگر جدا می شوند.
- در لیست پارامترها ، نوع هریک از پارامترها نیز مشخص می شود.

نکات نوشتن تابع

- فراخوانی تابع با نام آن انجام میشود.
- نامگذاری برای تابع ، از قانون نامگذاری برای متغیرها تبیعت میکند.
- الگوی تابع قبل از تابع main انجام میشود:

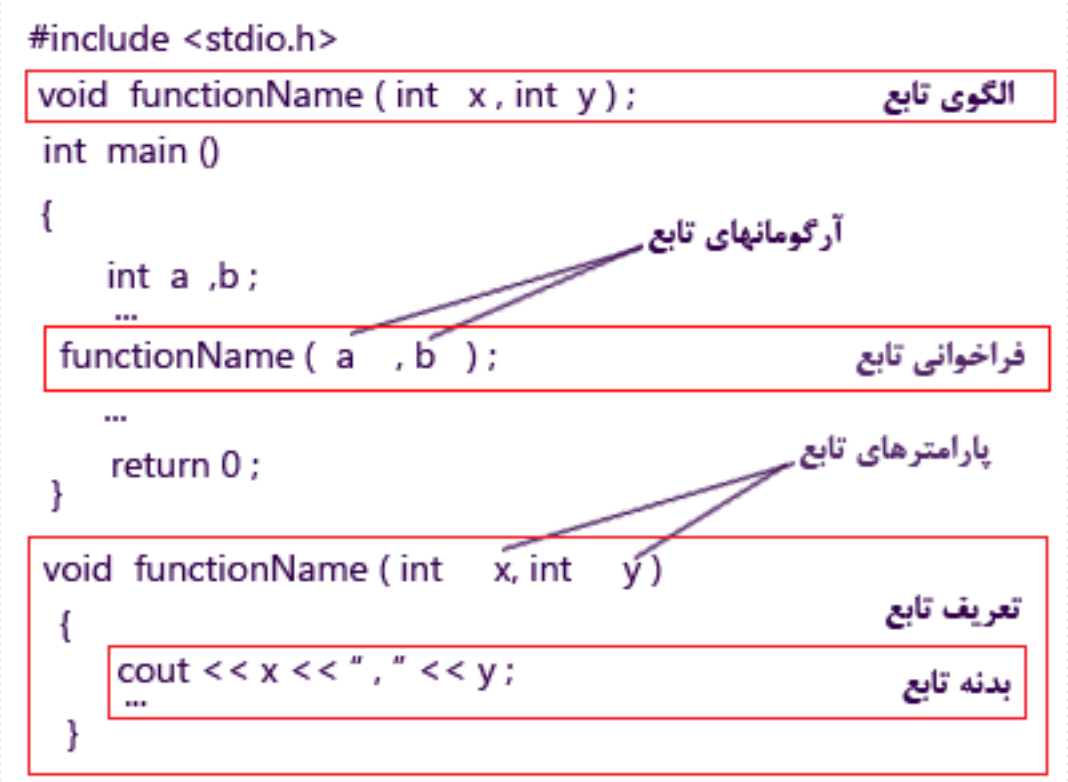
نوع تابع نام تابع (لیست پارامترها);

- متغیر مورد نیاز هرتابع داخل همان تابع تعریف می شود.
- هیچ تابعی نمیتواند از متغیرهای توبع دیگر استفاده کند.
- مقادیر بین توابع از طریق پارامترها منتقل می شود.
- تعرییف تابع در داخل تابع دیگر امکان پذیر نیست.

نکات نوشتن تابع

- توابع از نظر مقدار برگرداننده به تابع اصلی :
  - توابع که هیچ مقداری برنمیگردانند void
  - توابعی که یک مقدار را برمیگردانند.
  - توابعی که چندین مقدار را از طریق پارامترها برمیگردانند.
- هنگام اعلان الگوی تابع، نیاز به ذکر اسامی پارامترها نیست بلکه ذکر نوع پارامترها کفایت میکند.

شیوه به کارگیری تابع در برنامه



روش های فراخوانی تابع

- فراخوانی با ارجاع
- **فراخوانی با مقدار**
  - **در فراخوانی توابع باید نام تابع و نام پارامترها را بیان کنیم**
  - **در اینجا به نام پارامترها، آرگومانهای تابع گفته می شود و دیگر نباید نوع آرگومانها را ذکر کنیم**

توابعی که یک مقدار را برمیگردانند

- به ما اجازه می‌دهند تا نتیجه محاسبات یا عملیات خاصی را به بخش دیگری از برنامه ارسال کنیم.

return عبارت ;

توابع inline

- این توابع به کامپایلر اجازه می‌دهند تا به جای فراخوانی یک تابع، کد آن را به طور مستقیم در محل فراخوانی قرار دهد.
- باعث کاهش سربار فراخوانی تابع و افزایش سرعت اجرای برنامه می‌شود.
- برای تبدیل یک تابع به تابع inline، کافی است قبل از تعریف تابع، کلمه کلیدی inlineرا قرار دهیم

```
inline int square(int x) {
    return x * x;
}
```

محدودیت‌های توابع inline

- تصمیم نهایی با کامپایلر
- برای توابع بزرگ، تبدیل به تابع inline ممکن است باعث افزایش اندازه کد شود و بهینه‌سازی نباشد.
- توابع بازگشتی: توابع بازگشتی به طور معمول به عنوان تابع inlineپیاده‌سازی نمی‌شوند.

توابع همنام Overloaded Functions

- توابع همنام در C++به مجموعه ای از توابع گفته می شود که دارای نام یکسان اما لیست آرگومان های متفاوت هستند.
- این ویژگی به برنامه نویسان اجازه می دهد تا با استفاده از یک نام واحد، عملیات مختلفی را بر روی انواع داده های متفاوت انجام دهند.
- افزایش خوانایی کد
- کاهش احتمال خطا
- انعطاف‌پذیری بیشتر

Overloaded Functions توابع همنام

```cpp
#include <iostream>

using namespace std;

int sum(int a, int b) {
    return a + b;
}

double sum(double a, double b) {
    return a + b;
}

int main() {
    int x = 5, y = 3;
    double a = 2.5, b = 1.7;

    cout << sum(x, y) << endl;  // فراخوانی تابع sum برای اعداد صحیح
    cout << sum(a, b) << endl;  // فراخوانی تابع sum برای اعداد اعشاری

    return 0;
}
```

توضیح مثال قبل

- **در این مثال، دو تابع با نام sumتعریف شده‌اند که یکی برای جمع اعداد صحیح و دیگری برای جمع اعداد اعشاری استفاده می‌شود. کامپایلر با توجه به نوع داده‌های آرگومان‌ها، تابع مناسب را انتخاب می‌کند.**

قوانین تعریف توابع همنام

- **نام یکسان:** تمام توابع همنام باید نام یکسانی داشته باشند.
- **لیست آرگومان‌های متفاوت:** تعداد، ترتیب یا نوع داده‌های آرگومان‌ها باید در هر تابع متفاوت باشد.
- **نوع بازگشتی:** نوع داده بازگشتی می‌تواند یکسان یا متفاوت باشد.

کاربردهای توابع همنام

- **تعریف توابع برای انواع داده‌های مختلف:** مانند توابع ریاضی برای اعداد صحیح، اعشاری و مختلط.
- **تعریف توابع با تعداد آرگومان‌های متفاوت:** برای انجام عملیات‌های مختلف با ورودی‌های متفاوت.
- **تعریف توابع با پارامترهای پیش‌فرض:** برای ارائه مقادیر پیش‌فرض برای برخی از آرگومان‌ها.

توابع بازگشتی

- تابع بازگشتی به تابعی گفته می‌شود که در بدنه خود به خودش فراخوانی می‌شود.
- این نوع توابع برای حل مسائل تکراری و مسائل مربوط به ساختارهای داده‌ای بازگشتی مانند درخت‌ها و لیست‌های پیوندی بسیار مفید هستند.
- فرایند بازگشت تا زمانی که نوعی شرط برقرار شود تداوم می‌یابد.

توابع بازگشتی