=>else-if ladder:When we have multiple conditions to be tested at root level.

Syntax:

3 if(cond1)

4 {

5 }

6 else if(cond2)

7 {

8 }

9 else if (cond3)

10 {

11 }

12 else

13 {

14 }

EX:

16 #include<stdio.h>

17 int main()

18 {

19 char clrcode;

20 clrcode=getchar();

21 if(clrcode == 'B')

22 printf("\nBlack\n");

23 else if(clrcode == 'N')

24 printf("\nBrown\n");

25 else if(clrcode == 'R')

26 printf("\nRed\n");

27 else if(clrcode == 'O')

28 printf("\nOrange\n");

29 else

30 printf("\nEnter the correct colour code");

31 printf("\nProgram ended here\n");

32 return 0;

33 }

switch case:\*Case values must be numeric or ascii values.

\*switch case is faster than else-if ladder.

\*We can not have duplicate cases.

Syntax:

4 switch(cond)

5 {

6 case 1:

7 --

8 break;

9 case 2:

10 --

11 break;

12 default:

13 --

14 break;

15 }

EX:

17 #include<stdio.h>

18 int main()

19 {

20 char clrcode;

21 clrcode = getc(stdin); (or)switch((clrcode=getc(stdin)) For space optimezation

22 switch(clrcode)

23 {

24 case 'O':

25 printf("orange");

26 break;

27 case 'B':

28 printf("Black");

29 break;

30 case 'N':

31 printf("Brown");

32 break;

33 default:

34 printf("\nEnter the correct colour code\n");

35 break;

36 }

37 printf("\nProgram Ended\n");

38 return 0;

39 }

Loop:\*Initialize variable

\*cond check

\*sts to be exceuted

\*counter

Three Types of looping:

1.do while

2.while

3.for

two types of loop

1.entry controlled:First it will check condition, is it true, it will execute.while, for

2.exist controlled:First time it will exceute, then it will check for condtions.do while

do while:

syntax:

do

{

sts

} while(cond);

\*/

#include<stdio.h>

int main()

{

char taste='g';

int emptyplate=10;

int sFull=0;

do

{

printf("\nHave a Bite of food\n");

emptyplate--;

printf("\nStomach Full (1/0): ");

scanf("%d",&sFull);

printf("\nHow was the taste (g/b): ");

scanf(" ");

taste = getchar();

printf("\nEmpty plate value: %d",emptyplate);

}while ((taste == 'g') && ((sFull==0) && (emptyplate>=0)));

printf("\nProgram Ended\n");

return 0;

}

while

syntax:

while(cond1)

{

sts

}

\*/

#include<stdio.h>

#define SUCCESS 0

#define FAILURE -1

#define TRUE 1

#define FALSE 0

int diplaymenu();

int main()

{

int ch;

int flag = 0,a,b;

while(1)

{

flag = 0;

ch = displayMenu();

if (ch == 5);

break;

int a,b;

printf("\nEnter two numbers");

scanf("%d,%d",&a,&b);

switch(ch)

{

case 1:

printf("\nAddition function executed of %d + %d = %d\n",a,b,(a+b));

break;

case 2:

printf("\nSub function executed\n");

break;

case 3:

printf("\nMul function executed\n");

break;

case 4:

printf("\nDivision function executed\n");

break;

case 5:

printf("\nExist from the program\n");

flag = 1;

break;

default:

printf("\nEnter the correct choice\n");

}//end of switch case

if(flag == 1)

break;

}//end od while loop

printf("\nProgram Ended\n\n");

return SUCCESS;

}

int displayMenu()

{

int choice;

printf("\nPress,");

printf("\n1. Addition");

printf("\n2. Sub");

printf("\n3. Mul");

printf("\n4. Division");

printf("\n5. Exit");

printf("\nEnter the choice:");

scanf("%d",&choice);

return choice;

}