### 1. WAP to input an integer and display each byte of data.

#include<stdio.h>

typedef unsigned char BYTE;

int main()

{

unsigned int value;

Scanf();

BYTE a,b,c,d;

a=(value&0xFF);

b=((value>>8)&0xFF);

c=((value>>16)&0xFF);

d=((value>>24)&0xFF);

printf("a= %02X\n",a);

printf("b= %02X\n",b);

printf("c= %02X\n",c);

printf("d= %02X\n",d);

return 0;

}

### Define a structure pkt

### struct pkt{ char ch1; char ch2[2]; char ch3; };WAP to store each byte of data into the pkt

#include <stdio.h>

struct pkt

{

char ch1;

char ch2[2];

char ch3;

};

int main()

{

// unsigned int i = 0x11706151;

unsigned int i;

printf("Enter a number: ");

scanf("%x", &i);

printf("Original Value: \n");

printf("%x\n", i);

struct pkt p;

p.ch1 = i & 0xFF;

p.ch2[0] = (i >> 8) & 0xFF;

p.ch2[1] = (i >> 16) & 0xFF;

p.ch3 = (i >> 24) & 0xFF;

printf("\nValue of each byte: \n");

printf("%x, %x, %x, %x", p.ch1, p.ch2[0], p.ch2[1], p.ch3);

}

### WAP to combine each byte of data from pkt and display the number.

#include<stdio.h>

#include<string.h>

struct pkt{

char ch1;

char ch2[2];

char ch3;

};

int main()

{

struct pkt p;

int number,r,c=1;

printf("Enter a number of 4 digits \n");

scanf("%d",&number);

while(number!=0 && c<=4)

{

r=number%10;

if(c==1)

p.ch1=r;

if(c==2)

p.ch2[0]=r;

if(c==3)

p.ch2[1]=r;

if(c==4)

p.ch3=r;

c++;

number=number/10;

}

printf("The aggregated characters are: %d %d %d %d ",p.ch3,p.ch2[1],p.ch2[0],p.ch1);

printf("\nThe original number: %d%d%d%d",p.ch3,p.ch2[1],p.ch2[0],p.ch1);

}

### WAP to test your system stores data in Little Endian or Big Endian format.

#include <stdio.h>

#include<string.h>

int main()

{

unsigned int value;

scanf("%d",&value);

char \*c = (char \*) &value;

if (\*c)

printf("Little endian\n");

else

printf("Big endian\n");

int i;

return 0;

}