

SWITCH & BREAK STATEMENTS

The switch statement.

Repeated if..else if..else if.. constructs can be replaced by the switch statement, which has the format:

```
switch (expression) {
    case const-expr: statements;
    case const-expr: statements;
    ...
    default: statements;
}
```

Each case is labelled by one or more integer-valued constants or constant expressions. If a case matches the expression value, execution starts at that case. The case labelled 'default' is executed if none of the other cases are satisfied, but if default is not present then no statement is executed.

Because cases serve as labels only, execution falls through to the next case unless explicit action to escape is taken. Break and return are the most common ways to leave a switch and must be used unless it is required to deliberately proceed to the statements of the next case.

The break statement.

The break statement can be used with all three loop statements and with the switch statement, but not with the if...else statement. It permits the quick and easy exit from a loop, before the end is reached and the exit from a switch statement.

The break statement must not be used in any other structure to exit a branch or condition. Other key words that promote bad programming practice are 'exit' and 'goto' - these should not be used on this course.

Example Code:

```
void main (void)
{
    int Option;

    printf("\nEnter option 0 - 4: ");
    scanf("%d",&Option);
    switch(Option)
    {
        case 0: /*do something*/
            break;
        case 1: /*do something else*/
            break;
        case 2: /*do something different*/
            break;
        case 3: /*do something completely different*/
            break;
        case 4: /*OK go wild here*/
            break;
        default: /*tell the user to input 0,1,2,3 or 4!*/
    }
}
```