



§. 基础知识题 - 变量类型

要求：

- 1、完成本文档中所有的题目并写出分析、运行结果
- 2、无特殊说明，均使用VS2022编译即可
- 3、直接在本文件上作答，**写出答案/截图（不允许手写、手写拍照截图）**即可；填写答案时，为适应所填内容或贴图，**允许调整**页面的字体大小、颜色、文本框的位置等
 - ★ 贴图要有效部分即可，不需要全部内容
 - ★ 在保证一页一题的前提下，具体页面布局可以自行发挥，简单易读即可
 - ★ **不允许**手写在纸上，再拍照贴图
 - ★ **允许**在各种软件工具上完成（不含手写），再截图贴图
 - ★ 如果某题要求VS+Dev的，则如果两个编译器运行结果一致，贴VS的一张图即可，如果不一致，则两个图都要贴
- 4、转换为pdf后提交
- 5、**4月16日前**网上提交本次作业（在“文档作业”中提交）



§. 基础知识题 - 变量类型

贴图要求：只需要截取输出窗口中的有效部分即可，如果全部截取/截取过大，则视为无效贴图

例：无效贴图

```
Microsoft Visual Studio 调试控制台
Hello, world!
D:\WorkSpace\VS2019-Demo\Debug\cpp-demo.exe (进程 7484)已退出, 代码为 0。
按任意键关闭此窗口. . .
```

例：有效贴图

```
Microsoft Visual Studio 调试控制台
Hello, world!
```



§. 基础知识题 - 函数基础

附：用WPS等其他第三方软件打开PPT，将代码复制到VS2022中后，如果出现类似下面的**编译报错**，则观察源程序编辑窗的右下角是否为CR，如果是，单击CR，在弹出中选择CRLF，再次CTRL+F5运行即可

The screenshot shows the Visual Studio 2022 IDE interface. In the top-left, there's a code editor window titled "demo.cpp" with the following content:

```
1 #include <iostream>
2 using namespace std;
3 int main()
4 {
5     cout << "Hello, 同济!" << endl;
6     return 0;
7 }
```

The status bar at the bottom right indicates "行: 4 字符: 2 空格 SYS CR". A red arrow points to the "CR" option in a dropdown menu that appears when the cursor is over the "CR" button. Another red arrow points to the "输出" (Output) tab at the bottom left.

In the bottom half of the screen, the "输出" (Output) tab displays the build log:

```
生成开始于 22:23...
1>—— 已启动生成: 项目: demo-cpp, 配置: Debug Win32 ——
1>demo.cpp
1>D:\WorkSpace\VS2022-demo\demo-cpp\demo.cpp(1,1): warning C4335: 检测到 Mac 文件格式: 请将源文件转换为 DOS 格式或 UNIX 格式
1>D:\WorkSpace\VS2022-demo\demo-cpp\demo.cpp(1,10): warning C4067: 预处理器指令后有意外标记 - 应输入换行符
1>MSVCRTD.lib(exe_main.obj) : error LNK2019: 无法解析的外部符号 _main, 函数 "int __cdecl invoke_main(void)" (?invoke_main@@YAHXZ) 中引用了该符号
1>D:\WorkSpace\VS2022-demo\Debug\demo-cpp.exe : fatal error LNK1120: 1 个无法解析的外部命令
1>已完成生成项目“demo-cpp.vcxproj”的操作 - 失败。
===== 生成: 0 成功, 1 失败, 0 最新, 0 已跳过 =====
===== 生成于 22:23 完成, 耗时 01.132 秒 =====
```

A red box highlights the warning and error messages related to file format and linking.



§. 基础知识题 - 变量类型

1、自动变量及形参的分配与释放

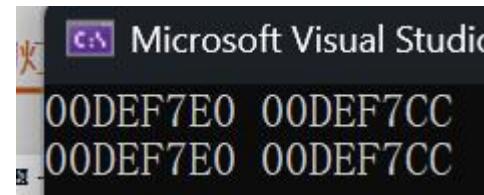
A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

void f1(int x)
{
    int y;
    cout << &x << ',' << &y << endl;//打印地址
}
void f2(long p)
{
    float q;
    cout << &p << ',' << &q << endl;//打印地址
}
int main()
{
    f1(10);
    f2(15L);
    return 0;
}
```

1、运行结果截图及结论

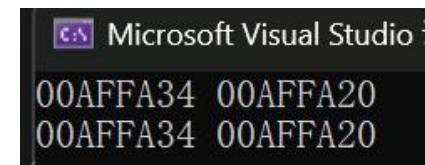
截图：



结论：

1. 1 _x_ 和 _y_ 共用了从00DEF7E0开始的4个字节空间
1. 2 _p_ 和 _q_ 共用了从00DEF7E0开始的4个字节空间

2、把f2中float q改为short q，运行结果截图及结论



结论：

2. 1 _q_ 和 _p_ 共用了从00DEF7E0开始的2个字节空间



§. 基础知识题 - 变量类型

1、自动变量及形参的分配与释放

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;
void f1(int x)
{
    int y;
    cout << &x << ',' << &y << endl;
}
int main()
{
    /* 注意：必须改为自己学号-姓名，否则0分 */
    cout << "2452654-郭炫君" << endl;
    f1(10);
    cout << "..." << endl;

    f1(10);
    cout << "..." << endl;

    f1(10);
    cout << "..." << endl;
    return 0;
}
```

1、运行结果截图

```
2452654-郭炫君
012FFBA0 012FFB8C
...
012FFBA0 012FFB8C
...
012FFBA0 012FFB8C
...
```

2、结论：

2.1 本示例中，三次调用时分配的x占用__相同__(相同/不同)空间，三次调用时分配的y占用__相同__(相同/不同)空间

2.2 总结形参x和自动变量y的分配和释放规则
形参x和自动变量y会在函数被调用的时候分配空间，调用结束之后会在分配的位置处被释放。下次重新在同样的位置分配空间。



§. 基础知识题 - 变量类型

1、自动变量及形参的分配与释放

C. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

void f1()
{
    int a = 15;
    cout << &a << ', ' << a << endl;
}

void f2()
{
    long a = 70000;
    cout << &a << ', ' << a << endl;
}

void f3()
{
    short a = 23;
    cout << &a << ', ' << a << endl;
}

int main()
{
    f1();
    f2();
    f3();
    return 0;
}
```

1、运行结果截图

```
00CFF82C 15
00CFF82C 70000
00CFF82C 23
```

2、结论：

2.1 f1/f2/f3中的三个a占用__相同__ (相同/不同) 空间

2.2 如果当前正在执行f2函数，则f1中的a__已释放__ (未分配/已释放)， f3中的a_未分配__ (未分配/已释放)



§. 基础知识题 - 变量类型

1. 自动变量及形参的分配与释放

D. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;
void f3()
{
    short a = 23;
    cout << "f3" << &a << ',' << a << endl;
}
void f2()
{
    long a = 70000;
    cout << "f2-1" << &a << ',' << a << endl;
    f3();
    cout << "f2-2" << endl;
}
void f1()
{
    int a = 15;
    cout << "f1-1" << &a << ',' << a << endl;
    f2();
    cout << "f1-2" << endl;
}
int main()
{
    /* 注意：必须改为自己学号-姓名，否则0分 */
    cout << "2452654-郭炫君" << endl;
    f1();
    return 0;
}
```

1、运行结果截图

```
2452654-郭炫君
f1-1 005DF680 15
f2-1 005DF59C 70000
f3 005DF4B8 23
f2-2
f1-2
```

2、结论：

2. 1 f1/f2/f3中的三个a占用__不同__(相同/不同)空间
2. 2 如果当前正在执行f1函数的cout-1语句，则f2中的a__未分配__(未分配/已分配/已释放)， f3中的a__未分配__(未分配/已分配/已释放)
2. 3 如果当前正在执行f1函数的cout-2语句，则f2中的a__已释放__(未分配/已分配/已释放)， f3中的a__已释放__(未分配/已分配/已释放)
2. 4 如果当前正在执行f2函数的cout-1语句，则f1中的a__已分配__(未分配/已分配/已释放)， f3中的a__未分配__(未分配/已分配/已释放)
2. 5 如果当前正在执行f2函数的cout-2语句，则f1中的a__已分配__(未分配/已分配/已释放)， f3中的a__已释放__(未分配/已分配/已释放)
2. 6 如果当前正在执行f3函数的cout语句，则f1中的a__已分配__(未分配/已分配/已释放)， f2中的a__已分配__(未分配/已分配/已释放)
2. 7 上述2.2~2.6问题中如果某个a是已分配状态，则此时这个a在何处？

f1中的a在f1的栈帧中，地址00CFF888，f2中的a在f2的栈帧中，地址00CFF7A4，f3中的a在f3的栈帧中，地址00CFF6C0

§. 基础知识题 - 变量类型



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 - 变量类型

2. 局部变量的作用范围

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;
void fun()
{
    int i, a;
    a=15;
    for(i=0;i<10;i++) {
        int y;
        y=11;
        a=16;
    }
    y=12;
    a=17;
}
int main()
{
    /* 注意：必须改为自己学号-姓名，否则0分 */
    cout << "2452654-郭炫君" << endl;
    fun();
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

error C2065: “y” : 未声明的标识符

2、解释出现的error/warning的原因

变量的声明在for循环内部，属于for循环的局部变量，只在for循环的内部可以访问，但在for循环的外部无法被访问到。



§. 基础知识题 - 变量类型

2、局部变量的作用范围

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

void fun()
{
    int i, a=15;
    {
        int y;
        y=11;
        a=16;
        {
            int w=10;
            y=12;
            a=13;
            w=14;
        }
        w=15;
    }
    y=12;
    a=17;
}

int main()
{
    fun();
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

error C2065: “w” : 未声明的标识符
error C2065: “y” : 未声明的标识符

2、解释出现的error/warning的原因

变量声明在函数内部，是函数内部的局部变量，函数外部无法访问。w, y都是在函数内部定义的，外部使用不了。



§. 基础知识题 - 变量类型

2. 局部变量的作用范围

C. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

void fun()
{
    a=14;
}

int main()
{
    /* 注意：必须改为自己学号-姓名，否则0分 */
    cout << "2452654-郭炫君" << endl;

    int a;
    a=15;
    fun();
    a=16;
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图 error C2065: “a” : 未声明的标识符

2、结论：在某个函数(main)中定义的自动变量，在它的调用函数(fun)中__不允许__(允许/不允许)访问

§. 基础知识题 - 变量类型



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 - 变量类型

3、全局变量的作用范围

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int f1()
{
    a=15;
}

int a;

int main()
{
    a=16;
    return 0;
}

int f2()
{
    a=17;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

: error C2065: “a” : 未声明的标识符
上一页

2、解释出现的error/warning的原因
全局变量a在函数f1后才声明的，f1内部的a未被定义，从而报错。



§. 基础知识题 - 变量类型

3. 全局变量的作用范围

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;
int a;
void f1()
{
    a=15;
    cout << "fa=" << a << ', ' << &a << endl;
}
int main()
{
    /* 注意：必须改为自己学号-姓名，否则0分 */
    cout << "2452654-郭炫君" << endl;

    a=10;
    cout << "ma1=" << a << ', ' << &a << endl;
    f1();
    cout << "ma2=" << a << ', ' << &a << endl;
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

```
2452654-郭炫君
ma1=10 0043C19C
fa=15 0043C19C
ma2=15 0043C19C
```

2、由运行结果中的地址可以证明，f1和main中访问的变量a____相同____(不同/相同)的a



§. 基础知识题 - 变量类型

3. 全局变量的作用范围

C. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int f1(int a)
{
    a=15;
    cout << "fa=" << a << ',' << &a << endl;
    return a;
}

int main()
{
    int a =10;
    cout << "ma1=" << a << ',' << &a << endl;
    a = f1(a);
    cout << "ma2=" << a << ',' << &a << endl;

    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

```
ma1=10 0133FC04
fa=15 0133FB30
ma2=15 0133FC04
```

2、由运行结果中的地址可以证明，f1和main中访问的变量a____不同____(不同/相同)的a

3、a不是全局变量，解释为什么ma1和ma2两句cout输出的a值不相同？a是如何被改变的？
cout-ma1和cout-ma2这两句在main函数内部，a是同一个变量，所以地址一样，而main函数内部将a=10作为实参传递给了f1函数，f1函数返回值是15，但这个过程不改变main函数内部a的值，这个返回值赋值给main函数内部的a，a的值发生了变化。

§. 基础知识题 - 变量类型



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 - 变量类型

4. 变量同名

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int a=10, b;
void f1()
{
    int a=5, b;
    cout << "a1=" << a << ',' << &a << endl;
    cout << "b1=" << b << ',' << &b << endl;
}
void f2()
{
    cout << "a2=" << a << ',' << &a << endl;
    cout << "b2=" << b << ',' << &b << endl;
}
int main()
{
    f1();
    f2();
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

```
a1=5 00F9FC44
b1=-858993460 00F9FC38
a2=10 0021C044
b2=0 0021C19C
```

2、由b可知，局部变量不初始化，初值为-858993460；全局变量不初始化，初值为0

3、由截图可知，全局变量a/b的起始地址差332个字节；
局部变量a/b之间差12个字节；全局和局部之前差
13.56MB（单位KB/MB均可），说明这是两个不同的存
储区，全局变量在静态存储区，局部变量在动态
存储区。



§. 基础知识题 - 变量类型

4. 变量同名

B. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

int a=10;
short a;

void f1()
{
    int x=5;
    double x=1.2;

    short p=1, p=2;
}

int main()
{
    f1();
    return 0;
}
```

注：如果是error，贴error截图
如果是warning，贴warning截图+运行结果
如果正常，贴运行结果

1、截图

```
error C2371: “a” : 重定义; 不同的基类型
, 5):

(): error C2371: “x” : 重定义; 不同的基类型
, 9):

(): error C2374: “p” : 重定义; 多次初始化
, 2, 11):

(): error C2086: “short p” : 重定义
, 2,
```

2、结合4.A/4.B可以得知： 不同 (相同/不同)级别的变量允许同名； 相同 (相同/不同)级别的变量不允许同名；变量同名是的使用规则是 作用域不同，允许变量重名，作用域相同，不允许变量同名。

§. 基础知识题 - 变量类型



此页不要删除，也没有意义，仅仅为了分隔题目



§. 基础知识题 - 变量类型

5. 自动变量与静态局部变量

A. 观察下列程序的运行结果，回答问题并将程序的运行结果截图贴上(如果有错则贴错误信息截图)

```
#include <iostream>
using namespace std;

void f1()
{
    int a=1;
    a++;
    cout << "a=" << a << ',' << &a << endl;

    static int b=1;
    b++;
    cout << "b=" << b << ',' << &b << endl;
}

int main()
{
    f1();
    f1();
    f1();
    return 0;
}
```

注: 如果是error, 贴error截图
如果是warning, 贴warning截图+运行结果
如果正常, 贴运行结果

1、截图

```
a=2 006FFB70
b=2 00ACC044
a=2 006FFB70
b=3 00ACC044
a=2 006FFB70
b=4 00ACC044
```

2、结合a/b各自的地址和值，得到结论为：

自动变量a多次调用，则__每次__(每次/仅第一次)进行初始化，函数运行结束后_会__(会/不会)释放空间，下次进入时__再次分配__(再次分配/继续使用上次的空间)

静态局部变量b多次调用，则_仅第一次__(每次/仅第一次)进行初始化，函数运行结束后_不会__(会/不会)释放空间，下次进入时__继续使用上次的空间__(再次分配/继续使用上次的空间)

根据上面的分析结果，自动变量应该放在_动态数据区_____(动态数据区/静态数据区)，静态局部变量应该放在静态数据区_____ (动态数据区/静态数据区)

§. 基础知识题 - 变量类型



此页不要删除，也没有意义，仅仅为了分隔题目