EMBEDDED C

Module 2: 20 Marks → 2 x 10 Programs

Control:

- if - else, nested if
- else - if ladder
- switch case

Loops:

- do - while
- while
- for
- break
- continue
- goto, exit

volatile

typedef – Alternate name to existing data type

register – Store in CPU for faster access

Global Variable

static variable → confine the scope! No need for reinitialization!

extern var1; → Variable declared in other file

```
func1 () {
static a = 10;
a++;
printf ("%d", a)
}
int main () {
func1();
func2();
}
```

SCANF:

%.12Ld → D: show error

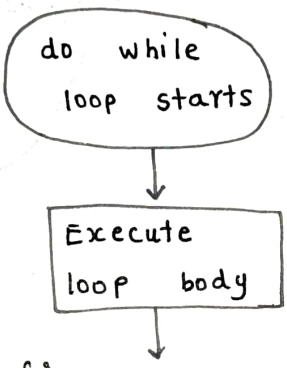
↳ Read only first 12 digit.

PRINTF:

%8d → sign also included → 8 digit

30.7.24

Do - while:



i = -99 ; i <= 10 ; i++)

-99
-98
.
.
.
10

goto:

goto label:

label;
statement;

Forward

label 2;

goto label:

Back

exit() → Terminate program @ any point.

Pattern Program in cat 1 — 5 Marks

1. Total Digits in a number
2. Sum of digits
3. Armstrong Number
4. Prime Number
5. Reverse

```
#include <stdio.h>
#include <math.h>
int main () {
        int num, temp1, temp2, digit, sum = 0, count=0;
        printf ("Enter No. :");
        scanf ("%d", &num);

        temp1 = num; temp2 = num;
        while (temp1 > 0) {
                digit = temp1 %.10;
                temp /= 10;
                count ++;
        }
```

~~dig~~

```
while ( temp2 > 0 ) {

        digit = num % 10;
        ~~num~~ temp2 /= 10;
        sum += pow (digit, count);

    }

if (num == sum) {

        printf ("Armstrong No.");

}
else {

    printf (" Not Armstrong No.");

}
```