Computer Programming & Problem Solving CS100

Mrs Sanga G. Chaki

Department of Computer Science and Engineering

National Institute of Technology, Goa

March, 2023

The Case Control Structure

Case Control Structure

1. When we have many choices, we use a series of if-elses
2. C provides a special control statement that allows us to handle such cases effectively
3. Switch control statement
4. switch-case-default: 3 keywords

Case Control Structure

1. General format

```
switch (expression) {
        case const-expr-1: S-1
        case const-expr-2: S-2
        :
        :
        case const-expr-m: S-m
        default: S
        }
```

1. expression is any integer-valued expression
2. const-expr-1, const-expr-2,…are any constant integer-valued expressions
3. Values must be distinct
4. S-1, S-2, …,S-m, S are statements/compound statements
5. Default is optional, and can come anywhere (not necessarily at the end as shown)

Case Control Structure: Behaviour

1. expression is first evaluated
2. It is then compared with const-expr-1, const-expr-2,…for equality in order
3. If it matches any one, all statements from that point till the end of the switch are executed (including statements for default, if present)
4. Use break statements if you do not want this (see example)
5. Statements corresponding to default, if present, are executed if no other expression matches

Case Control Structure: Example

```
main( )
{
int i = 2 ;
switch ( i )
    {
    case 1 :
    printf ( "I am in case 1 \n" ) ;
    case 2 :
    printf ( "I am in case 2 \n" ) ;
    case 3 :
    printf ( "I am in case 3 \n" ) ;
    default :
    printf ( "I am in default \n" ) ;
    }
}
```

1. The output of this program would be:

```
I am in case 2
I am in case 3
I am in default
```

Case Control Structure: Behaviour

1. If it matches any one, all statements from that point till the end of the switch are executed (including statements for default, if present)
2. Use break statements if you do not want this (see example)
3. BREAK: it is upto you to get out of the switch then and there by using a break statement

Case Control Structure: Example

```
main( )
{
int i = 2 ;
switch ( i ){
    case 1 :
    printf ( "I am in case 1 \n" ) ;
    break ;
    case 2 :
    printf ( "I am in case 2 \n" ) ;
    break ;
    case 3 :
    printf ( "I am in case 3 \n" ) ;
    break ;
    default :
    printf ( "I am in default \n" ) ;
}
}
```

1. The output of this program would be:

I am in case 2

Case Control Structure: What you cannot do

1. A float expression cannot be tested using a switch
2. Cases can never have variable expressions
   - for example it is wrong to say **case a +3 :**
3. But you can have case 1+2: because they are constants
4. Multiple cases cannot use same expressions. Illegal:
   1. switch (a)

```
    {
    case 3 :
    ...
    case 1 + 2 :
    ...
}
```

Switch vs If-else: when to use them?

1. Switch is faster for complex conditions – when there are more number of conditions
2. If conditions in the if-else were simple and less in number then if-else would work out faster