Quiz 2 (Close book, 1 A4 cheat sheet, 45 minutes)
Solution

Student ID__________________Name (Print)_________________________________

Your paper will not be graded unless you endorse the following statement:
I have neither given nor received inappropriate assistance on this quiz.

Signature_______________________________________________________________

<table>
<thead>
<tr>
<th>Items</th>
<th>Points</th>
<th>Scores</th>
</tr>
</thead>
<tbody>
<tr>
<td>Problem 1</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 2</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 3</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 4</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 5</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 6</td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>Problem 7</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Problem 8</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Problem 9</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Problem 10</td>
<td>10</td>
<td></td>
</tr>
<tr>
<td>Problem 11</td>
<td>30</td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>100</td>
<td></td>
</tr>
</tbody>
</table>
1. (5pts) Circle the proper relational logical expression in Fortran 95/2003, assuming all variables are declared as integer type
   A. a == ‘2’
   B. a .and. b
   C. a /= b*c
   D. a

2. (5pts) Circle the proper IF-statement in Fortran 95/2003, assuming all variables are declared as real type
   A. IF ( index <= 3 )
   B. IF ( temp > 95.5 .and. temp <= 99.5 ) THEN
   C. IF .not. total <= 0.0 THEN
   D. IF ( abs(a) > 1.0 ) THEN a = 1.0

3. (5pts) Circle the proper number of iterations defined by the following counting loop DO statement in Fortran 95/2003
   DO index = 2, 10, 3
   A. 3
   B. 4
   C. 5
   D. 6

4. (5pts) Circle the case-selector expression that picks up integer values of 10 and above:
   A. case ( 10:inf )
   B. case ( x > 10 )
   C. case ( 10: )
   D. case ( 10:100 )

5. (5pts) Circle the proper value of variable y after the execution of the following partial Fortran 95/2003 source code (assuming index is declared as integer):
   
   y = 0
   DO index = 1,5
       y = y + index * 2
   END DO
   
   A. 15
   B. 20
   C. 25
   D. 30
6. (5pts) Circle the proper control construct that can implement the following mathematical formula

\[ f(x) = \begin{cases} 
0 & \text{if } x \leq 0 \\
\frac{1}{n} \sum_{i=1}^{n} x^i & \text{otherwise}
\end{cases} \]

A. Branching  
B. Loop  
C. Branching nested inside Loop  
D. Loop nested inside Branching

7. (10pts) In order to implement the following formula in a program,

\[ y(x) = \begin{cases} 
\tan(x) & x < 0 \\
x^3 & 0 \leq x \leq 1 \\
1 + \log_{10}(x) & x > 1
\end{cases} \]

A branching construct is required. Please finish the following flowchart to demonstrate the algorithm of the implementation. The logical expression and the statements inside each branch must be precise.
8. (10pts) Please predict the values held in the variables result and index at the end of the execution of each iteration in the following Fortran 95/2003 partial code:

```fortran
result = 0.0
index = 1
do while (index < 4)
    if (result < 2.0) then
        result = result + index
    else
        result = result + 1.0
    end if
    index = index + 1
end do
```

<table>
<thead>
<tr>
<th>The Iteration #</th>
<th>The value of result at the end of the iteration</th>
<th>The value of index at the end of the iteration</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1.0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>3.0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>4.0</td>
<td>4</td>
</tr>
</tbody>
</table>
9. (10pts) Please translate the following counting loop partial code into a DO-WHILE loop partial code in Fortran 95/2003 (assume that index is declared as integer type, and include statement outside the DO-WHILE loop if necessary):

```fortran
DO index=6,-6,-3
   write(*,*) index
END DO
```

index=6

DO WHILE (index >= -6 )

   write(*,*) index

   index=index-3

END DO
10. (10pts) The following Fortran 95/2003 code used a nested loop construct to process an input character string.

Program stringWork
Implicit none
Integer, parameter:: len=3
Character(len):: inputStr="", outputStr=""
Integer:: i=0, j=0

Write(*,*) "Enter a 3-character string:"
Read(*,*) inputStr

Do i=1:len
    outputStr(i:i) = inputStr(i:i)
    Do j=i+1:len
        If (outputStr(i:i)>inputStr(j:j)) then
            outputStr(i:i) = inputStr(j:j)
        end if
    End do
End do

Write(*,*) outputStr
End program

Assume that a user entered a string of “zxy” from the keyboard during run-time execution, predict the output from the program. You can use the following information.

| ‘x’ > ‘y’ | .false. |
| ‘y’ > ‘z’ | .false. |
| ‘z’ > ‘x’ | .true. |

‘xxv’
11. (30pts) Ideal Gas Law

An ideal gas is one in which all collisions between molecules are perfectly elastic. Such a gas can be characterized by three quantities: absolute pressure \( P \), volume \( V \), and absolute temperature \( T \). The volume of a gas can be determined given the pressure and temperature using the following equation

\[
V = \frac{nRT}{P}
\]

where
- \( P \) is the pressure of gas in kilopascals (kPa),
- \( V \) is the volume of the gas in liters (L),
- \( n \) is the number of molecules of the gas in units of moles (mol),
- \( R \) is the universal gas constant \( (8.314 \text{ L} \cdot \text{kPa} / \text{mol} \cdot \text{K}) \),
- \( T \) is the absolute temperature in kelvins (K).

(Note: 1 mol = 6.02 \times 10^{23} \text{ molecules})

Assume that a sample of an ideal gas contains 2 mols of molecules at a temperature of 273K. Write a Fortran 95/2003 program to calculate and print out the volume of this gas as its pressure varies from 1 to 1001 kPa in steps of 100 kPa. You can ignore any comments lines.

```fortran
program idealGasVolume
    implicit none
    real, parameter :: R = 8.314, n = 2, T = 273
    real :: P = 0.0, V = 0.0
    
P = 1.0
    do while (P <= 1001)
        V = n * R * T / P
        write (*, *) "Pressure=", P, " Volume=", V
        P = P + 100.0
    end do
end program
```