**1.**

请简单说明什么是连接器相关的强符号、弱符号？

**2.**

下列关于静态库链接的描述中，错误的是( )

A. 链接时，链接器只拷贝静态库中被程序引用的目标模块

B. 使用库的一般准则是将它们放在命令行的结尾

C. 如果库不是相互独立的，那么它们必须排序

D. 每个库在命令行只须出现一次即可

**3.**

文本, 信件

AI 生成的内容可能不正确。

**4.**

**在foo.c文件中包含如下代码：**

int foo(void) {

int error = printf("You ran into a problem!\n");

return error;

}

经过编译和链接之后，字符串"You ran into a problem!\n"会出现在哪个段中？

1. .bss
2. .data
3. .rodata
4. .text

**5.**

在此题中，REF(x.i)🡪DEF(x.k)表示链接器将模块i中的符号x引用与模块k中的x定义联系起来。用这种表示方法说明链接器如何解析每个模块中多重定义符号的引用，如果有链接错误则指出“ERR”，如果是可以任意选择则填写“UNKNOWN”。

（1）

/\* Module 1 \*/

int main()

{

}

/\* Module 2 \*/

int main;

int p2()

{

}

请解析main.1和main.2的引用

（2）

/\* Module 1 \*/

void main()

{

}

/\* Module 2 \*/

int main=1;

int p2()

{

}

请解析main.1和main.2的引用

（3）

/\* Module 1 \*/

int x;

void main()

{

}

/\* Module 2 \*/

double x=1.0;

int p2()

{

}

请解析x.1和x.2的引用

**6.**

假设a和b表示当前目录中的目标模块或静态库，而a->b表示a依赖于b，也就是b中定义了一个被a引用的符号。对于以下各种场景，请给出最小的命令行（含有最少的目标文件和库参数的数量），使得静态链接器能够解析所有符号引用。

1）p.o→libx.a→liby.a

2）p.o→libx.a→liby.a且liby.a→libx.a→p.o

**7.**

对于以下myfile.c程序代码

int hello1;

char myfun(int a)

{ int b=10;

hello1=b+a;

hello1=hello1\*f1(a);

printf(“%d\n”,hello1);

return hello1;

}

请指出那些符号是链接器符号，哪些符号需要重定位？

**8.**

图片包含 文本

AI 生成的内容可能不正确。

**9.**

文本

AI 生成的内容可能不正确。

**10.**

ELF格式可用来保存哪些类型的目标文件？

**11.**

文本, 信件

AI 生成的内容可能不正确。

**12.**

文本, 信件

AI 生成的内容可能不正确。

**13.**

可重定位目标文件中机器代码、只读数据、未初始化的全局变量以及已初始 化数据通常分别保存在哪些节里

**14.**

表格

AI 生成的内容可能不正确。

**15.**

将C语言程序到生成可执行文件需要经过哪些步骤？各自的输出是什么？

**16.**

1. 已知如下程序

#include <stdio.h>

int var1=1;

int var2;

int myfun1()

{ return var2+var1;

}

int main(){

{ int c=4;

var1=var2+c+myfun1()+myfun2();

printf(“result var1=: %d \n”,var1);

return;

}

请问执行gcc main.c myfun2.c命令时，在链接开始时扫描完main.o之后，未定义符号集U,已定义符号集D和代码集合E各自包含什么？

**17.**

已知main.o中定义了main()、调用了myfuns.o里的m1()和m2()函数，且调用了printf()函数。在libc.a里的printf.o定义了printf()，libc.a里的sqrt.o定义了sqrt()。在自己编写的myfuns.o里定义了m1()和m2()，其中m1()又调用了sqrt()，m2()则调用了printf()。请问编译时，gcc main.o libc.a myfuns.o和gcc main.o myfuns.o libc.a能否正确产生可执行文件？

**18.**

请问下面main.c和fun1.c两个源文件中出现多处同名变量，是否能正确进行链接并产生可执行文件？如果能，请给出下面代码运行结果；如果不能，则说明理由。

**//main.c**

**int a=10;**

**int main()**

**{ int b=100;**

**a=a+fun1(b);**

**printf (“a=: %d\n”,a);**

**return ;**

**}**

**//fun1.c**

**int b=81;**

**int fun1(int b)**

**{**

**int a=60;**

**a=a+sqrt(b);**

**return a;**

**}**

**19.**

表格

AI 生成的内容可能不正确。

**20.**

文本

AI 生成的内容可能不正确。

**21.**

表格

AI 生成的内容可能不正确。

文本

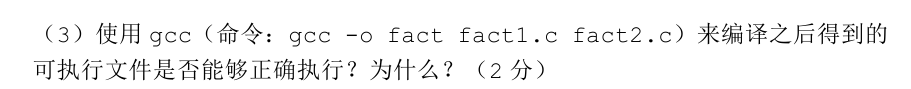
AI 生成的内容可能不正确。

表格

AI 生成的内容可能不正确。

表格

AI 生成的内容可能不正确。



**22.**

对于以下myfile.c程序代码

int hello1;

char myfun(int a)

{ int b=10;

hello1=b+a;

hello1=hello1\*f1(a);

printf(“%d\n”,hello1);

return hello1;

}

1）请指出那些符号是链接器符号，哪些符号需要重定位？

2）如果hello1改为static int hello1，其符号属性会发生什么变化？它会被分配到哪个ELF节？

3）ELF是什么？请画出myfile.o的ELF文件布局（包括节头表、.text、.data、.bss、.symtab、

.rel.text等），并标注hello1和myfun的存储位置。

4）解释f1是如何重定位的？

**答案：**

**1. 强符号是指已经初始化的全局变量和定义的函数名**

**弱符号是指未初始化的全局变量**

**2. 【答案】C。**

**解释：A: 链接时只需拷贝用到的目标模块；B：静态库也可以放在目标文件的前面；C: 正确；D：如果相互调用的库，在命令行必须重复出现。**

**3. B**

**4.** 【答案】C.

**5.** **参考答案：**

REF（2，main）-> DEF（1，main）：（根据C/C++标准，函数和变量符号属于不同类型，但链接器通常不会区分符号类型，仅通过名称匹配）  
或ERR：（同一符号在不同模块中被定义为不同类型，违反了 “强符号优先” 原则，导致链接错误）

（2）ERR

**（3）**REF（1，x）-> DEF（2，x）

**6.** **1) gcc p.o libx.a liby.a**

**2) gcc p.o libx.a liby.a libx.a**

**7.** **链接器符号: hello1，myfun，f1，printf**

**需要重定位: hello1，f1，printf**

**8.** **D**

**9.** **B**

**10.** **可重定位目标二进制文件；可执行文件，共享库**

**11.** **B**

**12.** **C**

**13.** **text节，rodata节，bss节, data节**

**14.** 表格

AI 生成的内容可能不正确。

**15.** **预处理：**主要对源代码进行一些宏替换和头文件包含等操作。main.c → main.i。

**编译：**将预处理后的代码翻译成汇编语言代码。main.i → main.s。

**汇编：**将汇编语言代码转化为机器码指令。main.s → main.o。

**链接：**将目标文件及所需的库函数等链接成最终的可执行文件。main.o → main.out

**16. 暂无**

**17. 暂无**

**18. 暂无**

**19.** **B考点：链接的基本概念，符号解析的原则**

**20.** **C：考察需要进行重定位的条件。A在链接前不在同一目标文件中，BD都 位于.data 段中，这些都需要重定位。C位于栈中或者寄存器中，不需要重 定位。**

**21.**

表格

AI 生成的内容可能不正确。