

## Chapter 13

13.1.

Name	Type	Offset	Scope
operand1	int	0	main
operand2	int	-1	main
operation	char	-3	main
result	int	-2	main

13.3.

```
if (a)
    x = b;
else
    x = c;
```

13.5.

```
AND    R0, R0, #0        ; init r0 at 0
LDR    R1, R5, #0

BRz    CASE_1           ; compare x==0
ADD    R1, R1, #-1
BRz    CASE_2           ; compare x==1
BR     CASE_DEF        ; goto default case
```

```
CASE_1:
ADD    R1, R0, #3
STR    R1, R5, #-1 ; y = 3
```

```
CASE_2:
ADD    R1, R0, #4
STR    R1, R5, #-1 ; y = 4
BR     END_SWITCH ; break
```

```
CASE_DEF:
ADD    R1, R0, #5
STR    R1, R5, #-1 ; y = 5
BR     END_SWITCH ; break
```

```
END_SWITCH:
.
.
.
```

13.7. This if-else statement **cannot** be converted into a switch statement. All cases labels must be integral constants. The if conditional ( $x == y$ ) cannot be converted into a case label for the switch.

13.9. a. 0  
b. 0  
c. 11 4

13.11.

```
#include <stdio.h>
#define TRUE 1
#define FALSE 0

int main()
{
    char nextChar;    /* Next character in email address */
    int gotAt  = FALSE; /* Indicates if At @ was found */
    int gotDot = FALSE; /* Indicates if Dot . was found */
    int charCount = 0;

    printf("Enter your email address: ");

    do {
        scanf("%c", &nextChar);
        charCount++;

        if (nextChar == '@' && charCount > 1) {
            gotAt = TRUE;
            charCount = 0;
        }

        if (nextChar == '.' && gotAt == TRUE && charCount > 1) {
            gotDot = TRUE;
            charCount = 0;
        }
    }
    while (nextChar != ' ' && nextChar != '\n');

    if (gotAt == TRUE && gotDot == TRUE && charCount > 1)
        printf("Your email address appears to be valid.\n");
    else
        printf("Your email address is not valid!\n");
}
```

13.13.

```
#include <stdio>

int main()
{
    int i;
    int sum;

    i = 0;

    do
    {
        if (i % 4 == 0)
            sum = sum + 2;
        else if (i % 4 == 1)
            sum = sum - 6;
        else if (i % 4 == 2)
            sum = sum * 3;
        else if (i % 4 == 3)
            sum = sum / 2;

        i++;

    }
    while (i <= 100);

    printf("%d\n", sum);
}
```

13.15.

```
a.   for ( ; condition; )
      loopbody;

b.   init;
      while (condition)
      {
          loopbody;
          reinit;
      }
```

13.17. It counts the number of bits that are set in the two's complement representation of the integer provided by the user.

**Questions in the text denoted by the question mark icon:**

Page 353      It "echoes" the user input back to the monitor.

Page 355      Loop 1: 0 1 2 3 4 5 6 7 8 9 10  
Loop 2: a b c d e f g h i j k l m n o p q r s t u v w x y z  
Loop 3: Counts the number of bits that are set in inputValue