Homework Assignment 3

Section___________________

Student ID_________________________Name (Print)__________________________

Review of Chap3 Branch Construct, and user-defined function

Chapman exercise
Four coding assignments:
1) Chapman 3-4
2) Chapman 3-8
3) Chapman 3-11
4) Wrap the main algorithm of 3-11 into a user-defined function, then write a wrapper program to test the function.

For all the four coding assignments, please submit only the source code using Blackboard->Digital Dropoff. No hard-copy program planning is needed. This assignment is due 10/6/08 at 5pm for section 1 and 2, and due 10/7/08 at 5pm for section 3.
Solution

3-4) The following Fortran statements are intended to alert a user to dangerously high oral thermometer readings (values are in degrees Fahrenheit). Are they correct or incorrect? If they are incorrect, explain why and correct them.

```fortran
IF ( temp < 97.5 ) THEN
    WRITE (*,*) 'Temperature below normal'
ELSE IF ( temp > 97.5 ) THEN
    WRITE (*,*) 'Temperature normal'
ELSE IF ( temp > 99.5 ) THEN
    WRITE (*,*) 'Temperature slightly high'
ELSE IF ( temp > 103.5 ) THEN
    WRITE (*,*) 'Temperature dangerously high'
END IF
```

Incorrect. Corrected version is implemented in source code: 3-4.f95

```fortran
!Human body temperature check
program testTemp
    implicit none
    real::temp=0.0
    write(*,*) "Enter the temperature:"
    read(*,*) temp
    IF ( temp < 97.5 ) THEN
        WRITE (*,*) 'Temperature below normal'
    ELSE IF ( temp < 99.5 ) THEN
        WRITE (*,*) 'Temperature normal'
    ELSE IF ( temp < 103.5 ) THEN
        WRITE (*,*) 'Temperature slightly high'
    ELSE
        WRITE (*,*) 'Temperature dangerously high'
    END IF
end program
```

```
./a.out
Enter the temperature:
97
Temperature below normal
[jxue@backus ~hw3]$ ./a.out
Enter the temperature:
98
Temperature normal
[jxue@backus ~hw3]$ ./a.out
Enter the temperature:
100
Temperature slightly high
[jxue@backus ~hw3]$ ./a.out
Enter the temperature:
104
Temperature dangerously high
```
3-8) Suppose that a student has the option of enrolling for a single elective during a term. The student must select a course from a limited list of options: English, History, Astronomy, or Literature. Construct a fragment of Fortran code that will prompt the student for a choice, read in the choice, and use the answer as the case expression for a SELECT CASE construct. Be sure to include a default case to handle invalid inputs.

Solution in 3-8.f95

```fortran
!Course selection validation

program selectCourse
    implicit none
    character(16)::courseName=""
    logical::valid=.false.

    write(*,*) "Enter the course name:" 
    read(*,*) courseName 

    select case ( courseName )
        case ('English', 'History', 'Astronomy', 'Literature' )
            valid = .true.
            write(*,*) "Your selection is valid"
        case default
            valid = .false.
            write(*,*) "Your selection is invalid"
    end select
end program
```

```
./a.out
Enter the course name:
Literature
Your selection is valid
[jxue@backus ~hw3]$ ./a.out
Enter the course name: programming
Your selection is invalid
[jxue@backus ~hw3]$ ./a.out
Enter the course name: Astronomy
Your selection is valid
```
3-11) **Decibels**: In Exercise 2-21, we wrote a program to calculate a power level in decibels with respect to a 1-mW reference level. The equation implemented was

$$dB = 10 \log_{10} \left( \frac{P_2}{P_1} \right)$$

where $P_2$ is the power level being measured, and $P_1$ is reference power level (1 milliwatt). This equation uses the logarithm to the base 10, which is undefined for negative or zero values. Modify the program to trap negative or zero input values, and inform the user of the invalid input values.

**Source code: 3-11.f95**

```fortran
!Chapman 3-11, add input validation to 2-22 decibel program

program decibels
    implicit none
    real, parameter::p1=1
    real::p2=0, p2dB=0.

    write(*,*) "Please enter a power level in milliwatte:"
    read(*,*) p2

    !new code for 3-11 inserted from here in original 2-22.f95
    if ( p2 <= 0 ) then
        write(*,*) "Input power must be greater than 0.0"
        stop
    end if
    !new code for 3-11 end

    p2dB = 10*log10(p2/p1)
    write(*,*) p2dB
end program
```

```
./a.out
Please enter a power level in milliwatte:
0
Input power must be greater than 0.0
[jxue@backus ~hw3]$ ./a.out
Please enter a power level in milliwatte:
12
10.79181
```
Chapman 3-11 with user-defined function calls

Source code: hw3p4.f95

```fortran
program decibels
    implicit none
    real, parameter:: p1=1
    real:: p2=0
    real:: dBfunction

    write(*,*) "Please enter a power level in milliWatt:"
    read(*,*) p2

    if ( p2 <= 0 ) then
        write(*,*) "Input power must be greater than 0.0"
        stop
    end if

    write(*,*) dBfunction( p1, p2 )
end program

real function dBfunction( p1, p2 )
    implicit none
    real, intent(in):: p1, p2

    dBfunction = 10*log10(p2/p1)
end function
```