

LabWindows/CVI程序员认证的考试范围

最终目标：根据给定的一个中等复杂度的规范或测量需求，程序员能够使用LabWindows/CVI给出一个解决方案。

任务	目标
检查CVI环境	识别CVI窗口 <ul style="list-style-type: none"> ● 工作间窗口 ● 用户界面窗口 ● 源代码窗口 ● 输出窗口 ● 变量观察窗口
导航CVI环境	<ul style="list-style-type: none"> ● 菜单 ● 工具栏 ● 选项
演示功能面板的使用	<ul style="list-style-type: none"> ● 访问在线文档 ● 以交互的方式定义变量 ● 通过交互式执行方式运行功能面板
演示调试工具	<ul style="list-style-type: none"> ● 源代码单步运行和调试 ● 演示使用变量观察和其他调试窗口 ● 设置下一条语句
演示程序的编译	<ul style="list-style-type: none"> ● 演示程序编译 ● 比较调试版本和发布版本的程序
开发应用程序的用户界面	演示UI编辑器 <ul style="list-style-type: none"> ● 演示UI工具 ● 演示代码生成功能
演示用户界面组件	<ul style="list-style-type: none"> ● 面板 ● 控件 ● 定制控件——创建与使用 ● 菜单 <ul style="list-style-type: none"> ◇ 工具栏
演示用户界面属性	<ul style="list-style-type: none"> ● 定义回调函数 ● 应用面板或控件的一般设置 <ul style="list-style-type: none"> ◇ 模式——使用并解释 <ul style="list-style-type: none"> ● 热模式 ● 正常模式 ● 验证 ● 指示器
识别与用户界面相关的文件	<ul style="list-style-type: none"> ● 识别以下文件的使用： <ul style="list-style-type: none"> ◇ .uir文件 ◇ .h文件

任务	目标
开发应用程序的代码	<ul style="list-style-type: none"> ● 定义输入、输出数据 <ul style="list-style-type: none"> ✧ 演示CVI扫描函数 ✧ 演示CVI格式化函数 ✧ 演示文件IO函数的使用 ✧ 演示操作定制控件的函数 ✧ 演示UI库函数 <ul style="list-style-type: none"> ● Get/Set（取值/赋值）属性函数 ● Get/Set（取值/赋值）值函数 ● 事件处理函数 ● 加载/卸载函数
开发回调函数的代码	<ul style="list-style-type: none"> ● 用于用户界面组件 ● 用于非用户界面组件（如：DataSocket）
定义应用程序的线程模型	<ul style="list-style-type: none"> ● 定义多线程术语 ● 创建线程 <ul style="list-style-type: none"> ✧ 定义线程池 ✧ 定义线程安全队列和变量
定义应用程序间的通信模型	<ul style="list-style-type: none"> ● 通过ActiveX自动化控件控制软件服务器 ● 通过DataSocket连接传递数据 ● 与LabVIEW接口 ● TCP通信 ● 用于第三方编译器的静态库 <ul style="list-style-type: none"> ✧ .lib文件 ✧ .obj文件 ● 动态链接库（DLL） <ul style="list-style-type: none"> ✧ 使用LabWindows /CVI，创建/调试一个DLL ✧ 调用DLL内的函数 ● 调用Windows API（SDK）
开发仪器驱动程序	<ul style="list-style-type: none"> ● 识别仪器驱动程序组件 <ul style="list-style-type: none"> ✧ 源代码 ✧ 头文件 ✧ 功能面板 ✧ 文档 ● 创建仪器驱动程序 <ul style="list-style-type: none"> ✧ 创建类和函数树 ● 编辑仪器驱动程序 <ul style="list-style-type: none"> ✧ 调试仪器驱动程序 ● 访问仪器驱动程序

任务	目标
配置和发布应用程序	<ul style="list-style-type: none">• 创建可执行的发布程序• 创建带有安装程序的发布工具包• 解释对运行引擎 (run time engine) 的要求• 解释运行时的后向兼容性