TRIM-2 — Programming & Problem solving | Krishnaraj PT
109054 . I3

19/01/22

ASSIGNMENT - 1

9.1. a. Explain Nested If statements

→ When an If statement is used inside another If statement, that set of If statements is called Nested If statements.

→ You cannot have nested 'else' statements. only 'If' statements.

→ It is a concept that is common to almost all programming languages.

eg. Syntax

```
if ( a == b) {
if (b = c) {
        statements ;
}
statements ;
}
```

This If statement is inside the outer one.

eg.

```
#include <stdio.h>
int main()
{
    int a;
    printf("Enter a number");
    scanf("%d", &a);
    if (a % 2==0)
    {
        if (a % 5 ==0)
        { printf("Number is divisible
          by both 2 and 5");
        }

        else if (a % 3 ==0)
        { printf("Number is divisible
          by both 2 and 3");
        }
    } // Divisiblety test.
    return 0;
}
```

B. Use of Break statement:

→ (1) To exit an iteration in any loop

(2) To end an infinite loop

(3) To move on to the next statements after particular 'case' execution of 'switch case' statements.

(4) Def: Break is a keyword that can be used within loops to terminate the iteration of that loop at that point.

(5) eg:

```
switch ( choice_1 )
{
    case 1 : printf ("A");
             break;
    case 2 : ....
             break;
    default : .....
}
```

Q.2. What is an array? Give examples and advantages of using arrays.

→ Def:

An ~~arra~~ array is a collation of similar data elements stored at contiguous memory locations.

→ It is the simplest data structure where each element is only accessed by using its index number.

Advantages

(1) Multiple data types of the same type can be represented by a single name

(2) Reduces use of variables.

(3) Memory efficient

(4) Avoids Memory overflow and shortage

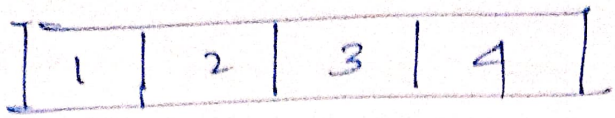
(5) Elements can be accessed randomly very fast using this index number.

(6) using arrays, other Data structures like linked list, stacks, ~~queues~~ queues etc can be implemented.

(7) Representing Tables and Matrices.

(9) → `int a[4] = { 1, 2, 3, 4 };`

a[1] a[3]



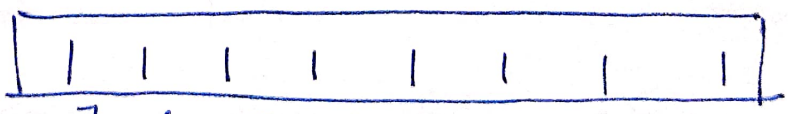
a[0] a[2]

(eg:)

```
{ printf (" Enter Roll NO. of 10 students ");
  int a[10] ;
  for (int i =0 ; i <10 ; i++)
  {   scanf (" %d", & a[i]);
  }
  return 0;
}
```

9. Explain signed and unsigned data representation

→ Data in Binary is represented as powers of 2.



$= (255)_{10}$

* this is an unsigned Binary number as ALL bits are used to represent the number.

→ There is no need of a 'sign' bit in unsigned numbers.

So max range is

$$0 \text{ to } (255)_{\text{decimal}}$$

→ Only positive numbers can be represented.

Signed Numbers :

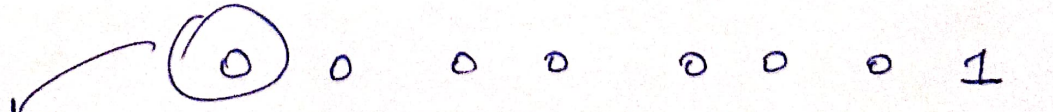
→ You can Represent negative and positive numbers.

→ Range is -127 to $+127$ for 8 bit integers.

→ Left most bit is used to indicate if number is positive or negative.

(1 Rep Negative No.)

→ $(-1)_{10}$.



0 Rep (+)ve Numbers.

→ $(1)_{10}$

Q.9, Convert given Binary number to Hexadecimal :

→ $(11001)_2$

$= 1 \times 2^0 + 0 \times 2^1 + 0 \times 2^2 + 1 \times 2^3 + 1 \times 2^4$

$= (25)_{10}$

16 | 25
16 | 1 — Rem : 9
0 — Rem : 1

So Hex $= (19)_{16} = (11001)_2$

$= (25)_{10}$