Bunch-2

H/w → consider a currency system in which there are notes of 6 denominations, 1, 2, 5, 10, 50, 100. If a sum of N is entered through the keyboard, write a Program to compute the smallest no. of notes that will combine to give N.

Decision

if-else:-

Simple if:

```
if (codtn)
{
    stmt 1;  → use braces if
       2       more than 1
}              statement
```

if-else:-

```
if (cdtn)  → relational operators
    stmt 1;
else
    stmt 2;  → Prog: check equality of no's
             ✓ even/odd.
```

Nested-if:-

```
if (cdtn)
    stmt 1;
else
{  if (cdtn)      → Prog
       stmt 2;      grades based on avg.
   else
       stmt 3;    ✓ Leap yr or not.
}
```

Caution:-

usually on edtos
but,

```
if (3+2%5)
 if (a=10)
  if (-5)
```

} all happens to be true

irrespective of integer or float, issue is whether zero (or) non-zero

Caution:

```
if (i==5);
    printf ("you entered 5 ");
        ↓
if(i==5)
    ;      → nothing gets evaluated
```

else if ladder

%	above or equal to 60	- 1st division
	50-59	- 2nd "
	40-49	- 3rd "
	< 40	- fail.

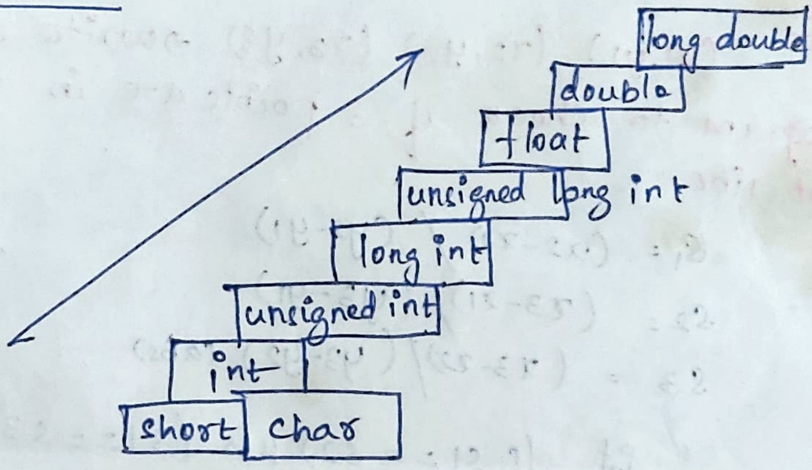
```
if (per >= 60)
        printf
else if ( per >= 50)
        printf
else if ( per >= 40)
        printf
else
        printf ("fail");
```

float → 6 digits of prec.
double → 14 "

<u>Implicit</u>
Type conversions:-

lower type → higher type

<u>seq.of rules</u>



float → int = no frac part
double → float = causes rounding of digits
long int → int ⇒ dropping of excess higher order bits.

explicit conversion:-

force the type conversion,

(ratio) → float

$$ratio = \frac{female\_no}{male\_no}$$

$$ratio = (float)\ female\_no / male\_num$$

Switch

257 case labels.

Prob:

(x1, y1), (x2, y2) (x3, y3) → write a c Program to check if 3 points are in a strt line.

```
S1 = (x2-x1) / (y2-y1)
S2 = (x3-x1) / (y3-y1)
S3 = (x3-x2) / (y3-y2) (abs)

if ((S1 == S2) && (S1 == S3))
    collinear
else
    Not
```

write a program to find out if it lies on x-axis, y-axis or origin.

```
if (x == 0 && y == 0)
    Print origin
else if (x == 0 && y != 0)
    Print Y-axis
else if (x != 0 && y == 0)
    print x-axis
else
    Print (neither).
```

H/W: write a c program to check a Δ is Valid or not.

3 ~~side~~ angles i/p → valid only when sum = 180°

cdtl optr:-

1) y = (x > 5 ? 3 : 4);

2) int big, a, b, c;
   big = (a > b ? (a > c ? a : c) : (b > c) ? b : c));

3) (i = = 1 ? printf ("equal") : printf ("not equal"));

Switch:

```
Switch (int expr)
{
    case constant 1 :
        stmt 1 ;
        break ;
    case constant 2 :
        stmt 2 ;
        break ;
    .
    .
    default :
```

→ Cases can be in any order.

→ no braces for multiple statements

→ default, is optional.

Switch (i+j*k)

switch (23+45 % 4 * k)

case 'a':
'A':

case 'b':
'B':

✓ allowed.

→ mainly used for menu-driven programs.
→ nested switch is done.
→ float cannot be used.
→ switch works faster than if-else ladder

Cases cannot have (case a+3 : ✗
Variable expr).

✗ case 3 :
case 1+2 : ✓ valid

Goto statement.

```
goto out :

out :     → label.
```

ex).

```
if (goals <= 5)
    goto sos;
else
```

```
    Printf " (great ");
    exit(1);
  }

SOS:
       Printf ("To err is human\n");
}.
```

Any no. of goto.

exit(i) → lib fn. <stdlib.h>

Looping statements

→ actions over & over → with variations each time.
(Looping)
→ set of instrs repeatedly.
→ specified no:of times | or till a condition is reached.

while loop

→ repeat till the cdtn remains true.

Syntax:

```
        while (condition)
              statement ;

        while (condition)
        {
              stmt1;
        }     stm22;
```