Design and Realization of Computer Testing System for VC Programming

Zhou Min, Yao Li

Abstract

In order to enhance students’ mastery of VC programming and improve their capacity to solve practical problems, computer testing system and computer-aided teaching method are applied in the course of VC programming. This paper gives a brief introduction to progresses and problems in designing the VC programming testing system.

Keyword: computer-aided teaching system; computer testing system; practice system

1. Introduction

VC programming is enjoying increasing popularity among programmers with its multi-function, expressiveness, flexibility, wide-application, high-efficient target program, and many other features. Consequently, many colleges and universities have started to offer courses of “VC Programming” for non-computer majors, hoping to facilitate students with one development tool, the design method of structured program, and the object-oriented programming method. In most cases, this is the first time for the students to learn a programming language, so most of them feel at a loss when programming and the course is too difficult to arouse their interest. In the mean time, students show an inclination to attach greater attention to theoretical study over practical exercises. To solve this problem, not only the teaching methods, but also the testing method should be improved. That is why we have developed the computer-aided teaching system, computer-based practice system, and the computer testing system for VC programming.
2. Composition of computer testing system

This Computer Testing System is designed to measure students’ mastery of VC programming language, and is used in mid-term exams, finals, as well as mock exams and daily practices. This system of client/server structure uses UDP protocol and Winsock offered by Visual Basic to realize the data transfer of login information, question selection, and test scores. The functions of each part will be discussed in the following paragraphs.

2.1. Server

The function of server is to offer testing item pool, to give out and collect testing papers, and to monitor the whole process.

There are four types of questions saved in the server, namely, multiple-choice, blank-filling, error-correcting, and programming. Questions for formal tests and mock tests are all saved here, but separately.

On the server side, teachers can appoint exam room, monitor the testing process, and the monitor will display the current name list of examinees and examinees who have submitted the paper. The monitoring interface is shown as figure 1.

When a student log in on a client computer, the server can identify it is in the appointed exam room or not with its IP address. If it is from the exam room, a test paper randomly composed of questions in the item pool will be sent to the client, and the student’s score will be saved in the server’s database when submission. If it is not from the exam room, a mock exam paper will be sent instead.

![Monitoring interface on the server.](image-url)
1. The code for IP identifying is:

```vba
Dim ps, zsks As String
Winsock1.GetData ps, vbString 'ps computer number sent by the client
Data5.Refresh
Data5.Recordset.FindFirst "Address='" & ps & "'
computerno = Mid(Data5.Recordset("Computer Number"), 2, 1)
If InStr(jifang, computerno) <> 0 Then
    zsks = "T" 'Formal test
Else
    zsks = "F" 'Mock exam
End If
```

2. The code for question selection(multiple choice) is:

```vba
Randomize 'Random selection of ten multiple choices
q1 = Int(Rnd() * Data1.Recordset.RecordCount / 10) + 1
qq = "trim(tq)=' " & Trim(Str(q1)) & "' " 'tqq1 Question Number
Data1.Recordset.FindFirst qq
For i = 1 To 10
    xzt(i, 1) = Data1.Recordset("ti")
    xzt(i, 2) = Data1.Recordset("A")
    xzt(i, 3) = Data1.Recordset("B")
    xzt(i, 4) = Data1.Recordset("C")
    xzt(i, 5) = Data1.Recordset("D")
    xzt(i, 6) = Data1.Recordset("da")
Data1.Recordset.FindNext qq
```

2.2. Client (student)

The function of the client computer (the student computer) is to accomplish the login and the testing paper. When the testing paper is generated, students can choose the four question type on the corresponding running interface, namely, multiple-choice, blank-filling, error-correcting, and programming. When the paper is finished, students should click “submit”. The paper will be automatically scored by the client, and the results will be transferred to the backstage supporter’s database and saved there. The interface of students’ mock exam is shown as Figure 2.
Fig. 2. Interface on the client/student computer

2.3. Practice and mock exam system

To encourage and give immediate feedback to students’ practices, the system will show the results and the answer key on the screen after the mock exam for reference.

3. Technological investigation

Sharing the common features of other computer testing and practicing system, this system has been improved in the two aspects, i.e. time limiting and progress managing.

3.1. Limiting the testing time

When doing the computer test, students are also required to finish it within the limited time. At present, most testing systems read the submitting time by adding the appointed testing time to the login time. A timer is used to check the examinee’s system time during the test, and if the time boundary is reached, the paper will be automatically scored by the computer, the result will be transferred to the server, and the testing system closed. There is one potential problem in such a system, which is if the examinee changed the system time, the testing time of this examinee will be prolonged and unfairness will occur. To avoid this, we have designed a form with a timer whose interval attribute is 1000 millisecond. This time also claims a static variable “n” to measure the time used in the test. The timer starts as soon as an examinee log in and the submitting time will be measured by the DateAdd Function. The start time and the time remaining will be shown on the screen according to the value of variable “n”. 2 minutes before the submitting time, the system will remind examinees to save the answered questions in time and submit on time. This improvement can free the testing system from the control of system time. The code of the timer is as the following:

Private Sub Timer1_Timer()
    Static n As Long  ‘n refers to the seconds
    n = n + 1
"
Label8.Caption = Format(DateAdd("s", -n, Endtime - stime), "h:mm:ss")
'The remaining time will be shown; Endtime refers to the submitting time; stime refers to the login time

If n >= elapsed * 60 Then
    Unload me
    Unload MDIForm1
    Submit the score.Show
    Submit the score.ZOrder
End If
End Sub

3.2. Managing the process

For the convenience of the examinees, there is a menu-item to start the Microsoft Visual c++ when they are to do the programming. However, examinees will restart the VC program when confronted with programming difficulties, thus many VC will be saved in memory occupying too much resource and slowing down the running velocity and causing other mistakes. To avoid this, once the VC menu-item is clicked, a corresponding program will be started which forbids the restart of VC and display "The VC window is already in progress. Please do not open another." This is in fact a matter of process management. The code is shown as the following. (Please refer to the notes when some statements is not easy to understand.)

Dim flag As Boolean
Dim theloop As Long, ret
Dim proc As PROCESSENTRY32
Dim snap As Long
Dim exename As String
flag = True
snap = CreateToolhelpSnapshot(TH32CS_SNAPall, 0) 'Obtain the “snapshot” of the process
proc.dwSize = Len(proc)
theloop = ProcessFirst(snap, proc) 'Obtain the first process and its return value
Do While theloop <> 0 'When the return value is not ZERO, the nest process starts
    exename = proc.szExeFile
    If InStr(exename, "MSDEV.EXE") <> 0 Then
        x = MsgBox("The VC window is already in progress. Please do not open another!", 48, "Information Cue")
        flag = False
        Exit Do
    End If
    proc.szExeFile = "                  "
    theloop = ProcessNext(snap, proc)
Loop
CloseHandle snap 'Close the “snapshot”.
If flag = True Then
    i = Shell("C:\Microsoft Visualudio\Common\MSDev98\Bin\MSDEV.EXE", 1)
End If
4. Conclusion

This Computer Testing System has been used in the course of VC Programming for non-computer majors since 2006. This system has a friendly interface and high level security. It also has a large and open-ended question database covering all the key and difficult points of the course. More importantly, it enables a large population to do the test at the same time. Thanks to all this advantages, it has played a significant role in improving students’ ability of in actual application and enhancing excellence in teaching.

Acknowledgements

I would like to take this opportunity to express my heartfelt gratitude to those who have supported me during this study.

Deep gratitude is owed to my collaborator Yao Li, who has spent much of her precious time on this paper. Had she not devoted her painstaking efforts to reading carefully each draft and giving valuable suggestions, the completion of this thesis would have been impossible.

I would also like to extend my sincere gratitude and genuine respect to the authors whose words I have cited or quoted and whose researches are used as my references.

Last, my thanks would go to my family members, who have always given me their consistent support and consolation in times of joy and times of sorrow of both my research and personal life.

References