#8 Switch Case Control Instruction

By Saurabh Shukla | 2017©mysirg.com

Switch case control instruction

We studied earlier in decision control that how and when to use if, if-else. They are good enough to select one from the two available options. When choices are two, we can use if-else. Even if we have multiple choices we can use multiple if-else, sometimes we use if-else ladder or nested if-else for that reason.

Too much of nesting is not easy to understand. Moreover, nesting dictates the dependencies of conditions in the nested format. However, there are cases where you want to choose one from the n available option with no logical nesting required. For such cases switch case is an ideal solution.

(What is logical nesting? When a statement has to be run which depends on some N conditions and conditions are dependent, like if first condition meets then only check for second condition, similarly if second condition is true then check for the third condition. This is nested conditions or logical nesting)

- In this section we are going to learn three keywords switch, case and default.
- Apart from these keywords we will also see usage of break keyword in switch body.
- Remember; continue keyword cannot be used in switch body. It can reside inside loop body.
- Following is the syntax of switch

```
1   int main()
2   {
3       ….
4       ….
5       switch( expression)
6       {
7           case constant1 :
8                   block-1
9           case constant2 :
10                  block-2
11          case constant3 :
12                  block-3
13          …
14          default:
15              default-block
16      }
17      …..
18  }
19
```

- Switch case control is ideal in the situation where you have multiple options and every option has some value associated for identification of the case.
- Expression in switch parenthesis is evaluated as an integer result unlike if or while, where result of condition is either true or false.
- The result of expression in switchs's parenthesis is then looking for the correct match in among the available case statements in the body of switch.
- So control jumps from switch to appropriate matching case and then executes all the lines written in that case. In the absence of break statement, control keeps on executing other cases down the line till the end of switch body.
- break keyword can be used to throw control out of the switch body.
- case keyword is followed by a constant value, which must be unique integer only. It can never be a variable or expression. It can never be a real constant. It can be a character because they are basically numbers (ASCII codes).
- You can write any number of cases in switch body.
- default keyword is used to handle the default case, that is when no case matches.
- It is not mandatory to write default keyword at the end of all cases, but you often see it in practice.
- Following is an example program to understand the use of switch case control statement. It is a menu driven program, where a menu is given to the user and user is supposed to select a value from the available options.

```
1   int main()
2   {
3       int a, b, result, ch;
4
5       while(1) // condition is always true, thus an infinite loop
6       {
7           printf("\n1. Addition");
8           printf("\n2. Subtraction");
9           printf("\n3. Multiplication");
10          printf("\n4. Division");
11          printf("\n5. Exit");
12          printf("\n\nEnter your choice");
13          scanf("%d",&ch);
14      switch(ch)
15      {
16          case 1:
17              printf("Enter two numbers");
18              scanf("%d%d",&a,&b);
19              result=a+b;
20              printf("Sum is %d", result);
21              break; //it is used to move control out of switch b
22          case 2:
23              printf("Enter two numbers");
24              scanf("%d%d",&a,&b);
25              result=a-b;
26              printf("Difference is %d", result);
```

```
27              break; //it is used to move control out of switch bo
28          case 3:
29              printf("Enter two numbers");
30              scanf("%d%d",&a,&b);
31              result=a*b;
32              printf("Product is %d", result);
33              break; //it is used to move control out of switch bo
34          case 4:
35              printf("Enter two numbers");
36              scanf("%d%d",&a,&b);
37              result=a/b;
38              printf("Quotient is %d", result);
39              break; //it is used to move control out of switch bo
40          case 5:
41              exit(0);
42          default:
43              printf("Invalid Entry");
44      }
45      getch();
46    }
47  return(0);
48 }
49
```

- There are few things to discuss about above program.
    - Notice while loop which executes infinite times till you select 5 from the menu.
    - In case 5, we use a predefined function exit. The job of this function is to terminate program. Argument 0 in the function depicts the normal termination. Argument could be 1 passed in the function in the case of abnormal termination.
    - The keyword break is used in each case as we want to transfers the control outside switch body
    - Whenever user selects a number as a choice other than specified, switch moves the control to default segment.
    - No need to write keyword break after default because it is already present at the end of switch body.

Remember:

- You cannot have duplicate case in a switch body. Constants after case keyword must be unique.
- You can have switch inside a switch case.

References

YouTube video links

- Lecture 8 Switch case Control Instruction

- https://www.youtube.com/watch?v=BDPhf6moDL8&feature=youtu.be&list=PL7ersPsTyYt2Q-SqZxTA1D-melSfqBRMW

Exercise

1. Write a menu driven program with the following choices
    a. Check Prime
    b. Factorial
    c. Fibonacci series
    d. LCM of two numbers
2. Write a menu driven program of calculator.