

CS410 – Visual Programming

Solved Subjective Portion

Qno.1 How can I use the CopyTo method of the Windows Forms controls collection to copy controls into an array? (3)

Answer:-

The CopyTo method will take the current controls collection and copy all the controls into an array. In order to use it, you must specify the array and the starting point. For instance, the following code copies the controls to the MyArrayOfControls array starting at the first element

Ref <http://msdn.microsoft.com/en-us/magazine/cc164035.aspx>

Qno.2 Write a complete syntax of "Getparents" functions? What kind of value it return and when this function is use?

Answer:-

Qno.3 Differentiate pen &brush?

Answer:-

PEN	Brush
A pen is a graphics tool that an application for Microsoft Windows uses to draw lines and curves	A brush is a graphics tool that a Windows based application uses to paint the interior of polygons, ellipses, and paths
Pens to draw freehand lines, straight lines, and curves.	brushes use to paint shapes
pens to designate trends in graphs and to outline bar graphs and pie charts	Brushes to paint the sections of pie charts and the bars in bar graphs.

REF :: Handouts Page No.69

Qno.4 It is sometimes more efficient for an application to draw directly in a window without relying on the WM_PAINT message. How this task can be accomplished (i.e. how can we draw in a window directly without using WM_PAINT message)? (5)

Answer:-

The task can be accomplished which is given below:-

This can be useful when the user needs immediate feedback, such as when selecting text and dragging or sizing an object. In such cases, the application usually draws while processing keyboard or mouse messages.

To draw in a window without using a WM_PAINT message, the application uses the GetDC or GetDCEX function to retrieve a display device context for the window. With the display device context, the application can draw in the window and avoid intruding into other windows. When the application has finished drawing, it calls the ReleaseDC function to release the display device context for use by other applications. application draws a selection and an intervening WM_PAINT message occurs, the application must ensure that any drawing done during the

message does not corrupt the selection. To avoid this, many applications remove the selection, carry out usual drawing operations, and then restore the selection when drawing is complete

REF :: Handouts Page No.167

Qno.5 How Windows keep track of the files? (3)

Answer:-

The system requires instance handles to keep track of all modules. The system assigns a handle to each copy of a running executable or .dll.

REF :: Handouts Page No.78

Qno.6 Can you write a class without specifying namespace? Which namespace does it belong to by default? (2)

Answer:-

Yes, you can, and then the class belongs to global namespace which has no name. For commercial products, naturally, you wouldn't want global namespace.

REF:: http://www.netsqlinterviewquestions.com/-net_interview_questions/580_can-you-write-a-class-without-specifying-namespace-which-namespace-does-it-belong-to-by-default-.aspx?topicid=2

Qno.7 "In the GDI environment there are two working spaces", Name these two. (2)

Answer:-

In the GDI environment there are two working spaces

1. Logical
2. Physical

REF :: Handouts Page No.140

Qno.8 how many kinds of macros are there?

Answer:-

There are two kinds of macros.

1. Object-like macros resemble data objects when used,
2. Function-like macros resemble function calls.

REF :: Handouts Page No.40

Qno.9 What are the GDI environment working space names?

Answer:-

In the GDI environment there are two working spaces

1. Logical
2. Physical

REF :: Handouts Page No.140

Qno.10 List down three Pre-Defined GDI objects in window

Answer:-

Pre-defined GDI objects in Windows are:

- Pens
- Brushes
- Fonts

- Palettes

REF :: Handouts Page No.178

Qno.11 what are macros and its types? Explain it with example.

Answer:-

There are two kinds of macros.

- 1 Object-like macros resemble data objects when used,
- 2 Function-like macros resemble function calls.

Example:-

Here's a macro that computes the maximum of two numeric values:

```
#define min(X, Y) ((X)>(Y) ? (X):(Y))
```

Explain

To define a macro that takes arguments, you use the #define command with a list of parameters in parentheses after the name of the macro. The parameters may be any valid C identifiers separated by commas at the top level (that is, commas that aren't within parentheses) and, optionally, by white-space characters. The left parenthesis must follow the macro name immediately, with no space in between

REF :: Handouts Page No.40

Qno.12 what is a process?

Answer:-

A series of actions or steps taken to achieve an end, a process is a collection of interrelated work tasks initiated in response to an event that achieves a specific result for the customer of the process.

REF:: https://www.google.com/webhp?source=search_app#hl=en&tbo=d&sclient=psy-ab&q=What+is+Process&oq=What+is+Process&gs_l=hp.3..0l4.1060964.1063012.3.1063209.15.14.0.1.1.2.515.2870.0j11j2j5-1.14.0.les%3B.0.0...1c.1.z147022bvT0&pbx=1&bav=on.2.or.r_gc.r_pw.r_cp.r_qf.&fp=1cb31f4b3d702abf&bpcl=39650382&biw=1024&bih=624

Qno.13 briefly defines Modal Loop?

Answer:-

Modal loop is run by Modal dialogs and process message as does application message loop. That's why program execution is transfer to modal loop so the modal loop itself gets messages and dispatch message.

REF :: Handouts Page No.246

Qno.14 Write windows Programming control process?

Answer:-

- 1 Edit Control
- 2 Static Control

REF :: Handouts Page No.120

Qno.15 Explain Pointer to Constant, and constant to Painter?**Answer:-****Constant pointer to variable data:**

```
char * const ptr = buff.           // constant pointer to variable data
*ptr = 'a';
ptr = buff2;                       // it will be an error
```

since we have declared ptr as a “constant pointer to variable data”, so we can change the contents of the place where ptr is pointing at, i.e. data but being a constant variable, the ptr value i.e. the address it contains cannot be modified.

Variable pointer to Constant data:

```
const char * ptr = buff.           //variable pointer to constant data
*ptr = 'a';                       // it will be an error
ptr = buf2;
```

[REF :: Handouts Page No.64](#)

Qno.16 Write the complete syntax or "get parent function"**Answer:-**

GetParent function returns the parent handle of the specified child. This function will be useful when the parent of the child window to use.

Syntax::

```
HWND GetParent
(
HWND hWnd // handle to child window
);
```

[REF :: Handouts Page No.185](#)

Qno.17 Types of assertion and name them?**Answer:-**

There are three types of assertion:

1 Preconditions

- Specify conditions at the start of a function.

2 Post conditions

- Specify conditions at the end of a function.

3 Invariants

- Specify conditions over a defined region of a program

[REF :: Handouts Page No.49](#)

Qno.18 Write the characteristics of child windows?**Answer:-**

Following are the characteristics of child windows.

- A child window always appears within the client area of its parent window.
- Child windows are most often as controls.
- A child window sends WM_COMMAND notification messages to its parent window.
- When a child window is created a unique identifier for that window is specified in hMenu parameter of CreateWindow()

REF :: Handouts Page No.183

Qno.18 what will happen if GetUpdateRect returns zero?**Answer:-**

An application should call the GetUpdateRect function to determine whether the window has an update region. If GetUpdateRect returns zero, the application should not call the BeginPaint and EndPaint functions.

REF :: Handouts Page No.164

Qno.19 Define Client area?**Answer:-**

The client area is the part of a window where the application displays output, such as text or graphics. For example, a desktop publishing application displays the current page of a document in the client area. The application must provide a function, called a window procedure, to process input to the window and display output in the client area.

REF :: Handouts Page No.84

Qno.20 WIN MAIN describe with detail**Answer:-**

WinMain is the starting point in Every Win32 GUI programs. WinMain has four Parameters these are,

1. First is instance of the current application.
2. Second parameter is also an instance of this application which is used for the previous application of the same type that is already running. It is used only in Window 16bit editions or Windows 3.1. Windows 32bit editions do not support this parameter. It is here just for compatibility.
3. Third parameter is a command line argument of string type which is a type defined as char *.
4. Fourth parameter is windows style.

REF :: Handouts Page No.73

Qno.21 Message queues of each application (2)**Answer:-**

An application must remove and process messages posted to the message queues of its threads

REF :: Handouts Page No.104

Qno.22 Paint function usage (3)**Answer:-**

Paint function performs following tasks.

- The BeginPaint() function prepares the specified window for painting and fills a PAINTSTRUCT structure with information about the painting.
- BeginPaint() first erases the background of window's client area by sending WM_ERASEBKGD message.
- If the function succeeds, the return value is the handle to a display device context for the specified window.

[REF :: Handouts Page No.160](#)

Qno.23 Differentiate Desktop Window and Application Window**Answer:-**

<u>Desktop Windows</u>	<u>Application Windows</u>
When you start the system, it automatically creates the desktop window. The desktop window is a system-defined window	When you start an application, the system also associates a taskbar button with the application. The taskbar button contains the program icon and title
uses a bitmap to paint the background of the screen	Most applications also create other windows, either directly or indirectly, to perform tasks related to the main window
A system configuration application, such as a Control Panel item, changes the desktop wallpaper by using the System Parameters Info function	An application window includes elements such as a title bar, a menu bar, the window menu (formerly known as the system menu),

[REF :: Handouts Page No.114](#)

Qno.24 show the implementation of _cdecl calling convolution with respect to (5)**Answer:-**

- 1: Argument passing order.
- 2: stack maintenance responsibility.
- 3: name decoration convention.

[REF :: Handouts Page No.54](#)

Qno.25 What happened if GetUpdateWn returns zero (3)**Answer:-**

If GetUpdateRect returns zero, the application should not call the BeginPaint and EndPaint functions.

[REF :: Handouts Page No.164](#)

Qno.26 A window may have more than one windows inside it explain the line(2)**Answer:-**

A Window may further contain more windows inside it. For example let's take a calculator; A calculator contains more windows in forms of buttons, radio buttons and check boxes.

- Every Window has its parent and zero or more siblings.
- Top level window has desktop as its parent.

[REF :: Handouts Page No.113](#)

Qno.27 What is the function of ws_paint in Windows class(3)

Answer:-

DispatchMessage function to a window procedure when the application obtains a WM_PAINT message from message Queue by using the GetMessage or PeekMessage functions.

[REF :: Handouts Page No.113](#)

Qno.28 Message queuing 2 marks

Answer:-

Message Queue is created when every any GDI function call is made or send message or post Message function calls are made. Message Queue can be attached to every thread either it is User interface thread or worker threads. User Interface threads always a message queue

[REF :: Handouts Page No.332](#)

Qno.29 Kernel tasks 3marks

Answer:-

Kernel is a main module of the operating system. This provides system services for managing threads, memory, and resources.

Kernel has to perform very important responsibilities e.g.

1. Process Management
2. File Management
3. Memory Management (System and Virtual Memory)

[REF :: Handouts Page No.68](#)

Qno.30 Difference between __stdcall and __cdecl calling convention

Answer:-

cdecl and __stdcall just tells the compiler whether the called function or the calling function cleans up the stack. In __stdcall calling convention, the called function cleans up the stack when it is about to return. So if it is called in a bunch of different places, all of those calls do not need to extra code to clean up the stack after the function call.

In __cdecl calling convention, it is the caller function that is responsible for cleaning the stack, so every function call must also need to include extra code to clean up the stack after the function call.

[REF :: Handouts Page No.54](#)

Qno.31 Erase window function explanation 5marks

Answer:-

Erase windows uses the GetClipBox function to retrieve the logical coordinates of the area to erase and passes these coordinates to the FillRect function. Applications that process these messages can use similar techniques. The system supplies a window device context with the WM_ICONERASEBKGND message regardless of whether the corresponding window has a private device context.

[REF :: Handouts Page No.154](#)

Qno.32 Clipboard Working 3mraks

Answer:-

In Windows, data is shareable among applications. We can use it for copying the data from one file to the other in same format.e.g from notepad to MS Word.. All the text or image data you have previously copied can now be pasted in other application.

REF :: Handouts Page No.72

Qno.33 write down complete syntax of "getDC" function?(2)

Answer:-

The system retrieves a device context from the cache whenever an application calls the GetDC or BeginPaint function; the system returns the DC to the cache when the application subsequently calls the ReleaseDC or EndPaint function.

REF :: Handouts Page No.151

Qno.34 infinite recursion (3)

Answer:-

Infinite recursion, a special case of an infinite loop that is caused by recursion. This revised function will only run out of stack space if n is less than 1 or n is too large; error checking would remove the first case. For information on recursive functions which never run out of stack

REF :: http://en.wikipedia.org/wiki/Infinite_loop#Infinite_recursion

Qno.35 what is stack? (2)

Answer:-

In computer science, a stack is an area of memory that holds all local variables and parameter used by any function, and remembers the order in which functions are called so that function returns occur correctly.

REF:: [http://en.wikipedia.org/wiki/Stack_\(abstract_data_type\)](http://en.wikipedia.org/wiki/Stack_(abstract_data_type))

Qno.36 what is extern storage class?

Answer:-

Extern defines a global variable that is visible to all object modules. When you use 'extern' the variable cannot be initialized as all it does is to point the variable name at a storage location that has been previously defined

REF :: Handouts Page No.60

Qno.37 An application can set up for itself any logical coordinates system, using API. Write down any two.(3)

Answer:-

There are two types of brushes: logical and physical. A logical brush is one that you define in code as the ideal combination of colors and/or pattern that an application should use to paint shapes. A physical brush is one that a device driver creates, which is based on your logical-brush definition.

REF :: Handouts Page No.70

Qno.38 what happened if an application does not process WM_ErasebkGrd messages but pass it defWindowProcs.(5)

Answer:-

If it processes **WM_ERASEBKGD**, the application should use the message's *wParam* parameter to draw the background. This parameter contains a handle to the display device context for the window. After drawing the background, the application should return a nonzero value. This ensures that **BeginPaint** does not erroneously set the **fErase** member of the **PAINTSTRUCT** structure to a nonzero value (indicating the background should be erased) when the application processes the subsequent **WM_PAINT** message.

An application can define a class background brush by assigning a brush handle or a system color value to the **hbrBackground** member of the **WNDCLASS** structure when registering the class with the **RegisterClass** function. The **GetStockObject** or **CreateSolidBrush** function can be used to create a brush handle. A system color value can be one of those defined for the **SetSysColors** function. (The value must be increased by one before it is assigned to the member.)

REF :: Handouts Page No.167

Qno.38 Define "Virtual-Key" message - 2 Marks

Answer:-

virtual-key code is a device-independent value defined by the system that identifies the purpose of a key. After translating a scan code, the keyboard layout creates a message that includes the scan code, the virtual-key code, and other information about the keystroke, and then places the message in the system message queue.

REF :: Handouts Page No.192

Qno.39 Differentiate between Super Classing and Sub Classing. 3 Marks

Answer:-

SuperClassing	SubClassing
Super-classing defines a class that adds new functionality to a predefined window class,	<i>Subclassing is allowed only within a process.</i>
Button or list box controls.	Win32 processes have separate address spaces
<i>Superclassing</i> involves creating a new class that uses the window procedure of an existing class for basic functionality.	<i>An application cannot subclass a window or class that belongs to another process.</i>

REF :: Handouts Page No.132

Qno.40 why the entries in the parent process table and child table? 2

Answer:- It means that the handle value that identifies a kernel object is identical in both the parent and the child processes

REF :: <http://www.coolinterview.com/interview/54829/>

Qno.41 Define instance Handle? Briefly explain

Answer:- This member is Application instance handle.

REF :: Handouts Page No.90

Qno.42 List down the important distance between subclassing and superclassing?

Answer:-

SuperClassing	SubClassing
Super-classing defines a class that adds new functionality to a predefined window class,	<i>Subclassing is allowed only within a process.</i>
Button or list box controls.	Win32 processes have separate address spaces
<i>Superclassing</i> involves creating a new class that uses the window procedure of an existing class for basic functionality.	<i>An application cannot subclass a window or class that belongs to another process.</i>

REF :: Handouts Page No.132

Qno.43 Write down the code c/c++ program that has 2 functions on take 4 integer variable as parameters and return their sum and other take 4 integer as arguments and return their multiplication.

Answer

```
#include <stdio.h>

// prototypes, the parameter names are optional,
// print the menu and obtains a selection
int GetChoice();
// inserts a number in the sorted array.
int AddNum (int num[], int sz);
// removes a number from the array.
int DelNum(int num[], int sz);
// prints out the sorted array
void PrintNums(int num[], int sz);

void main()
{
    // the number of elements in array A[]
    int Size = 0,
    // the array that will keep its numbers sorted
    A[20],
    // the selection made from the menu
    Selection;

    // keep doing this loop until a 4 or Quit is
    // selected from the menu...
    Selection = GetChoice();
    for( ; Selection != 4; )
    {
        if(Selection == 1)
            Size = AddNum(A, Size);
        else
            if(Selection == 2)
                Size = DelNum(A, Size);
            else
                PrintNums(A, Size);
        Selection = GetChoice();
    }
}

// assuming that user will enter only a valid int
int GetChoice()
{
    int Choice;

    printf("Enter 1 - Insert, 2 - Delete, ");
    printf("3 - List and 4 - Quit: ");
    scanf_s("%d", &Choice);
    return Choice;
}
```

```

}

int AddNum(int Num[ ], int sz)
{
    // local variables...
    int i, j, Number;
    // gets the number to insert
    printf("What number to insert? ");
    scanf_s("%d", &Number);
    // finds the place (i) to put the new number
    for(i = 0; i < sz; ++i)
        if(Number < Num[i])
            break;

    // shift the rest of the array by moving
    // the numbers up by one slot.
    for(j = sz; j > i; --j)
        Num[j] = Num[j - 1];
    // place the new number
    Num[i] = Number;
    // the array size is incremented
    // and return to the calling function
    return ++sz;
}

// assuming that the deleted item exists
int DelNum(int Num[], int sz)
{
    // local variables...
    int i, Number;
    // gets the number to be deleted
    printf("What number are you going to delete? ");
    scanf_s("%d", &Number);
    // find the place in the array to be deleted
    for(i = 0; i < sz; ++i)
        if(Number == Num[i])
            break;

    // the array size is decremented
    --sz;
    // shift the rest of the array by moving the numbers
    // down by one slot
    for( ; i < sz; ++i)
        Num[i] = Num[i + 1];
    // return sz to the calling function
    return sz;
}

void PrintNums(int Num[ ], int sz)
{
    // local variable...
    int i;

    for(i = 0; i < sz; ++i)
        printf("Num[%d] = %d\t", i, Num[i]);
    printf("\n");
}

```

Output of this Program

```

C:\WINDOWS\system32\cmd.exe
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 1
What number to insert? 40
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 1
What number to insert? 20
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 1
What number to insert? 60
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 3
Num[0] = 20    Num[1] = 40    Num[2] = 60
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 2
What number are you going to delete? 40
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 3
Num[0] = 20    Num[1] = 60
Enter 1 - Insert, 2 - Delete, 3 - List and 4 - Quit: 4
Press any key to continue . . .

```

REF <http://www.tenouk.com/clabworksheet/labworksheet13.html>

Qno.44 Write two macros that perform the same tasks as these functions perform? 5

Answer:-

A *macro* is a fragment of code which has been given a name. Whenever the name is used, it is replaced by the contents of the macro. There are two kinds of macros. They differ mostly in what they look like when they are used. *Object-like* macros resemble data objects when used, *function-like* macros resemble function calls.

You may define any valid identifier as a macro, even if it is a C keyword. The preprocessor does not know anything about keywords. This can be useful if you wish to hide a keyword such as `const` from an older compiler that does not understand it. However, the preprocessor operator `defined` can never be defined as a macro, and C++'s named operators cannot be macros when you are compiling C++.

To define a macro that takes arguments, you use the `#define` command with a list of parameters in parentheses after the name of the macro. The parameters may be any valid C identifiers separated by commas at the top level (that is, commas that aren't within parentheses) and, optionally, by white-space characters. The left parenthesis must follow the macro name immediately, with no space in between.

For example, here's a macro that computes the maximum of two numeric values:

```
#define min(X, Y) ((X)>(Y) ? (X):(Y))
```

[REF :: Handouts Page No.40](#)

Qno.45 Two common controls in window programming

Answer:-

Common controls are of these types.

1. Date Time Picker Control.
2. List View Control.

[REF :: Handouts Page No.121](#)

Qno.46 Write at least two tasks performed by BeginPaint () function? 3

Answer:-

- 1 The `BeginPaint()` function prepares the specified window for painting and fills a `PAINTSTRUCT` structure with information about the painting.
- 2 `BeginPaint()` first erases the background of window's client area by sending `WM_ERASEBKGND` message.

[REF :: Handouts Page No.160](#)

Qno.47 about _cdecl calling convention?

Answer:-

The `__cdecl` is the default calling convention for C programs. In this calling convention, the stack is cleaned up by the caller. The `__cdecl` calling convention creates larger executables than `__stdcall`, because it requires each function call to include stack cleanup code.

[REF :: Handouts Page No.54](#)

CS410 Visual Programming Solved Subjective

For Final Term Exam Preparation by Virtualians Social Network

MS-DOS and Windows operating

Only one DOS program can be executed at a given time and these programs own the system resources. While in Windows, we can execute several different programs simultaneously. Windows operating system doesn't give us the direct access to interrupts, video ports and memory etc.

Pointers

Pointers are very important and useful as with the help of them we can access a very large data structure.

Function Pointers

That is such pointers which points to the address of the function.

Arrays

Arrays are basically a data structure that is used to store homogenous or same type of data in it. A very useful thing, when we pass the name of the array as an argument to some function, then only the memory address of array is passed as parameters to the function and not all the values of an array are passed. In the end we

The custom or user defined data types

The custom or user defined data types are those data types which we create by our own selves according to the requirements or the given situation. The most commonly used custom data types include structures, unions etc. Unlike arrays, structures can store data members of different data types.

Unions

Unions are also a very important custom data type that at a given time contains only one element from its member list.

Enum declarations

Enum declarations create a new integer type. The integer type is chosen to represent the values of an enumeration type. Thus, a variable declared as enum is an int.

Typedefs

Typedefs are also used for making a synonym or provides way to shorten the long chain of keywords in declarations.

Preprocessor

The preprocessor is a program that runs prior to compilation and potentially modifies a source code file. It may add code in response to the `#include` directive, conditionally include code in response to `#if`, `#ifdef`, `#ifndef` directives or define constants using the

`#define` directive.

Macro

A simple macro is a kind of abbreviation. It is a name which stands for a fragment of code. Some standard pre-defined Macros include `FILE` , `LINE` , `DATE` , `__TIME` etc.

Bitwise Operators

The three major bitwise operators AND, OR, XOR. Bitwise operators operates on individual bits. Using “typedefs” provide an easy way to avoid the long names during the declarations and thus make our code more simple.

Typecasting

It is making a variable of one type, act like another type for one single application. The two types of type casting include the implicit type casting and the explicit type casting.

Assertions

In C, the assertions are implemented with standard `assert` macro, the argument to assert must be true when the macro is executed, otherwise the programmes aborts and printouts an error message.

Switch-Case Statement

The switch-case statement is a multi-way decision statement. Unlike the multiple decisions statement that can be created using if-else, the switch statement evaluates the conditional expression and tests it against numerous constant values.

Calling Conventions

The calling conventions tells us in which order the parameters will be passed in a function and whether the calling function or the called function is responsible for the cleaning of the stack.

The default calling convention for C programs is `cdecl` and in this convention, The caller is responsible for cleaning the stack after the function call. Similarly, the default calling convention for the windows programs is `stdcall`. Here the called function itself has to do the stack clean up and so no extra code is required for stack clean up with each function call. It is very obvious that the `cdecl` calling convention creates larger executables because it requires each function call to include the clean up code.

Storage classes

Storage classes are used to describe the scope and visibility of the variables and functions. The common storage classes are `auto`, `register`, `static`, and `extern` etc.

Lifetime of Variables

In the lifetime of variables each `{ }` block creates a new scope. Variables declared and initialized in a scope are deleted when execution leaves that scope.

Const Keyword

The `const` keyword is used to create a read only variable. Thus constant variables are not allowed to be modified after initialization.

Command line arguments

Command line arguments provide an easy way to pass some parameters to the programme in the main function when the programme execution starts. When using an executable that requires startup arguments to debug, we can type these arguments at the command line, or from within the development environment

Windows Components

Kernel, GDI and User are windows components.

Kernel Component

Kernel is the heart of Operating system.

GDI Component

GDI is a Graphics device interface in windows which is used to display and print graphics and

text objects.

User Component

The user component is responsible to control all the dialogs, menu, windows, Windows controls, etc.

‘Handles in windows’

„Handles in windows“ are the introductory part of handles used in windows.

Handles are 32 bit number that may be void * type which is defined in „WinUser.h“ header file.

WinMain

WinMain is the starting point of Windows Programs. We always write WinMain in every windows program

Register Window class

Register Window class using RegisterClass API

CreateWindow

CreateWindow API uses window class name, caption name, style of window, starting points, width and height of windows and windows parent handle or handle to owner mainly.

Window Handle

We have window handle in variable of type handle to window. Using window handle we can send different types of messages to the window.

Windows message procedure

After getting messages from application message queue, we dispatch messages to the windows message handling procedure. Windows message procedure inputs four parameters. These parameters are also the members of MSG structure. These parameters are important for everyone who is developing applications for Windows.

Window Application

For creating window, we register window class with the attributes required. After successful registration we create window. For creating window we use `CreateWindow` API. In

CreateWindow API we mention some of the Window styles. Window styles are used to create different styles of windows.

Windows Types

These are owned windows and child windows. We must keep this point in mind that owned windows and child windows have different concepts.

Owned Windows

Owned windows have the handle of their owner windows, these handle make a chain of owned windows. If we bring some change to owner window then the owned windows will response on some changes like minimize and destroying operations.

Child windows

These are the part of its parent's client area.

Threads

Threads are two types one is User interface thread and second is working thread. UI (User interface) thread is attached with user interfaces like windows, messages and dialog boxes.

System Window Classes

System Window Classes include buttons, combo boxes, list box, etc.

SetWindowLong and GetWindowLong APIs

We can change windows attributes by using SetWindowLong and GetWindowLong APIs. Using SetWindowLong and GetWindowLong, we can also make a new procedure and change the message behavior of a window. Using SetClassLong and GetClassLong, we can change one of the attributes of a registered class.

Subclassing

Changing class values will affect the change for every window that is using this class. This is called subclassing.

Super-classing

We register new window class by using the properties of previous window class.

GDI

GDI is window's most important component. GDI is very much useful for every programmer because it gives platform independent interface. So whenever we want to write something on screen or on printer we take a device context of that particular device either display or printer.

We used GetDC functions for getting device context of a display device or printer device to output a graphics or text data. Printing or drawing can always be done through Device context provided by Windows. Whenever window needs to draw or paint in its client area it receives WM_PAINT message.

Painting in Windows

For painting we used an important message i.e. WM_PAINT. This message is always sent when windows need to paint its portion or region that was previously invalidated. Windows paint only its region when it become invalidates otherwise it is not sent WM_PAINT message. In some cases, WM_PAINT messages are not sent—when menu drops down or small dialog boxes appear.

Invalidation and Validation of a Region

If region is invalidated then it will be receiving WM_PAINT messages until it become validate.

Sending and Posting of messages

Messages can be sent to windows procedure directly or it can be posted to message queue. All the sent messages are not returned until the message is processed but the posted messages are returned after posting the message in queue.

Windows Management Functions

These functions are very helpful to interact with windows and hierarchy of windows and also with windows handling, windows manipulation and windows management.

Function includes GetParent, GetDlgItem, CreateWindow and notification codes that are sent to window by controls or other child windows.

Controls

Controls are normally considered are child windows, because these can be placed in any windows and become the part of the window but controls can be main window.

Notification messages

Notification messages are considered to transfer information to parent window by child windows. Child windows can send notification message to parent windows which aim only to inform about some events to parent window. The notification events could be e.g. in case of edit control is selection change i.e. EN_SELCHANGE or EN_CLICKED in case of button.

Popup Window

Popup window is not a child window. Popup windows are very useful when to show message on screen or working in full screen modes in Microsoft Windows Operating systems.

Keyboard

Keyboard is used to input the system. Whenever we press a key on keyboard, we generate a message. This message directly goes to the Operating system and then rout to our application. Keyboard messages include Key down and key up messages. Another type of messages is character messages these messages also come from keyboard. Keyboard messages are translated to their Character values and then send to the application in form of character message.

Mouse

Mouse is another input device. Almost all user interfaces uses mouse as input device as well as keyboard. Mouse device is optional in the system but useful in complex applications. Mouse can be used to point anywhere onscreen. Mouse sends different messages e.g. mouse can send left button down message when the mouse left button is down and in the same way left button up message when the left button is up. During the movement of mouse pointer on screen mouse move message Input Devices 22 is always sent.

Caret

During input session, caret is used to position the keyboard. Caret shows character can be placed where the caret is blinking.

Resources

Resources are also very much important subject in Windows executable files. Resources are separately compiled using resource compiler. Resource compiler (RC) compiles them to binary resource and these binary resource are then become the part of final executable file. Resource files are simply text script files. Resource can be loaded from any DLL and EXE module. For loading the resource, we have useful resource functions like LoadString that loads a string from resource table and Load Icon etc. that loads an Icon data from resource data.

Menus

Menus are used by almost every application except some games or other system tools. Using menus we can watch different facilities or action provided by application. Using menu accelerators you can use short cut keys to operate menus.

Dialogs

Dialogs are another useful and multipurpose resource in windows. Dialogs are used to display temporary information and other data.

Dialogs Types

Dialogs are of two types: One is modal dialogs and second is modeless dialogs. Modal dialogs do not return control to the application until they are not ended or destroyed. Modeless dialogs act like a normal windows they return control after they have created. Message boxes are normally modal dialog boxes.

Dialog Box Template

A dialog box template is binary data that describes the dialog box, defining its height, width, style, and the controls it contains. Dialog box template becomes later part of the executable file.

Dialogs are of two types: Modal dialogs and Modeless dialogs. For dispatching message for Modeless dialogs, we can use IsDialogMessage Function in our main Message Loop. In windows lot of common dialogs are available. Print dialog is used to print the document, this dialog show the setting for printer and its path name. Choose color dialog help the user to choose the color of its own choice. File Open dialog are useful to open and save the files on disk.

Windows Controls

Windows controls are basic controls that are pre-registered in windows. Button, list box and edit box controls are helpful to display information in a very organized manner in a dialog box or in a window. Edit box control is simple to use. It has few message and notifications messages. Sending message to edit box window we can limit text in edit box, set text and get text etc. Button is another ubiquitous control in windows. Button is used almost in every user interactive application. Button is sent messages like edit box and list box, and also send notification messages to its parent window. List View is another useful control in windows systems. List view control list the items in its window. These items can be selected and clicked on each click list box send notification message to its parent window.

Common Dialogs

Common dialogs are very much useful in windows. Using common dialogs, you can show user to choose colors, files and printer, etc. Dialog resources are easy to use and easier to handle.

Controls can be displayed on the dialogs. Dialogs by default set the font and dimensions of the controls. Dialogs are used in many areas like configuration of hardware devices, internet connections, properties and database configurations. Another important dialogs are called property sheets. This property sheet enables you to select any category from the tabs.

Common Controls

Common Controls are the part of Microsoft Windows Graphics Operating System. Almost all the WYSIWYG application use Common Controls for their compatibility and user friendliness with windows. Common controls include controls like page controls, tree controls, and list view controls that is modified from windows original control, button control that is also modified from windows original controls, data and time picker control, status bar, progress bar, rebar controls. These all controls reside in common controls library and the library has shipped with many versions. Before using the library you must check the valid version of the library because different version of library contains different controls properties.

Dynamic Link Libraries

Dynamic link libraries are the windows executables but these cannot be executed by writing its name on command line or double clicking on it. These libraries contain separate modules that load and run in any process address space. Thread is the execution unit in a Process. A process can have more than one thread. There are two types of dynamic linking load time dynamic

linking and run time dynamic linking. In load time dynamic linking, a module makes explicit calls to exported DLL functions as if they were local functions and in run time dynamic linking Load library function is used to load the library and Get procedure address functions are called to get the address of the function from loaded library. DLL can export functions in the form definition files. In definition file we can provide ordinal as well. Ordinal is a number that is used to locate the function instead of function name.

Multitasking Operating Systems

Multitasking Operating systems are useful to run applications simultaneously. Threads and processes are the key features of Operating systems. Many Threads can work better than using single thread sometime.

Threads and Synchronization

To synchronize access to a resource, use one of the synchronization objects in one of the wait functions. The state of a synchronization object is either signaled or nonsignaled. The wait functions allow a thread to block its own execution until a specified nonsignaled object is set to the signaled state. Critical section objects provide synchronization similar to that provided by mutex objects, except that critical section objects can be used only by the threads of a single process. Event, mutex, and semaphore objects can also be used in a single-process. Another synchronization object is semaphore, events and mutex. Threads with synchronization problems have the best use in network applications.

Socket

Socket is important in an inter-process communication. Sockets are more reliable and secure. Socket version 2 is used these days. In windows, sockets are started using WSStartup API.

WSAStartup API starts and initializes Windows Sockets.

DNS

Domain Name System (DNS), the locator service of choice in Microsoft® Windows®, is an industry-standard protocol that locates computers on an IP-based network.

WinSock Functions

These functions include connect, recv, send, accept, bind, gethostbyname, etc.

Whois server

This application tells that the name is registered name or not. If the name is registered, then we cannot register it again.

HTTP

HTTP (hypertext transfer protocol) is used to transfer text data across the net work.

HTML

HTML is hyper text markup language. It simplifies a text script. Html is loaded in web browser and web browser translates the text and executes instruction written in form of text.

MIME

MIME is used for transferring media like image data and movie data.

CS410- VISUAL PROGRAMMING

SOLVED SUBJECTIVE FOR FINAL EXAM

FALL SEMESTER 2012

QNo.1 Synchronization objects?

Answer:-

A synchronization object is an object whose handle can be specified in one of the wait functions to coordinate the execution of multiple threads. More than one process can have a handle to the same synchronization object, making interprocess synchronization possible.

REF :: handouts Page No. 343

QNo.2 Define owner drawn items?

Answer:-

Owner-drawn items require an application to take total responsibility for drawing selected (highlighted), selected, and cleared states.

REF :: handouts Page No. 236

QNo.3 Give two disadvantages of the DLL version

Answer:-

1. **Saves memory and reduces swapping.**

Many processes can use a single DLL simultaneously, sharing a single copy of the DLL in memory.

2. **Saves disk space.**

Many applications can share a single copy of the DLL on disk. In contrast, each application built with a static link library has the library code linked into its executable image as a separate copy.

REF :: handouts Page No. 332

<http://msdn.microsoft.com/en-us/library/dtba4t8b.aspx>

QNo.4 `int connect( SOCKET s, const struct sockaddr* name, int namelen);` Define the all parameters

Answer:-

Parameters

s:

Descriptor identifying an unconnected socket.

name:

Name of the socket in the **sockaddr** structure to which the connection should be established.

namelen:

Length of *name*, in bytes

Return Values:

If no error occurs, connect returns zero. Otherwise, it returns SOCKET_ERROR, and a specific error code can be retrieved by calling WSAGetLastError.

REF :: handouts Page No. 367

QNo.5 dynamic library that export DLL function "hello! You are here in DLL"

Answer:-

Say you want to build this minimal function in mydll.c:

```
#include <stdio.h>
int
hello()
{
    printf ("Hello World!\n");
}
```

First compile mydll.c to object code:

gcc -c mydll.c

Then, tell gcc that it is building a shared library:

gcc -shared -o mydll.dll mydll.o

you can now link to the dll with a simple program:

```
int
main ()
{
    hello ();
}
```

Then link to your dll with a command like:

gcc -o myprog myprog.c -L./ -lmydll

REF :: <http://cygwin.com/cygwin-ug-net/dll.html>

QNo.6 Modeless dialog why we cannot use the control in parent

Answer:-

The application displays the dialog box (if it is not already displayed) by using the ShowWindow function. The application destroys the dialog box by using the DestroyWindow function. To support keyboard navigation and other dialog box functionality, the message loop for the dialog box must call the IsDialogMessage function.

REF :: [handouts Page No. 267](#)

QNo.7 Socket connect ?

Answer:-

The connect () system call connects the socket referred to by the file descriptor sockfd to the address specified by addr

REF :: <http://linux.die.net/man/2/connect>

QNo.8 list box styles?

Answer:-

There are two types of list boxes:

1. single-selection
2. multiple-selection.

REF :: [handouts Page No. 287](#)

QNo.9 pointer arithmetic ?

Answer:-

Pointer Arithmetic deals with performing addition and subtraction operations on pointer variables.

- increment a pointer (++)
- decrement a pointer (--)
- Address in pointer is incremented or decremented by the size of the object it points to (char = 1 byte, int = 2 bytes, ...)

REF :: [handouts Page No. 21](#)

QNo.10 what is Bind Function?

Answer:-

The bind function is used on an unconnected socket before subsequent calls to connect or listen functions.

REF :: [handouts Page No. 364](#)

QNo.11 what is the purpose of Pager control?

Answer:-

A pager control is a window container that is used with a window that does not have enough display area to show all of its content.

REF :: [handouts Page No. 305](#)

QNo.12 what is the Mean by Resource definition Statement?

Answer:-

The resource-definition statements define the resources that the resource compiler puts in the resource (.Res) file. After the .Res file is linked to the executable file, the application can load its resources at run time as needed.

REF :: [handouts Page No. 213](#)

QNo.13 why static variables are not destroyed when function returns

Answer:-

The static variable is not destroyed on exit from the function; instead its value is preserved, and becomes available again when the function is next called.

REF :: http://www2.its.strath.ac.uk/courses/c/subsection3_6_4.html

QNo.14 Some usages of Dialog?

Answer:-

The following sections discuss how to use both types of dialog boxes.

1. Modal Dialog Boxes
2. Modeless Dialog Boxes

REF :: [handouts Page No. 243](#)

QNo.15 How Transfer-Encoding is implemented in Exchange-2000.

Answer:-

Routing group boundaries and SMTP target destinations determine how Exchange 2000 Server encodes mail. Exchange 2000 will encode as quoted-printable or 7-bit or Transport-Neutral Encapsulation Format (TNEF) when sending between two servers /recipients in different routing groups, and to the Internet.

Exchange 2000 Server will encode in Binary or Summary TNEF when sending to a recipient/server in the same routing group.

REF :: <http://support.microsoft.com/kb/836555>

QNo.16 what is the purpose of Hot Key?

Answer:-

A hot key is a key combination that generates a WM_HOTKEY message, a message the system places at the top of a thread's message queue, bypassing any existing messages in the queue

REF :: [handouts Page No. 199](#)

QNo.17 Unregister the Class function

Answer:-The process must destroy all windows using the class before the .dll is unloaded and call the UnregisterClass function.

REF :: [handouts Page No. 76](#)

QNo.18 What happen if we use DEFwinProc instead of dialog procedure in dialog.

Answer:-

if you call any function that results in your dialog procedure receiving a window message, the nested window message could overwrite the return value you set using DWL_MSGRESULT.

REF :: [handouts Page No. 130](#)

QNo.18 If you include CS_DBLCLKS in your window call style, the windows procedure receives which message for double click. (Specify exact sequence).

Answer:-

An application-defined window does not, by default, receive double-click messages. Because of the system overhead involved in generating double-click messages, these messages are generated only for windows belonging to classes that have the CS_DBLCLKS class style. Your application must set this style when registering the window class. For more information, see Window Classes.

REF :: [handouts Page No. 208](#)

QNo.19 Enumeration increase level of abstraction?

Answer:-

The enum declaration creates a new integer type. By convention the first letter of an enum type should be in uppercase. The list of values follows, where the first name is assigned zero, the second 1, etc.

[REF :: handouts Page No. 36](#)

QNo.20 Provide the Description of EDIT Control given below

Answer:-

EM_CANUNDO
EM_GETTEXTLIMIT
EM_GETHANDLE

EM_CANUNDO

Returns TRUE if the edit control operation can be undone.

EM_GETTEXTLIMIT

The size can be up to a predefined limit of approximately 32 kilobyte (KB) for single-line edit controls. Because this limit can change, it is called a soft limit

EM_GETHANDLE

Returns a handle identifying the buffer containing the multiline edit control's text. It is not processed by single-line edit controls.

[REF :: handouts Page No. 277](#)

QNo.21 What is mutex object?

Answer:-

The CreateMutex function creates or opens a named or unnamed mutex object.

[REF :: handouts Page No. 346](#)

QNo.22 how variables can share their resources across multiple process?

Answer:- Variables can be shared across multiple processes by making the separate data section as following.

```
#pragma data_seg( [ [ { push | pop }, ] [ identifier, ] ] [ "segment-name" [, "segment-class" ] ] )
```

Specifies the data segment where initialized variables are stored in the .obj file. OBJ files can be viewed with the dumpbin application. The default segment in the .obj file for initialized variables is .data. Variables initialized to zero are considered uninitialized and are stored in .bss.

[REF :: handouts Page No. 330](#)

QNo.23 Scope of variables in DLL file?

Answer:-

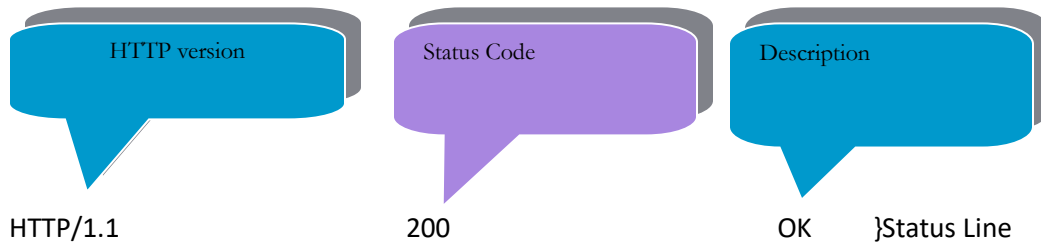
Variables defined in DLL have scope in memory until the DLL is loaded. After unloading, the variable scope is vanished. Locally defined variables are accessed within the DLL only. The variables that are set to export variables can be accessed outside the DLL if the DLL is statically linked.

[REF :: handouts Page No. 329](#)

QNo.24 the column was given about Status code of HTTP, Bad request, Request not found etc
we have to match the column according to the correct status code number

Answer:-

Request line is followed by 2 Carriage-Return /Line-feed sequences.



Content-type: text/html

Content-Length:2061

Headers delimited by CR/LF sequence

CRLF Actual data follows the headers

[REF :: handouts Page No. 382](#)

QNo.24 if variables declared as export?

Answer:-

Variables declared and initialized in a scope are deleted when execution leaves that scope

[REF :: handouts Page No. 66](#)

QNo.25 about keyboard accelerator (5)

Answer:-

A keyboard accelerator, also known as a shortcut key, is a keystroke or combination of keystrokes that generates a WM_COMMAND message. Keyboard accelerators are often used as shortcuts for commonly used menu commands, but you can also use them to generate commands that have no equivalent menu items. Include keyboard accelerators for any common or frequent actions, and provide support for the common shortcut keys where they apply.

[REF :: handouts Page No. 220](#)

QNo.26 Can macro takes two arguments. Explain it with example (5)

Answer:-

1. Function-like macros can take *arguments*, just like true functions. To define a macro that uses arguments.
2. you insert *parameters* between the pair of parentheses in the macro definition that make the macro function-like. The parameters must be valid C identifiers, separated by commas and optionally whitespace.

[REF :: handouts Page No. 47](#)

QNo.27 There are three types of assertion. write any two names

Answer:-

1. Preconditions

Specify conditions at the start of a function.

2. Post conditions

Specify conditions at the end of a function.

3. Invariants

Specify conditions over a defined region of a program.

[REF :: handouts Page No. 49](#)

QNo.28 Explain CGI

Answer:-

CGI is Common Gateway Interface. Win32 executable execute by the server. All browser request data is available at stdin (read using scanf() etc.) and all output sent to stdout (output using printf etc.) is sent to the browser instead of the server screen.

[REF :: handouts Page No. 396](#)

QNo.29 purpose of using of REbar control

Answer:-

Rebar controls act as containers for child windows. An application assigns child windows, which are often other controls, to a rebar control band.

[REF :: handouts Page No. 305](#)

QNo.30 two macro names that can message map functionality

1. `__FILE__`

This macro expands to the name of the current input file, in the form of a C string constant. This is the path by which the preprocessor opened the file, not the short names specified in #include or as the input file name argument.

2. `__LINE__`

This macro expands to the current input line number, in the form of a decimal integer constant. While we call it a predefined macro, it's a pretty strange macro, since its "definition" changes with each new line of source code.

[REF :: handouts Page No. 41](#)

QNo.31 Difference between shortcut key and system menu

Answer:-

Shortcut key

1. A keyboard accelerator, also known as a shortcut key
2. Include keyboard accelerators for any common or frequent actions, and provide support for the common shortcut keys where they apply.

System menu

1. The GetSystemMenu function allows the application to access the window menu (also known as the system menu)
2. Menu items on the window menu send WM_SYSCOMMAND messages

[REF :: handouts Page No. 241](#)

QNo.31 difference between send message () and post message ()

Answer:-

send message ()	post message ()
• SendMessage control is not returned to the calling application until the window that the message was sent to has completed processing the sent message, • calls the window procedure for the specified window and does not return until the window procedure has processed the message. • SendMessage is a synchronous function.	• PostMessage control is returned to the calling application immediately, regardless of whether or not the sent message has been processed. • Sends a message in the message queue associated with the thread and returns without waiting for the thread to process that message. • PostMessage is an asynchronous function

REF

http://wiki.answers.com/Q/What_is_the_difference_between_PostMessage_and_SendMessage

QNo.31 Two disadvantages of thread

Answer:-

1. Thread's major disadvantage is that they make the system slow because thread uses the time sharing concept that is another name multitasking.
2. A multitasking operating system divides the available processor time among the processes or threads that need it.

REF :: handouts Page No. 334

**QNo.32 in my application, we are enable enumeration variable to findResource but not found?
What I do for this behavior**

Answer:-

FindResource replacement and how to change control style more than once in Application. As deeper we're digging in Silverlight 2.0, we finding more and more WPF things and we're really missing in Silverlight. One of such things is FindResources.

In WPF I had Visual and Logical tree, so I was able to travel the tree to find resource I need. Let's see an example of the application. We have one resource defined in App level

Configuration Cod:-

```
<Application.Resources>
  <Style TargetType="Button" x:Key="green">
    <Setter Property="Background" Value="Green"/>
  </Style>
</Application.Resources>
Another resources are defined in different levels of Page
<UserControl.Resources>
  <Style TargetType="Button" x:Name="red">
    <Setter Property="Background" Value="Red"/>
  </Style>
</UserControl.Resources>
```

```

<Grid x:Name="LayoutRoot" Background="White">
  <Grid.Resources>
    <Style TargetType="Button" x:Name="blue">
      <Setter Property="Background" Value="Blue"/>
    </Style>
  </Grid.Resources>
  <Button Content="Click me" Click="Button_Click">
    <Button.Resources>
      <Style TargetType="Button" x:Name="yellow">
        <Setter Property="Background" Value="#FFFFFF00"/>
      </Style>
    </Button.Resources>
  </Button>
</Grid>

```

REF :: http://dedjo.blogspot.com/2008_03_01_archive.html

QNo.33 List box styles pointer arithmetic

Answer:-

List box styles

Window styles that control the appearance and operation of a list box

List box items are sorted, arranged in multiple columns, drawn by the application, and so on.

Pointer Arithmetic

Deal with performing addition and subtraction operations on pointer variables.

REF :: [handouts Page No. 21 ,287](#)

QNo.34 Define pager and list-view

Answer:-

Pager:-

A pager control is a window container that is used with a window that does not have enough display area to show all of its content

List-view

A list-view control is a window that displays a collection of items. The control provides several ways to arrange and display the items.

REF :: [handouts Page No. 305](#)

QNo.35 GetParent Syntax?

Answer:-

HWND GetParent

(

HWND hWnd // handle to child window

);

REF :: [handouts Page No. 186](#)

QNo.36 what is the purpose of Bind Function?

Answer:- The bind function associates a local address with a socket. The bind function is used on an unconnected socket before subsequent calls to connect or listen functions. It is used to bind to either connection-oriented (stream) or connectionless (datagram) sockets. When a socket is created with a call to the socket function, it exists in a namespace (address family), but it has no name assigned to it. Use the bind function to establish the local association of the socket by assigning a local name to an unnamed socket.

REF :: [handouts Page No. 377](#)

QNo.37 what is the purpose of List View Control?

Answer:-

1. List View is another useful control in windows systems.
2. List view control list the items in its window. These items can be selected and clicked on each click list box send notification message to its parent window.
3. List View control makes data binding easier than previous controls. It has included styling with CSS, flexible pagination, and sorting, inserting, deleting, and updating features.

REF :: [handouts Page No. 298](#)

QNo.38 what is a custom resource?

Answer:-

Resource is binary data that you can add to the executable file of a Windows-based application. A resource can be either standard or defined. The data in a standard resource describes an icon, cursor, menu, dialog box, bitmap, enhanced metafile, font, accelerator table, message-table entry, string-table entry, or version information. An application- defined resource, also called a custom resource, contains any data required by a specific application.

REF :: [handouts Page No. 213](#)

QNo.39 Differentiate between Simple Windows Programs & Dynamic Link Library programs?

Answer:-

A DLL (Dynamic Link Library) is a file that can be loaded and executed by programs dynamically. Basically it's an external code repository for programs. Since usually several different programs reuse the same DLL instead of having that code in their own file, this dramatically reduces required storage space. A synonym for a DLL would be library.

DLL does not have main function but exe has main function

Here DLL is in process component, both component and consumer will share same memory and Exe is out process component, it will run in its own memory.

An exe is an executable program. A DLL (Dynamic Link Library) is a file that can be loaded and executed by programs dynamically. Basically it's an external code repository for programs. Since usually several different programs reuse the same DLL instead of having that code in their own file, this dramatically reduces required storage space. A synonym for a DLL would be library.

DLL file it is usually the file contains information about programs. EXE is file which can be run and reading a information from DLL and other files to execute program.

REF :: http://wiki.answers.com/Q/What_is_the_difference_between_an_EXE_and_a_DLL

QNo.40 what are the three conditions of sending WM_Paint Message?

Answer:-

When a minimized window is maximized, Windows requests the application to repaint the client area. The WM_PAINT message is sent when the system or another application makes a request to paint a portion of an application's window. The message is sent when the UpdateWindow or RedrawWindow function is called, or by the DispatchMessage function when the application obtains a WM_PAINT message by using the GetMessage or PeekMessage function.

[REF :: handouts Page No. 65](#)

QNo.40 Define the names of resource-definition statements' categories?

Answer:-

The resource-definition statements can be divided into the following categories:

- Resources
- Controls
- Statements

[REF :: handouts Page No.214](#)

QNo.41 Write down any three parameter of send function?

1. The flags parameter can be used to influence the behavior of the function beyond the options specified for the associated socket.
2. Calling send with a zero len parameter is permissible and will be treated by implementations as successful
3. addr: Optional pointer to a buffer that receives the address of the connecting entity, as known to the communications layer.

[REF :: handouts Page No.383](#)

QNo.41 briefly explain Checkbox?**Check Boxes**

A check box consists of a square box and application-defined text (label), an icon, or a bitmap, that indicates a choice the user can make by selecting the button. Applications typically display check boxes in a group box to permit the user to choose from a set of related, but independent options. For example, an application might present a group of check boxes from which the user can select error conditions that produce warning beeps. A check box can be one of four styles: standard, automatic, three-state, and automatic three-state, as defined by the constants BS_CHECKBOX, BS_AUTOCHECKBOX, BS_3STATE, and BS_AUTO3STATE, respectively. Each style can assume two check states: checked (a check mark inside the box) or cleared (no check mark). In addition, a three-state check box can assume an indeterminate state (a grayed box inside the check box). Repeatedly clicking a standard or automatic check box toggles it from checked to cleared and back again. Repeatedly clicking a three-state check box toggles it from checked to cleared to indeterminate and back again. When the user clicks a check box (of any style), the check box receives the keyboard focus. The system sends the check box's parent window a WM_COMMAND message containing the BN_CLICKED notification code. The parent window doesn't acknowledge this message if it comes from an automatic check box or automatic three-state check box, because the system automatically sets the check state for those styles. But the parent window must acknowledge the message if it comes from a check box or three-state

check box because the parent window is responsible for setting the check state for those styles. Regardless of the check box style, the system automatically repaints the check box once its state is changed.

REF :: handouts Page No.282

QNo.42 Write down the basic socket operation.

Answer:-

The following are the basic operations performed by both server and client systems.

1. Create an unbound socket
 2. Binding Server
 3. Connecting Client
 4. Listen
 5. Accept
 6. Send
 7. Receive
- 27.7 Windows Socket Library

REF :: handouts Page No.358

QNo.43 What is meant by process 2marks

Answer:-

When you tell your computer to run a program, a new process is created which runs the code in that program. A process is an instance of a program

REF :: http://wiki.answers.com/Q/What_is_meant_by_process_in_computer_science

QNo.44 MIME meant?

Answer:-

MIME stands for Multi-purpose Internet Mail Extensions.

MIME contains encoding features, added to enable transfer of binary data, e.g. images (GIF, JPEG etc.) via mail. Using MIME encoding HTTP can now transfer complex binary data, e.g. images and video.

REF :: handouts Page No.380

QNo.45 MIME define "Resource Only DLLs?"

Answer:-

Resource Only DLL contains only resource of different language and local types. Resource only DLLs do not contain Entry Point or any DllMain Function.

Use of resource-only DLL is for internationalization

REF :: handouts Page No.322

QNo.46 How can I return multiple values from a function? Illustrate the concept using C++ code (5 marks)

you can only return 1 variable using "return". there is a way to do what you want though. basically you pass the address of the "two" and "five" variables.

example:

```
int n, two, five;
fcnName(n, &two, &five) //no need to return anything
{
    //your function code
}
```

QNo.47 If you include CS_DBLCLKS in your window call style, the windows procedure receives which message for double click. (Specify exact sequence)

Answer:-

CS_DBLCLKS class style. Your application must set this style when registering the window class. For more information, see Window Classes.

A double-click message is always the third message in a four-message series. The first two messages are the button-down and button-up messages generated by the first click. The second click generates the double-click message followed by another button-up message. For example, double-clicking the left mouse button generates the following message sequence:

REF :: [handouts Page No.208](#)

QNo.48 How do I restrict my window so it can't be resized larger or smaller than a certain size?

Answer:-

You restrict your window's size by handling the WM_GETMINMAXINFO message. Your handler receives a pointer to a MINMAXINFO struct, which you fill in with the minimum and/or maximum size for your window. Here is an example that keeps the window between 100x150 and 600x400 pixels in size.

☐ Collapse | [Copy Code](#)

```
LRESULT OnGetMinMaxInfo ( WPARAM wParam, LPARAM lParam )
```

```
{  
    MINMAXINFO* pmmi = (MINMAXINFO*) lParam;
```

```
    pmmi->ptMinTrackSize.x = 100;  
    pmmi->ptMinTrackSize.y = 150;  
    pmmi->ptMaxTrackSize.x = 600;  
    pmmi->ptMaxTrackSize.y = 400;
```

```
    return 0;  
}
```

In MFC, your OnGetMinMaxInfo() handler is passed a MINMAXINFO* directly, but otherwise the code is the same.

REF :: http://www.codeproject.com/Articles/1248/The-Code-Project-Visual-C-Forum-FAQ#ui_minmaxinfo

CS410- VISUAL PROGRAMMING

SOLVED SUBJECTIVE FOR FINAL EXAM

FALL SEMESTER 2012

QNo.1 Synchronization objects?

Answer:-

A synchronization object is an object whose handle can be specified in one of the wait functions to coordinate the execution of multiple threads. More than one process can have a handle to the same synchronization object, making interprocess synchronization possible.

REF :: handouts Page No. 343

QNo.2 Define owner drawn items?

Answer:-

Owner-drawn items require an application to take total responsibility for drawing selected (highlighted), selected, and cleared states.

REF :: handouts Page No. 236

QNo.3 Give two disadvantages of the DLL version

Answer:-

1. Saves memory and reduces swapping.

Many processes can use a single DLL simultaneously, sharing a single copy of the DLL in memory.

2. Saves disk space.

Many applications can share a single copy of the DLL on disk. In contrast, each application built with a static link library has the library code linked into its executable image as a separate copy.

REF :: handouts Page No. 332

<http://msdn.microsoft.com/en-us/library/dtba4t8b.aspx>

QNo.4 `int connect( SOCKET s, const struct sockaddr* name, int namelen);` Define the all parameters

Answer:-

Parameters

s:

Descriptor identifying an unconnected socket.

name:

Name of the socket in the **sockaddr** structure to which the connection should be established.

namelen:

Length of *name*, in bytes

Return Values:

If no error occurs, connect returns zero. Otherwise, it returns SOCKET_ERROR, and a specific error code can be retrieved by calling WSAGetLastError.

REF :: handouts Page No. 367

QNo.5 dynamic library that export DLL function "hello! You are here in DLL"

Answer:-

Say you want to build this minimal function in mydll.c:

```
#include <stdio.h>
int
hello()
{
    printf ("Hello World!\n");
}
```

First compile mydll.c to object code:

`gcc -c mydll.c`

Then, tell gcc that it is building a shared library:

`gcc -shared -o mydll.dll mydll.o`

you can now link to the dll with a simple program:

```
int
main ()
{
    hello ();
}
```

Then link to your dll with a command like:

`gcc -o myprog myprog.c -L./ -lmydll`

REF :: <http://cygwin.com/cygwin-ug-net/dll.html>

QNo.6 Modeless dialog why we cannot use the control in parent

Answer:-

The application displays the dialog box (if it is not already displayed) by using the ShowWindow function. The application destroys the dialog box by using the DestroyWindow function. To support keyboard navigation and other dialog box functionality, the message loop for the dialog box must call the IsDialogMessage function.

REF :: [handouts Page No. 267](#)

QNo.7 Socket connect ?

Answer:-

The connect () system call connects the socket referred to by the file descriptor sockfd to the address specified by addr

REF :: <http://linux.die.net/man/2/connect>

QNo.8 list box styles?

Answer:-

There are two types of list boxes:

1. single-selection
2. multiple-selection.

REF :: [handouts Page No. 287](#)

QNo.9 pointer arithmetic ?

Answer:-

Pointer Arithmetic deals with performing addition and subtraction operations on pointer variables.

- increment a pointer (++)
- decrement a pointer (--)
- Address in pointer is incremented or decremented by the size of the object it points to (char = 1 byte, int = 2 bytes, ...)

REF :: [handouts Page No. 21](#)

QNo.10 what is Bind Function?

Answer:-

The bind function is used on an unconnected socket before subsequent calls to connect or listen functions.

REF :: [handouts Page No. 364](#)

QNo.11 what is the purpose of Pager control?

Answer:-

A pager control is a window container that is used with a window that does not have enough display area to show all of its content.

REF :: [handouts Page No. 305](#)

QNo.12 what is the Mean by Resource definition Statement?

Answer:-

The resource-definition statements define the resources that the resource compiler puts in the resource (.Res) file. After the .Res file is linked to the executable file, the application can load its resources at run time as needed.

REF :: [handouts Page No. 213](#)

QNo.13 why static variables are not destroyed when function returns

Answer:-

The static variable is not destroyed on exit from the function; instead its value is preserved, and becomes available again when the function is next called.

REF :: http://www2.its.strath.ac.uk/courses/c/subsection3_6_4.html

QNo.14 Some usages of Dialog?

Answer:-

The following sections discuss how to use both types of dialog boxes.

1. Modal Dialog Boxes
2. Modeless Dialog Boxes

REF :: [handouts Page No. 243](#)

QNo.15 How Transfer-Encoding is implemented in Exchange-2000.

Answer:-

Routing group boundaries and SMTP target destinations determine how Exchange 2000 Server encodes mail. Exchange 2000 will encode as quoted-printable or 7-bit or Transport-Neutral Encapsulation Format (TNEF) when sending between two servers /recipients in different routing groups, and to the Internet.

Exchange 2000 Server will encode in Binary or Summary TNEF when sending to a recipient/server in the same routing group.

REF :: <http://support.microsoft.com/kb/836555>

QNo.16 what is the purpose of Hot Key?

Answer:-

A hot key is a key combination that generates a WM_HOTKEY message, a message the system places at the top of a thread's message queue, bypassing any existing messages in the queue

REF :: [handouts Page No. 199](#)

QNo.17 Unregister the Class function

Answer:-The process must destroy all windows using the class before the .dll is unloaded and call the UnregisterClass function.

REF :: [handouts Page No. 76](#)

QNo.18 What happen if we use DEFwinProc instead of dialog procedure in dialog.

Answer:-

if you call any function that results in your dialog procedure receiving a window message, the nested window message could overwrite the return value you set using DWL_MSGRESULT.

REF :: [handouts Page No. 130](#)

QNo.18 If you include CS_DBLCLKS in your window call style, the windows procedure receives which message for double click. (Specify exact sequence).

Answer:-

An application-defined window does not, by default, receive double-click messages. Because of the system overhead involved in generating double-click messages, these messages are generated only for windows belonging to classes that have the CS_DBLCLKS class style. Your application must set this style when registering the window class. For more information, see Window Classes.

REF :: [handouts Page No. 208](#)

QNo.19 Enumeration increase level of abstraction?

Answer:-

The enum declaration creates a new integer type. By convention the first letter of an enum type should be in uppercase. The list of values follows, where the first name is assigned zero, the second 1, etc.

[REF :: handouts Page No. 36](#)

QNo.20 Provide the Description of EDIT Control given below

Answer:-

EM_CANUNDO
EM_GETTEXTLIMIT
EM_GETHANDLE

EM_CANUNDO

Returns TRUE if the edit control operation can be undone.

EM_GETTEXTLIMIT

The size can be up to a predefined limit of approximately 32 kilobyte (KB) for single-line edit controls. Because this limit can change, it is called a soft limit

EM_GETHANDLE

Returns a handle identifying the buffer containing the multiline edit control's text. It is not processed by single-line edit controls.

[REF :: handouts Page No. 277](#)

QNo.21 What is mutex object?

Answer:-

The CreateMutex function creates or opens a named or unnamed mutex object.

[REF :: handouts Page No. 346](#)

QNo.22 how variables can share their resources across multiple process?

Answer:-Variables can be shared across multiple processes by making the separate data section as following.

```
#pragma data_seg( [ [ { push | pop }, ] [ identifier, ] ] [ "segment-name" [, "segment-class" ] ] )
```

Specifies the data segment where initialized variables are stored in the .obj file. OBJ files can be viewed with the dumpbin application. The default segment in the .obj file for initialized variables is .data. Variables initialized to zero are considered uninitialized and are stored in .bss.

[REF :: handouts Page No. 330](#)

QNo.23 Scope of variables in DLL file?

Answer:-

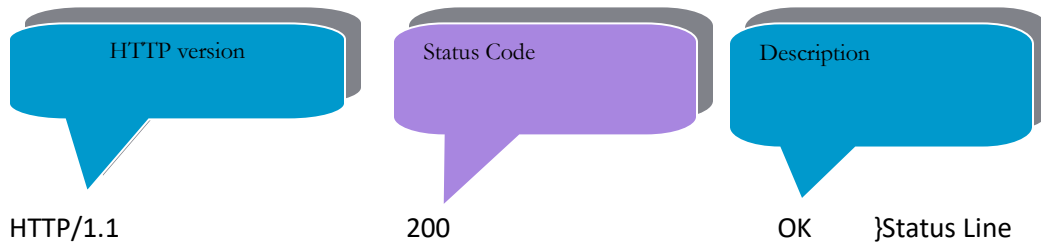
Variables defined in DLL have scope in memory until the DLL is loaded. After unloading, the variable scope is vanished. Locally defined variables are accessed within the DLL only. The variables that are set to export variables can be accessed outside the DLL if the DLL is statically linked.

[REF :: handouts Page No. 329](#)

QNo.24 the column was given about Status code of HTTP, Bad request, Request not found etc
we have to match the column according to the correct status code number

Answer:-

Request line is followed by 2 Carriage-Return /Line-feed sequences.



Content-type: text/html

Content-Length:2061

Headers delimited by CR/LF sequence

CRLF Actual data follows the headers

[REF :: handouts Page No. 382](#)

QNo.24 if variables declared as export?

Answer:-

Variables declared and initialized in a scope are deleted when execution leaves that scope

[REF :: handouts Page No. 66](#)

QNo.25 about keyboard accelerator (5)

Answer:-

A keyboard accelerator, also known as a shortcut key, is a keystroke or combination of keystrokes that generates a WM_COMMAND message. Keyboard accelerators are often used as shortcuts for commonly used menu commands, but you can also use them to generate commands that have no equivalent menu items. Include keyboard accelerators for any common or frequent actions, and provide support for the common shortcut keys where they apply.

[REF :: handouts Page No. 220](#)

QNo.26 Can macro takes two arguments. Explain it with example (5)

Answer:-

1. Function-like macros can take *arguments*, just like true functions. To define a macro that uses arguments.
2. you insert *parameters* between the pair of parentheses in the macro definition that make the macro function-like. The parameters must be valid C identifiers, separated by commas and optionally whitespace.

[REF :: handouts Page No. 47](#)

QNo.27 There are three types of assertion. write any two names

Answer:-

1. Preconditions

Specify conditions at the start of a function.

2. Post conditions

Specify conditions at the end of a function.

3. Invariants

Specify conditions over a defined region of a program.

[REF :: handouts Page No. 49](#)

QNo.28 Explain CGI

Answer:-

CGI is Common Gateway Interface. Win32 executable execute by the server. All browser request data is available at stdin (read using scanf() etc.) and all output sent to stdout (output using printf etc.) is sent to the browser instead of the server screen.

[REF :: handouts Page No. 396](#)

QNo.29 purpose of using of REbar control

Answer:-

Rebar controls act as containers for child windows. An application assigns child windows, which are often other controls, to a rebar control band.

[REF :: handouts Page No. 305](#)

QNo.30 two macro names that can message map functionality

1. `__FILE__`

This macro expands to the name of the current input file, in the form of a C string constant. This is the path by which the preprocessor opened the file, not the short names specified in #include or as the input file name argument.

2. `__LINE__`

This macro expands to the current input line number, in the form of a decimal integer constant. While we call it a predefined macro, it's a pretty strange macro, since its "definition" changes with each new line of source code.

[REF :: handouts Page No. 41](#)

QNo.31 Difference between shortcut key and system menu

Answer:-

Shortcut key

1. A keyboard accelerator, also known as a shortcut key
2. Include keyboard accelerators for any common or frequent actions, and provide support for the common shortcut keys where they apply.

System menu

1. The GetSystemMenu function allows the application to access the window menu (also known as the system menu)
2. Menu items on the window menu send WM_SYSCOMMAND messages

[REF :: handouts Page No. 241](#)

QNo.31 difference between send message () and post message ()**Answer:-**

send message ()	post message ()
• SendMessage control is not returned to the calling application until the window that the message was sent to has completed processing the sent message, • calls the window procedure for the specified window and does not return until the window procedure has processed the message. • SendMessage is a synchronous function.	• PostMessage control is returned to the calling application immediately, regardless of whether or not the sent message has been processed. • Sends a message in the message queue associated with the thread and returns without waiting for the thread to process that message. • PostMessage is an asynchronous function

REF

http://wiki.answers.com/Q/What_is_the_difference_between_PostMessage_and_SendMessage

QNo.31 Two disadvantages of thread**Answer:-**

1. Threads major disadvantage is that they make the system slow because thread uses the time sharing concept that is another name multitasking.
2. A multitasking operating system divides the available processor time among the processes or threads that need it.

REF :: handouts Page No. 334
**QNo.32 in my application, we are enable enumeration variable to findResource but not found?
What I do for this behavior**
Answer:-

FindResource replacement and how to change control style more than once in Application. As deeper we're digging in Silverlight 2.0, we finding more and more WPF things and we're really missing in Silverlight. One of such things is FindResources.

In WPF I had Visual and Logical tree, so I was able to travel the tree to find resource I need. Let's see an example of the application. We have one resource defined in App level

Configuration Cod:-

```

<Application.Resources>
  <Style TargetType="Button" x:Key="green">
    <Setter Property="Background" Value="Green"/>
  </Style>
</Application.Resources>

```

Another resources are defined in different levels of Page

```

<UserControl.Resources>
  <Style TargetType="Button" x:Name="red">
    <Setter Property="Background" Value="Red"/>
  </Style>
</UserControl.Resources>

```

```

<Grid x:Name="LayoutRoot" Background="White">
  <Grid.Resources>
    <Style TargetType="Button" x:Name="blue">
      <Setter Property="Background" Value="Blue"/>
    </Style>
  </Grid.Resources>
  <Button Content="Click me" Click="Button_Click">
    <Button.Resources>
      <Style TargetType="Button" x:Name="yellow">
        <Setter Property="Background" Value="#FFFFFF00"/>
      </Style>
    </Button.Resources>
  </Button>
</Grid>

```

REF :: http://dedjo.blogspot.com/2008_03_01_archive.html

QNo.33 List box styles pointer arithmetic

Answer:-

List box styles

Window styles that control the appearance and operation of a list box

List box items are sorted, arranged in multiple columns, drawn by the application, and so on.

Pointer Arithmetic

Deal with performing addition and subtraction operations on pointer variables.

REF :: [handouts Page No. 21 ,287](#)

QNo.34 Define pager and list-view

Answer:-

Pager:-

A pager control is a window container that is used with a window that does not have enough display area to show all of its content

List-view

A list-view control is a window that displays a collection of items. The control provides several ways to arrange and display the items.

REF :: [handouts Page No. 305](#)

QNo.35 GetParent Syntax?

Answer:-

HWND GetParent

(

HWND hWnd // handle to child window

);

REF :: [handouts Page No. 186](#)

QNo.36 what is the purpose of Bind Function?

Answer:- The bind function associates a local address with a socket. The bind function is used on an unconnected socket before subsequent calls to connect or listen functions. It is used to bind to either connection-oriented (stream) or connectionless (datagram) sockets. When a socket is created with a call to the socket function, it exists in a namespace (address family), but it has no name assigned to it. Use the bind function to establish the local association of the socket by assigning a local name to an unnamed socket.

REF :: [handouts Page No. 377](#)

QNo.37 what is the purpose of List View Control?

Answer:-

1. List View is another useful control in windows systems.
2. List view control list the items in its window. These items can be selected and clicked on each click list box send notification message to its parent window.
3. List View control makes data binding easier than previous controls. It has included styling with CSS, flexible pagination, and sorting, inserting, deleting, and updating features.

REF :: [handouts Page No. 298](#)

QNo.38 what is a custom resource?

Answer:-

Resource is binary data that you can add to the executable file of a Windows-based application. A resource can be either standard or defined. The data in a standard resource describes an icon, cursor, menu, dialog box, bitmap, enhanced metafile, font, accelerator table, message-table entry, string-table entry, or version information. An application- defined resource, also called a custom resource, contains any data required by a specific application.

REF :: [handouts Page No. 213](#)

QNo.39 Differentiate between Simple Windows Programs & Dynamic Link Library programs?

Answer:-

A DLL (Dynamic Link Library) is a file that can be loaded and executed by programs dynamically. Basically it's an external code repository for programs. Since usually several different programs reuse the same DLL instead of having that code in their own file, this dramatically reduces required storage space. A synonym for a DLL would be library.

DLL does not have main function but exe has main function

Here DLL is in process component, both component and consumer will share same memory and Exe is out process component, it will run in its own memory.

An exe is an executable program. A DLL (Dynamic Link Library) is a file that can be loaded and executed by programs dynamically. Basically it's an external code repository for programs. Since usually several different programs reuse the same DLL instead of having that code in their own file, this dramatically reduces required storage space. A synonym for a DLL would be library.

DLL file it is usually the file contains information about programs. EXE is file which can be run and reading a information from DLL and other files to execute program.

REF :: http://wiki.answers.com/Q/What_is_the_difference_between_an_EXE_and_a_DLL

QNo.40 what are the three conditions of sending WM_Paint Message?

Answer:-

When a minimized window is maximized, Windows requests the application to repaint the client area. The WM_PAINT message is sent when the system or another application makes a request to paint a portion of an application's window. The message is sent when the UpdateWindow or RedrawWindow function is called, or by the DispatchMessage function when the application obtains a WM_PAINT message by using the GetMessage or PeekMessage function.

[REF :: handouts Page No. 65](#)

QNo.40 Define the names of resource-definition statements' categories?

Answer:-

The resource-definition statements can be divided into the following categories:

- Resources
- Controls
- Statements

[REF :: handouts Page No.214](#)

QNo.41 Write down any three parameter of send function?

1. The flags parameter can be used to influence the behavior of the function beyond the options specified for the associated socket.
2. Calling send with a zero len parameter is permissible and will be treated by implementations as successful
3. addr: Optional pointer to a buffer that receives the address of the connecting entity, as known to the communications layer.

[REF :: handouts Page No.383](#)

QNo.41 briefly explain Checkbox?**Check Boxes**

A check box consists of a square box and application-defined text (label), an icon, or a bitmap, that indicates a choice the user can make by selecting the button. Applications typically display check boxes in a group box to permit the user to choose from a set of related, but independent options. For example, an application might present a group of check boxes from which the user can select error conditions that produce warning beeps. A check box can be one of four styles: standard, automatic, three-state, and automatic three-state, as defined by the constants BS_CHECKBOX, BS_AUTOCHECKBOX, BS_3STATE, and BS_AUTO3STATE, respectively. Each style can assume two check states: checked (a check mark inside the box) or cleared (no check mark). In addition, a three-state check box can assume an indeterminate state (a grayed box inside the check box). Repeatedly clicking a standard or automatic check box toggles it from checked to cleared and back again. Repeatedly clicking a three-state check box toggles it from checked to cleared to indeterminate and back again. When the user clicks a check box (of any style), the check box receives the keyboard focus. The system sends the check box's parent window a WM_COMMAND message containing the BN_CLICKED notification code. The parent window doesn't acknowledge this message if it comes from an automatic check box or automatic three-state check box, because the system automatically sets the check state for those styles. But the parent window must acknowledge the message if it comes from a check box or three-state

check box because the parent window is responsible for setting the check state for those styles. Regardless of the check box style, the system automatically repaints the check box once its state is changed.

REF :: handouts Page No.282

QNo.42 Write down the basic socket operation.

Answer:-

The following are the basic operations performed by both server and client systems.

1. Create an unbound socket
 2. Binding Server
 3. Connecting Client
 4. Listen
 5. Accept
 6. Send
 7. Receive
- 27.7 Windows Socket Library

REF :: handouts Page No.358

QNo.43 What is meant by process 2marks

Answer:-

When you tell your computer to run a program, a new process is created which runs the code in that program. A process is an instance of a program

REF :: http://wiki.answers.com/Q/What_is_meant_by_process_in_computer_science

QNo.44 MIME meant?

Answer:-

MIME stands for Multi-purpose Internet Mail Extensions.

MIME contains encoding features, added to enable transfer of binary data, e.g. images (GIF, JPEG etc.) via mail. Using MIME encoding HTTP can now transfer complex binary data, e.g. images and video.

REF :: handouts Page No.380

QNo.45 MIME define "Resource Only DLLs?"

Answer:-

Resource Only DLL contains only resource of different language and local types. Resource only DLLs do not contain Entry Point or any DllMain Function.

Use of resource-only DLL is for internationalization

REF :: handouts Page No.322

QNo.46 How can I return multiple values from a function? Illustrate the concept using C++ code (5 marks)

you can only return 1 variable using "return". there is a way to do what you want though. basically you pass the address of the "two" and "five" variables.

example:

```
int n, two, five;
fcnName(n, &two, &five) //no need to return anything
{
    //your function code
}
```

QNo.47 If you include CS_DBLCLKS in your window call style, the windows procedure receives which message for double click. (Specify exact sequence)

Answer:-

CS_DBLCLKS class style. Your application must set this style when registering the window class. For more information, see Window Classes.

A double-click message is always the third message in a four-message series. The first two messages are the button-down and button-up messages generated by the first click. The second click generates the double-click message followed by another button-up message. For example, double-clicking the left mouse button generates the following message sequence:

REF :: [handouts Page No.208](#)

QNo.48 How do I restrict my window so it can't be resized larger or smaller than a certain size?

Answer:-

You restrict your window's size by handling the WM_GETMINMAXINFO message. Your handler receives a pointer to a MINMAXINFO struct, which you fill in with the minimum and/or maximum size for your window. Here is an example that keeps the window between 100x150 and 600x400 pixels in size.

☐ Collapse | [Copy Code](#)

```
LRESULT OnGetMinMaxInfo ( WPARAM wParam, LPARAM lParam )
```

```
{  
    MINMAXINFO* pmmi = (MINMAXINFO*) lParam;
```

```
    pmmi->ptMinTrackSize.x = 100;  
    pmmi->ptMinTrackSize.y = 150;  
    pmmi->ptMaxTrackSize.x = 600;  
    pmmi->ptMaxTrackSize.y = 400;
```

```
    return 0;  
}
```

In MFC, your OnGetMinMaxInfo() handler is passed a MINMAXINFO* directly, but otherwise the code is the same.

REF :: http://www.codeproject.com/Articles/1248/The-Code-Project-Visual-C-Forum-FAQ#ui_minmaxinfo