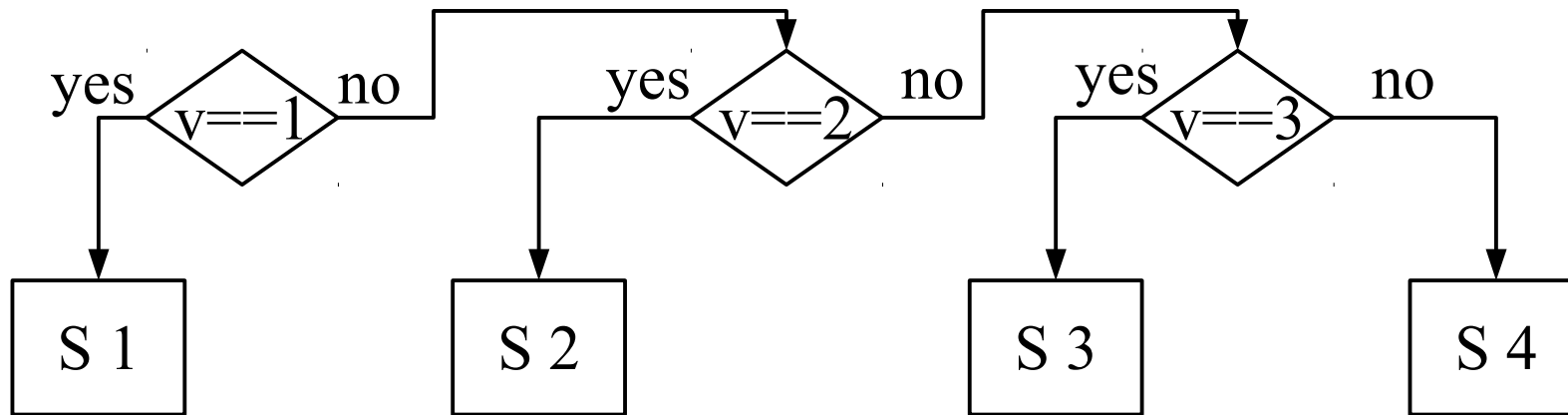


THE SWITCH STATEMENT:

The diagram structure of the switch statement is similar to a cascaded conditional, but the possible choices are given by the value of a variable (in this case we will call it v):



In Java the switch statement is as follows:

```
switch (v) {  
    case 1:  
        S1  
        break;  
    case 2:  
        S2  
        break;  
    case 3:  
        S3  
        break;  
    default:  
        S4  
        break;  
}
```

S1, S2, S3, and S4 are
sequence of statements.

If the break statement is not included in a case, then the next case will be executed.

Putting it all together:

We will write a complete program that prompts the user for two numbers and a menu option from:

1. Add
2. Subtract
3. Multiply
4. Divide

And outputs the result of the corresponding operation on the two numbers entered.

Sample interaction:

```
Enter one number : 2
Enter another number: 5
1. Add
2. Subtract
3. Multiply
4. Divide
Enter an option: 1
x + y = 7
```

Black letters represent the computer's output and red letters represent the user's input.

Algorithm:

We need three variables (x, y, z) for the two numbers and the answer

Prompt the user for one number

Read x

Prompt the user for another number

Read y

Display the menu (see previous slide)

Read op

If $op = 1$ then $z = x + y$

Print z

If $op = 2$ then $z = x - y$

Print z

If $op = 3$ then $z = x * y$

Print z

If $op = 4$ then $z = x / y$

Print z

We have to be careful with the last case, because y could be zero!. Also must consider the possibility that the user answered an invalid option number (other than 1, 2, 3, 4)

We will write this program from scratch:

```
public class OpMenu
{
    public static void main(String[] args)
    {
        double x,y,z;

        System.out.print("Enter one number : ");
        x=IO.readDouble();

        System.out.print ("Enter another number : ");
        y=IO.readDouble();

        System.out.println("1. Add");
        System.out.println("2. Subtract");
        System.out.println("3. Multiply");
        System.out.println("4. Divide");
        System.out.print("Enter option : ");

        int op=IO.readInt();

        switch(op){
        case 1:
            z=x+y;
            System.out.println("x + y = "+z);
            break;
        case 2:
            z=x-y;
            System.out.println("x - y = "+z);
            break;
```

print vs println:

After using print, the next output from the program will be on the same line.

After using println, the next output from the program will be on the next line.

The character that causes the line to change is called a Line Feed

```

case 3:
    z=x*y;
    System.out.println("x * y = "+z);
    break;
case 4:
    if (y==0) {
        IO.reportBadInput();
        return;
    }
    else{
        z=x/y;
        System.out.println("x / y = "+z);
    }
    break;
default:
    System.out.println("option must be 1, 2, 3, or 4");
    break;
} // switch
} // main
} //class

```

A return statement tells the program to exit the current method. In this case exiting the main method causes the program to end.

Instructions in the `println`: everything that is between quotes is considered a to be of a text type called String. The `+` operator is a String concatenation operator so that the result is a string.

For example, suppose that `z=3.4`, then

`"x + y = "+z`

is the string `"x + y = 3.4"`