

# 计算机扫盲篇

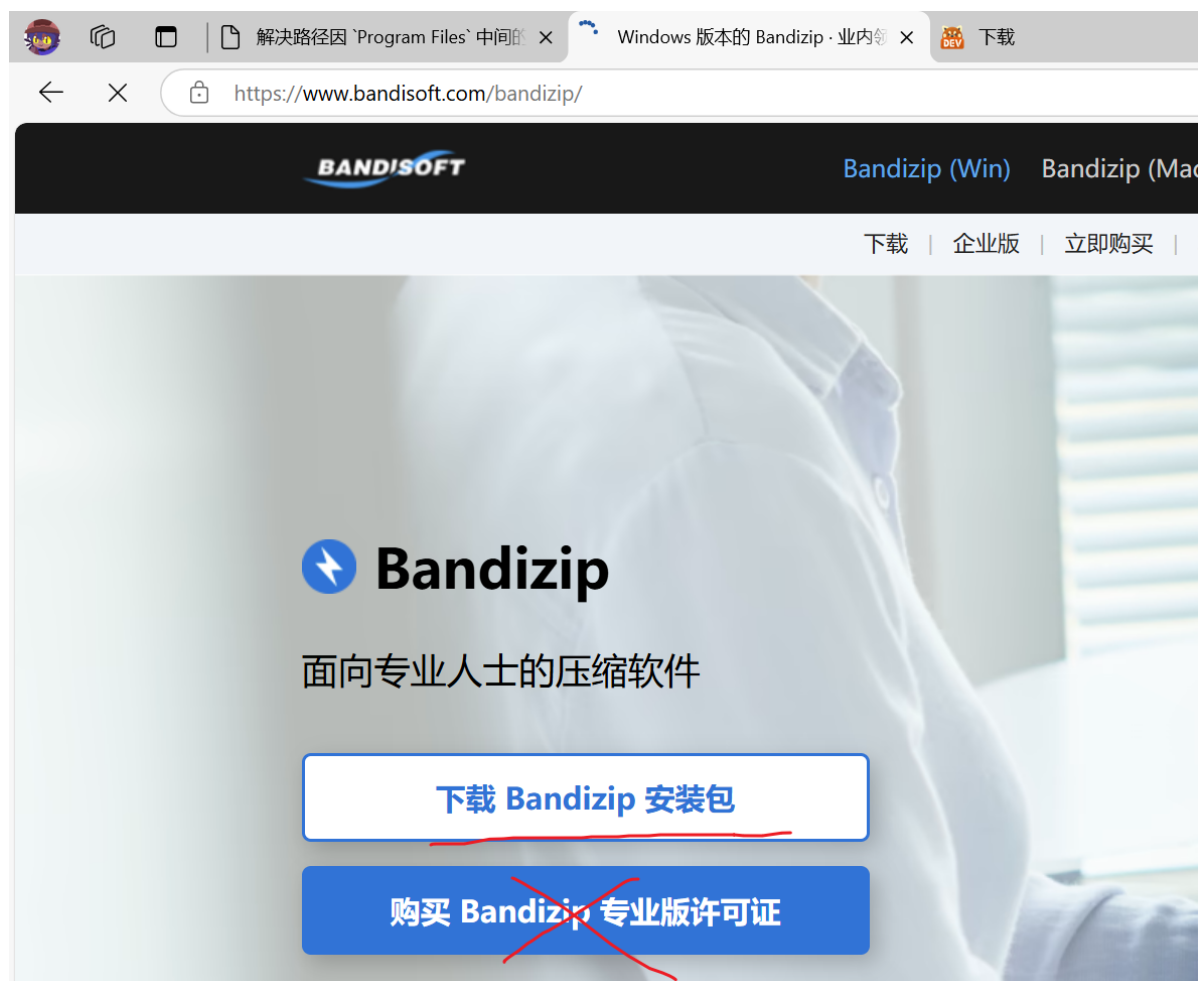
我们用的 所有软件 都是 免费的

## Lesson 0 —— Hello PC

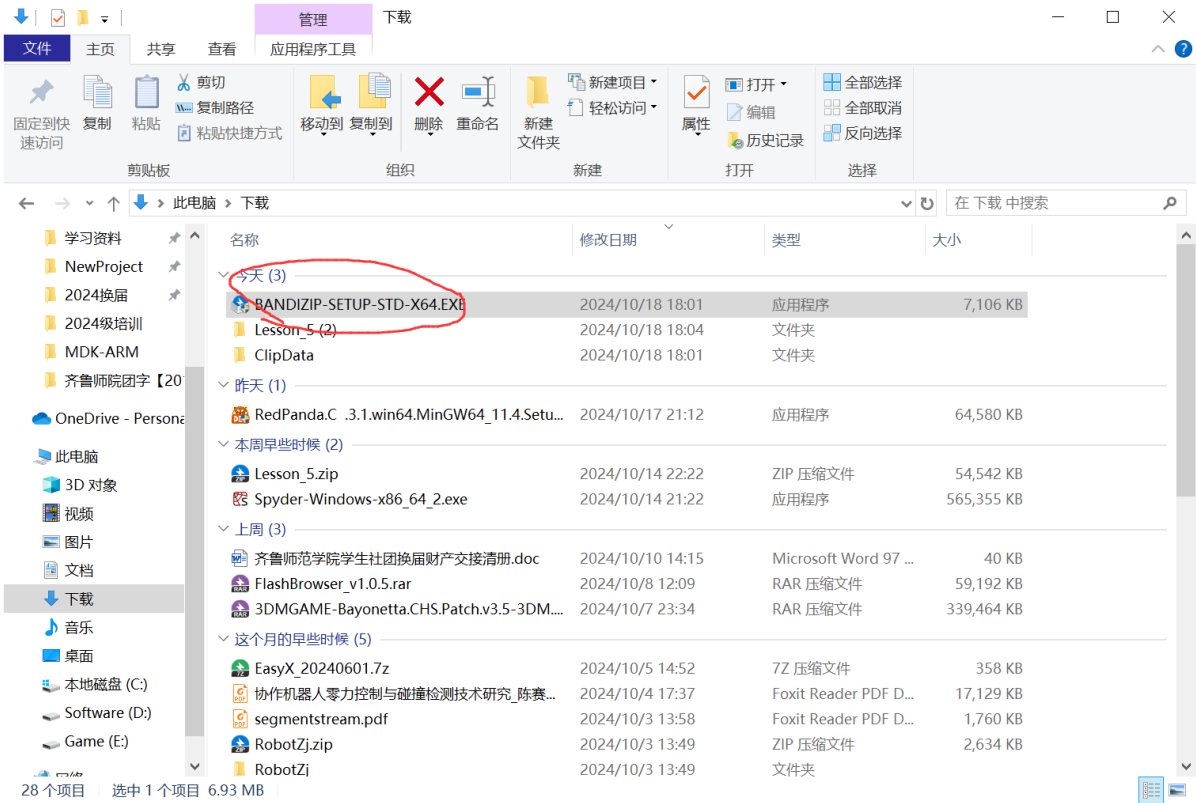
### 压缩软件

[Windows 版本的 Bandizip · 业内领先的压缩文件解决方案 \(bandisoft.com\)](https://www.bandisoft.com/bandizip/)

不要下载专业版，要花钱



# 软件安装








## 常见压缩包类型

.zip .7z .rar

## 解压文件

- 1. 右键压缩包——解压到此处 / 解压到xx / 解压到其他文件夹


## 打开(O)

-  通过 Code 打开
-  解压到此处(E)
-  智能解压到此处(S)
-  解压到 Lesson\_5 (3)\(T)
-  解压到其他文件夹(B)...

无法使用压缩文件预览功能。


请重新启动 Windows 资源管理器，  
或者重新启动计算机。

-  使用 Microsoft Defender扫描...
- 使用小米互传发送

 共享

打开方式(H) ▶

上传到百度网盘  
同步至其它设备

 通过QQ发送到 ▶

还原以前的版本(V)

发送到(N) ▶

剪切(T)

复制(C)

粘贴(P)

创建快捷方式(S)

删除(D)

重命名(M)

属性(R)

看 压缩的

路径


快捷方式


移


下载

名称

今天 (3)

 BANDIZIP


 Lesson\_5

 ClipData

昨天 (1)

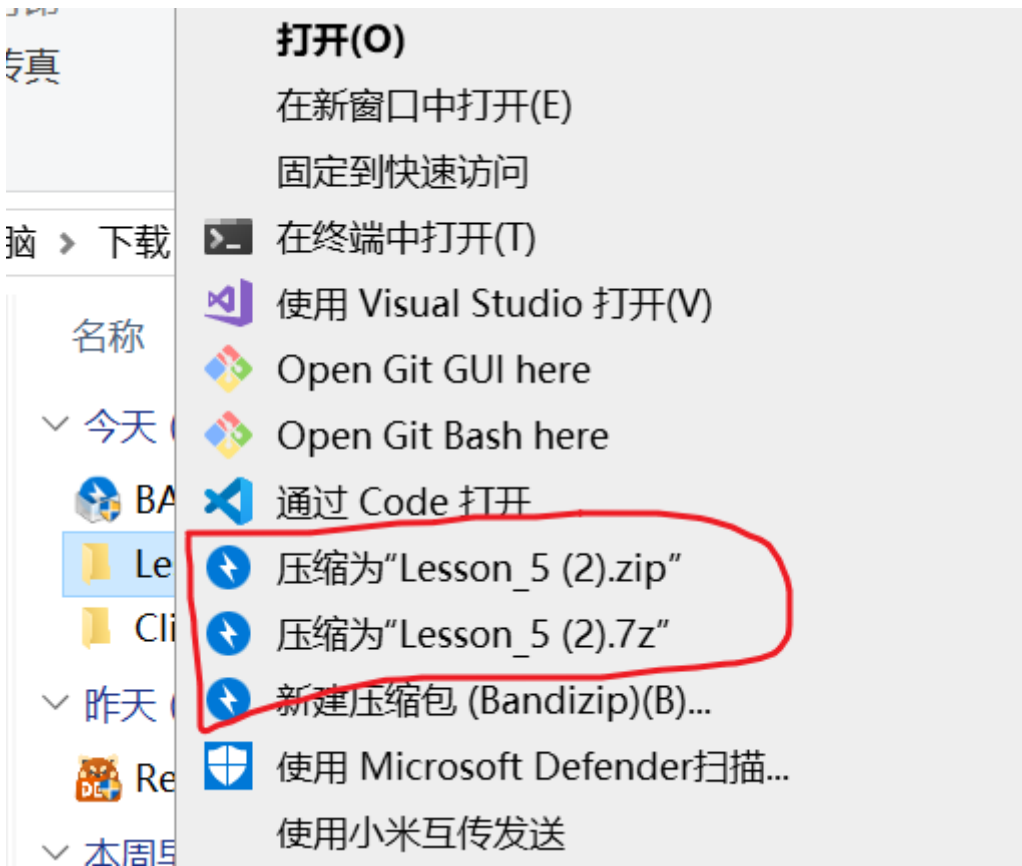
 RedPanda

本周早些时候

 Lesson\_5.zip

# 压缩文件

1. 右键文件 / 文件夹 —— 解压为xx



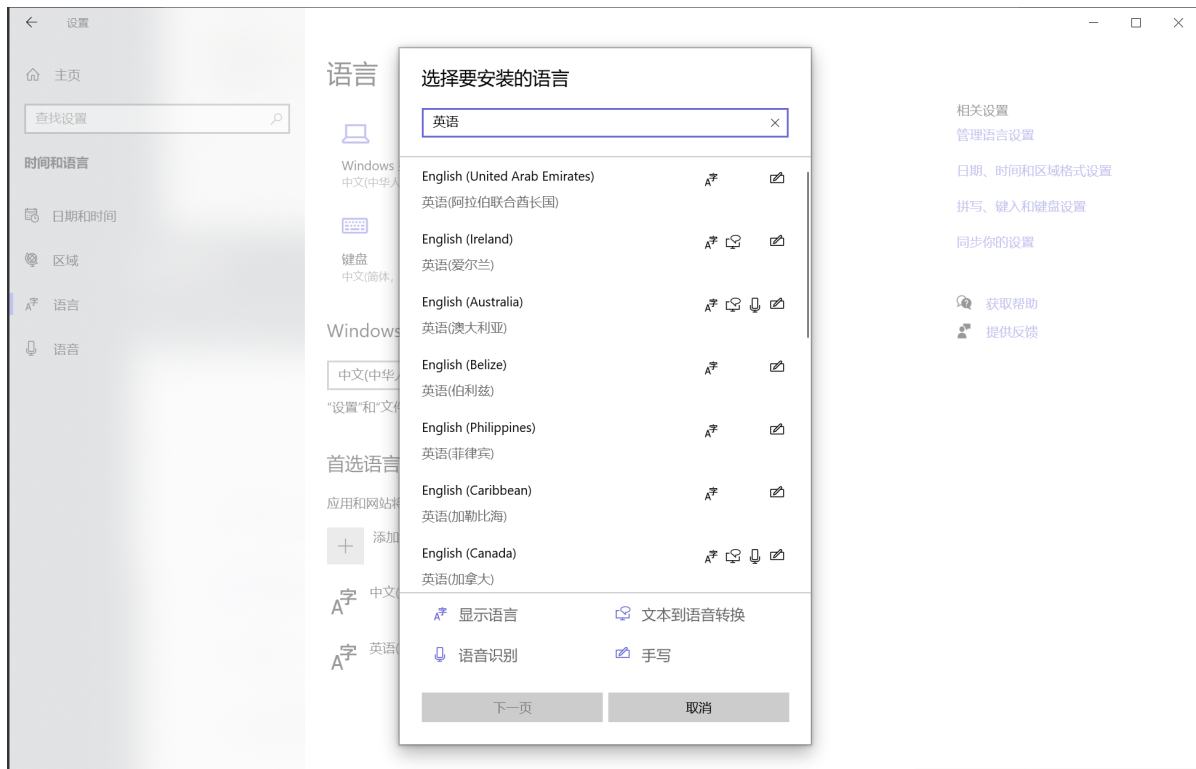
# 语言修改

为了防止全半角造成的对程序的影响

1. 设置 —— 时间与语言



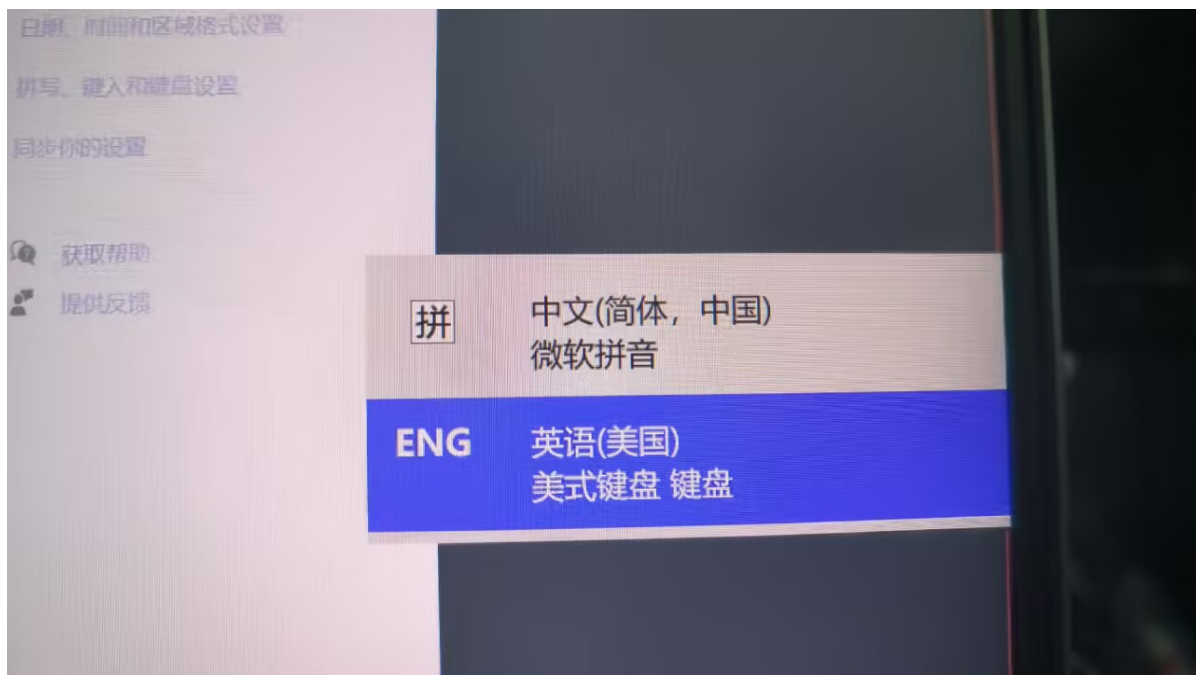
2. 语言——首选语言——添加语言——添加英语语言包



### 3. 切换语言

win10及以上：win + space（空格键）

win7：shift



# C语言培训篇

## 小熊猫c++

### 下载

[小熊猫C++ \(roygh.net\)](#)

如果你的电脑没有安装任何编译器，注意下载有编译器（MinGW32/64）的版本，否则无法编译程序

小熊猫CPP7		
11-05 文件投诉		
文件名	大小	时间
 RedPanda.C++.3.1.win64.NoCompiler.Setup.exe	10.4 M	2024-06-11
 RedPanda.C++.3.1.win64.NoCompiler.Portable.7z	9.0 M	2024-06-11
 RedPanda.C++.3.1.win64.MinGW64_11.4.Setup.exe	63.1 M	2024-06-11
 RedPanda.C++.3.1.win64.MinGW64_11.4.Portable.7z	62.9 M	2024-06-11
 RedPanda.C++.3.1.win32.NoCompiler.Setup.exe	10.9 M	2024-06-11
 RedPanda.C++.3.1.win32.NoCompiler.Portable.7z	9.5 M	2024-06-11
 RedPanda.C++.3.1.win32.MinGW32_11.2.Setup.exe	58.5 M	2024-06-11
 RedPanda.C++.3.1.win32.MinGW32_11.2.Portable.7z	61.3 M	2024-06-11
 小熊猫C++.3.0.winxp.mingw32.绿色版.7z	63.1 M	2024-05-12
 小熊猫C++.3.0.winxp.mingw32.setup.exe	61.4 M	2024-05-12

### 安装

1. 选择下载的文件，右键打开

🏠 > 下载		🔍 在下载 中搜索	
名称	修改日期	类型	大小
📁 今天 (3)			
 BANDIZIP-SETUP-STD-X64.EXE	2024/10/18 18:01	应用程序	7,106 KB
 Lesson_5 (2)	2024/10/18 18:04	文件夹	
 ClipData	2024/10/18 18:01	文件夹	
📁 昨天 (1)			
 RedPanda.C .3.1.win64.MinGW64_11.4.Setu...	2024/10/17 21:12	应用程序	64,580 KB

2. 如果电脑没有装编译器的话默认安装完整版本

## 选择组件

选择你想安装的小熊猫C++ 3.1 (x64) 功能组件。



请勾选你想安装的组件，并取消勾选你不想安装的组件。点击 [下一步(N)] 继续。

选定的安装的类型:

完整

或者，自定义选定想安装的组件:

- ☒ 程序文件 (必需)
  - ☒ MinGW-w64 X86\_64 GCC 11.4 编译器
  - ☒ 关联文件到小熊猫C++
  - ☒ 快捷方式
  - ☐ 压缩安装目录
  - ☐ 删除原有配置文件

所需空间: 492.8 MB

### 组件描述

将光标悬停在组件名称之上，即可显示它的功能描述。

Nullsoft Install System v

< 上一步(P)

下一步(N) >

取消(C)

### 3. 选择安装路径

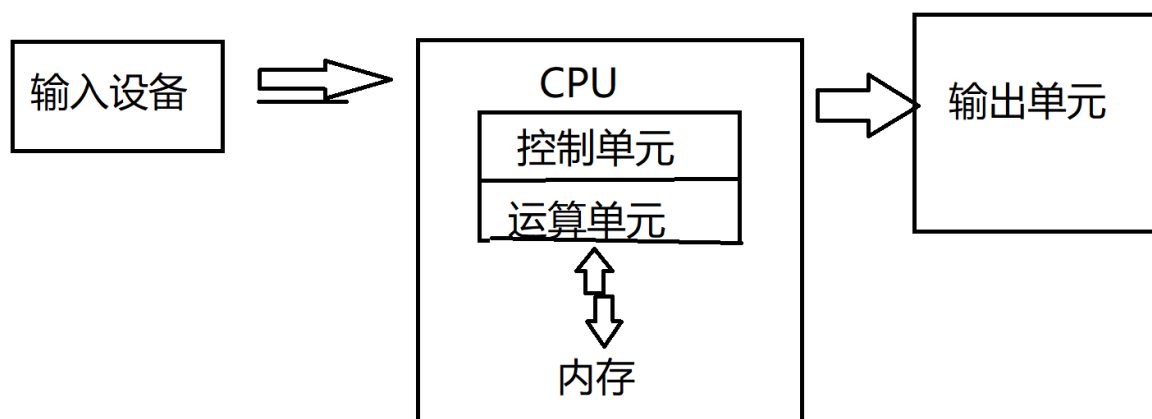


## 程序入门

### 计算机

#### 1. 冯诺依曼架构

##### 冯诺依曼架构



#### 2. 内存



地址

数据

## 数据类型

以32位编译器为例

### 1. 整形：

- short      短整型 8位
- int        整形 16位
- long       长整型 32位
- long long   64位

### 2. 浮点型

- float       单精度浮点 32位
- double     双精度浮点 64位

### 3. 字符型

- char      字符型 8位

### 4. 指针类型

- \*p      变量名为p的指针，32位

## C语言的基本数据类型

本篇文章主要是复习已经学习过的有关C语言的数据类型



<https://blog.csdn.net/aa98865646>

## 运算符

### 算术运算符

- 加      +
- 减      -
- 乘      \*
- 除法    /      (整除)

10 / 3 结果为 3

- 取模    %      (取余)

10 % 3 结果为 1

## 逻辑运算符

- 等于 ==
- 不等于 !=
- 大于 >
- 小于 <
- 大于等于 >=
- 小于等于 <=
- 逻辑与 &&

```
/*
 * 第二次培训 已知分数求等级 （逻辑与）
 */
#include <stdio.h>

int main()
{
    int score;

    printf("input score:");
    scanf("%d", &score);

    if(score >= 90 && score <= 100)
    {
        printf("A");
    }
    else if(score >= 80 && score < 90)
    {
        printf("B");
    }
    else if(score >= 70 && score < 80)
    {
        printf("C");
    }
    else if(score >= 60 && score < 70)
    {
        printf("D");
    }
    else if(score >= 50 && score < 60)
    {
        printf("E");
    }
}
```

```

else
{
    printf("err");
}

return 0;
}

```

- 逻辑或     ||

```

/*
 * 第二次培训 已知等级求分数区间 （逻辑或）
 */
#include <stdio.h>

int main()
{
    char LEVEL;

    printf("input LEVEL:");
    scanf("%c", &LEVEL);

    if(LEVEL == 'A' || LEVEL == 'a')
    {
        printf("90~100");
    }
    else if(LEVEL == 'B' || LEVEL == 'b')
    {
        printf("80~89");
    }
    else if(LEVEL == 'C' || LEVEL == 'c')
    {
        printf("70~79");
    }
    else if(LEVEL == 'D' || LEVEL == 'd')
    {
        printf("60~69");
    }
    else if(LEVEL == 'E' || LEVEL == 'e')
    {
        printf("50~59");
    }
    else

```

```

    {
        printf("err");
    }

    return 0;
}

```

## 位运算符

- &          与运算

```

1 & 1 结果为 1
0 & 1 结果为 0
0 & 1 结果为 0
0 & 0 结果为 0

```

- |          或运算

```

1 | 1 结果为 1
0 | 1 结果为 1
0 | 1 结果为 1
0 | 0 结果为 0

```

### 3. ASCII码

	DEC 十进制	OCT 八进制	HEX 十六进制	BIN 二进制	Symbol 符号	HTML 实体编码
+	97	141	61	01100001	a	&#097;

## 程序的结构

### 1. 顺序结构

```

#include <stdio.h>

int main()
{
    printf("Hello, world!");
    return 0;
}

```

```
int main()
```



```
printf(...);
```



```
return 0
```

## 2. 分支结构

### 2.1 if语句

```

if(判断条件)                // 如果判断条件为真
{
    执行的语句;
}
else if(判断条件)           // 否则 如果判断条件为真
{
    执行的语句;
}
else                        // 否则
{
    执行的语句;
}

```

## 2.2 switch语句

```

switch(需要判断的常量)
{
    case 常量1:              // 与常量1相同
        执行的语句;
        break;
    case 常量2:              // 与常量2相同
        执行的语句;
        break;
    default:                 // 与任一常量都不相同
        执行的语句;
        break;
}

```

```

#include <stdio.h>

int main()
{
    int i;
    printf("input i = ");
    scanf("%d", &i);

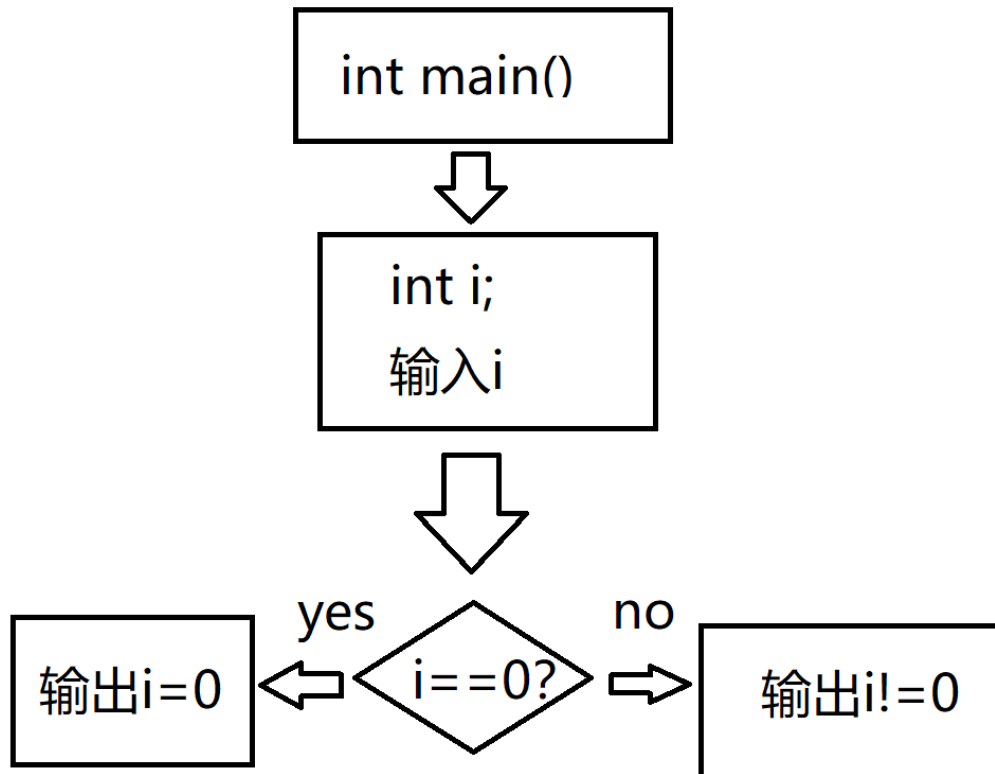
    if(i == 0)
    {
        printf("i == 0");
    }
    else
    {
        printf("i == 1");
    }
}

```

```
    return 0;  
}
```

## 三目运算符

判断条件？ 语句1： 语句2;



## 3. 循环结构

### 3.1 for语句

```
for(初始化; 判断; 更新)  
{  
    执行的语句;  
}
```

```
/**  
 * 阶乘  
 */  
  
#include <stdio.h>  
  
int main()  
{  
    int target;  
    int ret = 1;
```



```

printf("input your num:");
scanf("%d", &target);

for(int i=target; i>0; i--)
{
    printf("%d\n", i);
    ret = ret * i;
}

printf("%d\n", ret);
return 0;
}

```

### 3.2 while语句

```

while(判断条件)    // 先判断,再执行
{
    执行的语句;
}

```

```

/**
 * 猜数
 */
#include <stdio.h>
#include <stdlib.h>

int main()
{
    int target;
    int usr_num;
    target = rand() % 100;           // rand() 函数: 返回一个范围在 0 到
    RAND_MAX 之间的伪随机数。    使用#include <stdlib.h>
    printf("%d\n", target);

    printf("input your num:");
    scanf("%d", &usr_num);
    while(target != usr_num)
    {
        printf("wrong!\n");
        printf("input your num:");
        scanf("%d", &usr_num);
    }
    printf("true!\n");
    return 0;
}

```

```
}
```

### 3.3 do while语句

```
do                // 先执行下列语句
{
    执行的语句;
}while(判断条件); // 后判断
```

## 示例程序

## Q&A

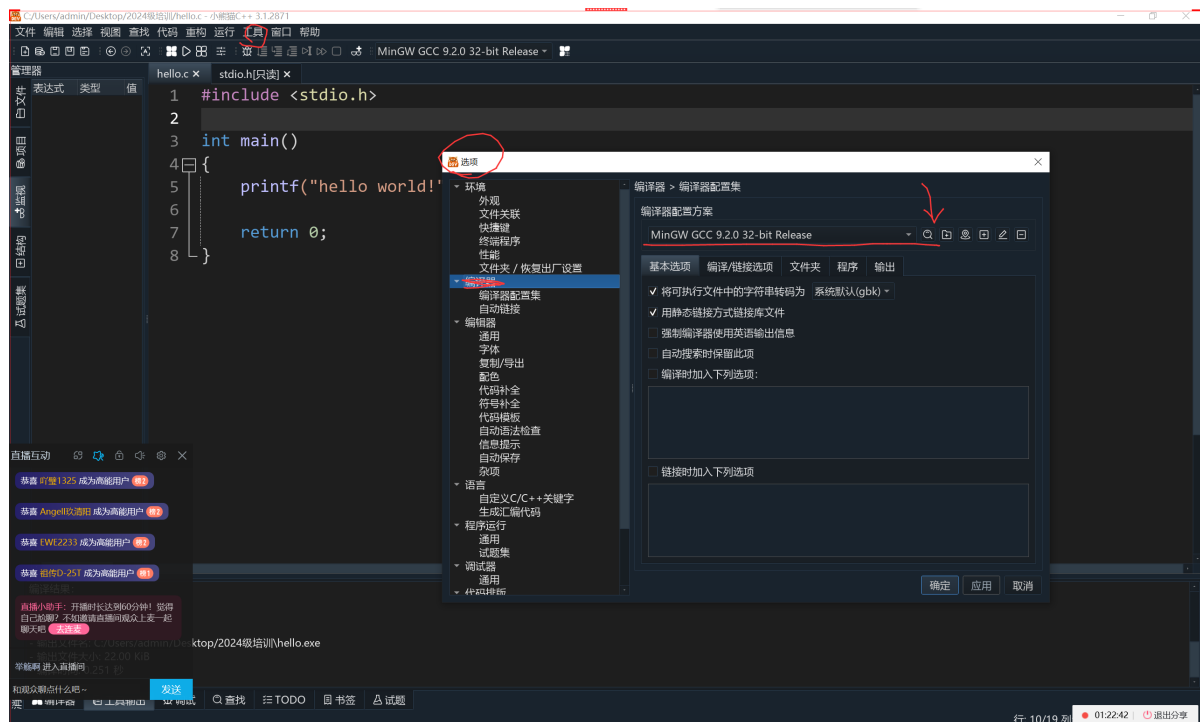
1. Q: 我的大熊猫编译及运行的按钮是灰色的

A: 编译器的选择

上方工具栏 工具——选项——编译器

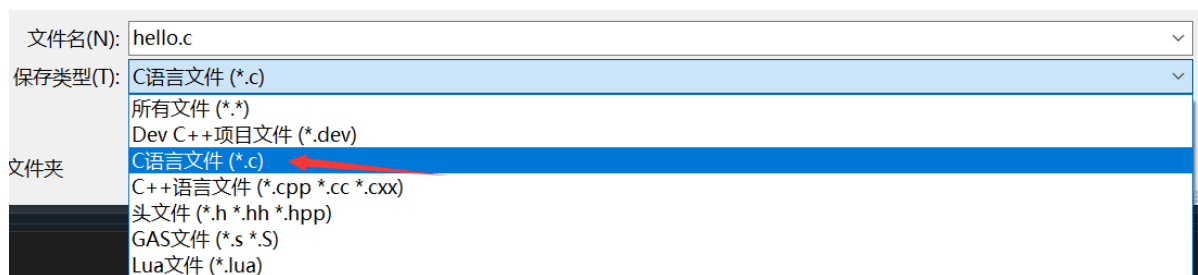
点击放大镜，可以搜索到MinGW编译器

选择Release方案



2. Q: 怎么保存为c文件

A: 文件——另存为——保存类型选择“C语言文件”



3. Q: 我上节课写的代码找不到了怎么办

A: 文件——文件历史 中可以找到之前打开过的文件

4. Q: 我使用for(int i;;)时报错了

A: 检查你的C语言标准，讲标准改为C99及以上即可解决

