Jiangxi University of Science and Technology

Chapter 3 Processing and Interactive Input

Leture0304 Symbolic Constants

3.5 Symbolic Constants

THE C PROGRAMMING LANGUAGE

➢Symbolic Constants 符号常量

- **Literal data**字面数据 refers to any data within a program that explicitly identifies itself.
- Literal values that ***appear many times*** in the same program are called ***magic numbers*** 幻数
- you can define the value once by equating the number to a ***symbolic name***
- **#define SALESTAX 0.05**
- **#define PI 3.1416**
- Also called symbolic constants and named constants 命名常量

3.5 Symbolic Constants

THE C PROGRAMMING LANGUAGE

- Program 3.18

```
1.  #include<stdio.h>
2.  #define SALESTEX 0.05
3.  int main(){
4.          float amount, taxes, total;
5.          printf("enter the amount  purchased:");
6.          scanf("%f", &amount);
7.          taxes=SALESTEX*amount;
8.          total=amount+taxes;
9.          printf("the sales taxe is %f\n", taxes);
10.         printf("the total bill is  %f is %5.2f\n", total);
11.         return 0;
12. }
```

 # sign is a signal to a C preprocessor

6. Case Study

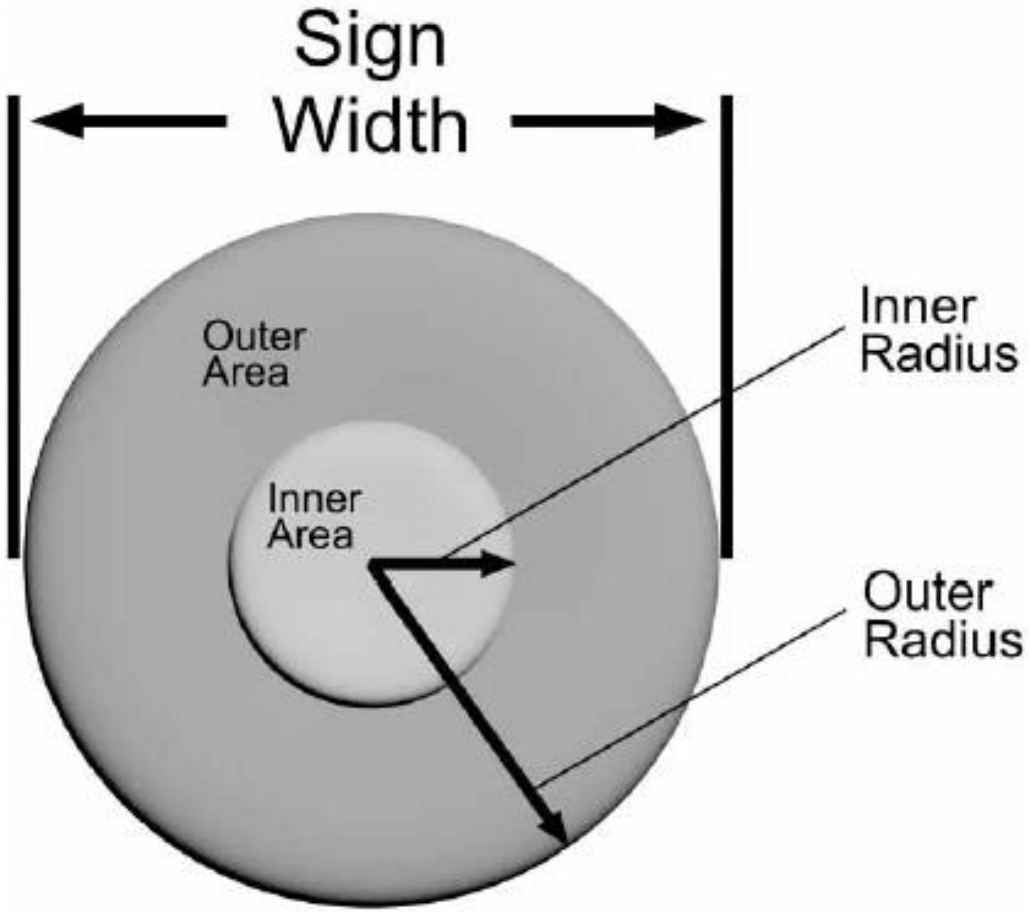

Figure 3.9 The Hit-The-Mark display

3.6 Case Study

Program 3.19

```
1.  #include <stdio.h>
2.  #include <math.h>
3.  #define SQFTPERQUART 200.0
4.  #define PI 3.14159
5.  int main(){
6.          float width, outerRadius, innerRadius;
7.          float totalArea, outerArea, innerArea;
8.          float red, blue;
9.  //get input data
10. printf("enter the width of display(in feet):");
11. scanf("%f", &width);
12. //determine two radius
13. outerRadius=width/2.0;
14. innerRadius=0.25*width;
15. //determine two area
16. totalArea=PI*pow(outerRadius,2);
17. innerArea=PI*pow(innerRadius,2);
18. outerArea=totalArea-innerArea;
```

6. Case Study

- **Program 3.19**

```
19.   //determine the gallons of paint needed;
20.    red =  innerArea / SQFTPERQUART;
21.    blue = outerRimArea / SQFTPERQUART;
22.    //provide the required outputs
23.    printf("the inner area is %5.2f sq. feet\n", innerArea);
24.    printf("the outer area is %5.2f sq. feet\n", outerArea);
25.    printf("red paint required  is %6.3f quarts\n", red);
26.    printf("blue paint required  is %6.3f quarts\n", blue);
27.    return 0;
28.  }
```

4.8 Summary

THE C PROGRAMMING LANGUAGE

- Arithmetic calculations can be performed using assignment statements or mathematical functions
- The assignment symbol, =, is an operator
- C provides the +=, -=, *= and /= assignment operators
- The increment operator, ++, adds 1 to a variable
- The decrement operator, --, subtracts 1 from a variable
- C provides library functions for calculating square root, logarithmic, and other mathematical computations

4.8 Summary

- Mathematical functions may be included within larger expressions
- scanf() is a standard library function used for data input
- When a scanf() function is encountered, the program temporarily suspends further statement execution until sufficient data has been entered for the number of variable addresses contained in the scanf() function call

4.8 Summary

- It is good programming practice to display a message, prior to a scanf() function call, that alerts the user as to the type and number of data items to be entered
- Field width specifiers can be included with conversion control sequences to explicitly specify the format of displayed fields
- Each compiled C program is automatically passed through a preprocessor
- Expressions can be made equivalent to a single identifier using the preprocessor #define command

Reference

- BOOK
- Some part of this PPT given by Prof 区 **Chengtian Ouyang)**
- with special thank
- https://www.codingunit.com/c-tutorial hello-world