

```
*****
Report: hw6_1 + hw6_2
Author: F74045034 劉俊林<laochanlam@gmail.com>
Class: 甲班
Description: 利用 integer pointer(hw6_1)和 union(hw6_2)來進行
              1. Float to binary number
              2. Binary number to float
              3. Binary number to double
              三者的運算.
```

```
*****
hw6_1 Code:
```

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int main(){
unsigned cher;
int i,j,k,m,n;
float f;
unsigned mask = 1<<31 ;

////////////////////////////////////

printf("Input the float number:");    //Float to binary
number

scanf("%f",&f);
cher =*(unsigned*)&f;

for (i=0;i<32;i++)
{
    putchar(mask & cher? '1' : '0' );
    cher <<=1;
}

////////////////////////////////////
```

```
printf("\n");          // Binary number to float
printf("Input binary number to convert float number:");
printf("\n");
```

```
int a[32] = {0};
char c[32];
int x,y;
```

```
scanf("%s",&c);
```

```
for (i=0;i<32;i++)
a[i] = c[i] - '0';
```

```
y=1;x=0;
for (i=8;i>=1;i--)          // Exponent
{
    if (a[i] == 1)
        x = x+y;
    y=y*2;
}
```

```
float rx,ry;
int e;
e=0;
ry=0.5;
for (i=9;i<=31;i++)//Mantissa
{
    if (a[i] == 1)
        rx = ry + rx;
    ry=ry*0.5;
}
```

```
e = x - 127;
```

```

m=0;n=0;j=0;

float ans = 1;

while (e>=0 & j==0) //Mantissa
{
    ans = ans*2;
    n++;
    if (n==e)
        j=1;
}

while (e<0 & j==0) //Mantissa
{
    ans = ans*0.5;
    n--;
    if (n==e)
        j=1;
}
float final = (1 + rx) * ans;
if (a[0]==1) //Sign bit
    final = final *-1;
printf("%f",final);

////////////////////////////////////

printf("\n"); //Binary number to double
printf("Input binary number to convert double number:");
printf("\n");

int la[64] = {0};
char lc[64];
long long int lx,ly;

scanf("%s",&lc);

```

```

        for (i=0;i<64;i++)
            la[i] = lc[i] - '0';

ly=1;lx=0;
for (i=11;i>=1;i--) // Exponent
    {
        if (la[i] == 1)
            lx = lx+ly;
        ly=ly*2;
    }

double lrx,lry;
long long int le;
le=0;
lry=0.5;
for (i=12;i<=63;i++) //Mantissa
    {
        if (la[i] == 1)
            lrx = lry + lrx;
        lry=lry*0.5;
    }

le = lx - 1023;
m=0;n=0;j=0;
double lans = 1;

while (e>=0 & j==0) //Mantissa
{
    lans = lans*2;
    n++;
    if (n==le)
        j=1;
}

while (e<0 & j==0) //Mantissa
{
    lans = lans*0.5;

```

```

        n--;
        if (n==le)
            j=1;
    }
    double lfinal = (1 + lrx) * lans;
    if (la[0]==1) //Sign bit
        lfinal = lfinal *-1;
    printf("%lf\n",lfinal);

    return 0 ;

}

```

Compilation:

```
gcc hw6_1.c -o hw6_1
```

Execution:

```
./hw6_1
```

Input:

1. Float
2. Binary
3. Binary 64-bit

Output:

```

Input the float number:
(1. binary number)
Input binary number to convert float number:
(2. float)
Input binary number to convert double number:
(3. double)

```

error message:

在 char 的轉換及次方運算有出現錯誤,解決後沒有問題.

hw6_2 Code:

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

union array{
    char c[32];
    char lc[64];
    };

int main(){
    unsigned cher;
    int i,j,k,m,n;
    float f;
    unsigned mask = 1<<31 ;
    union array ar;
    //////////////////////////////////////

    printf("Input the float number:");    //Float to binary
    number

    scanf("%f",&f);
    cher =*(unsigned*)&f;

    for (i=0;i<32;i++)
    {
        putchar(mask & cher? '1' : '0' );
        cher <<=1;
    }

    //////////////////////////////////////

    printf("\n");    // Binary number to float
    printf("Input binary number to convert float number:");
    printf("\n");
```

```

int a[32] = {0};
int x,y;
    scanf("%s",&ar.c);

    for (i=0;i<32;i++)
        a[i] = ar.c[i] - '0';

y=1;x=0;
for (i=8;i>=1;i--)          // Exponent
    {
        if (a[i] == 1)
            x = x+y;
        y=y*2;
    }

float rx,ry;
int e;
e=0;
ry=0.5;
for (i=9;i<=31;i++) //Mantissa
    {
        if (a[i] == 1)
            rx = ry + rx;
        ry=ry*0.5;
    }

e = x - 127;

m=0;n=0;j=0;

float ans = 1;

while (e>=0 & j==0) //Mantissa

```

```

{
    ans = ans*2;
    n++;
    if (n==e)
        j=1;
}

while (e<0 & j==0) //Mantissa
{
    ans = ans*0.5;
    n--;
    if (n==e)
        j=1;
}
float final = (1 + rx) * ans;
if (a[0]==1) //Sign bit
    final = final *-1;
printf("%f",final);

////////////////////////////////////

printf("\n"); //Binary number to double
printf("Input binary number to convert double number:");
printf("\n");

int la[64] = {0};
long long int lx,ly;

scanf("%s",&ar.lc);

for (i=0;i<64;i++)
    la[i] = ar.lc[i] - '0';

ly=1;lx=0;
for (i=11;i>=1;i--) // Exponent
{

```



```

        if (la[i] == 1)
            lx = lx+ly;
        ly=ly*2;
    }

```

```

double lrx,lry;
long long int le;
le=0;
lry=0.5;
for (i=12;i<=63;i++)        //Mantissa
    {
        if (la[i] == 1)
            lrx = lry + lrx;
        lry=lry*0.5;
    }

```

```

le = lx - 1023;

```

```

m=0;n=0;j=0;
double lans = 1;

```

```

while (e>=0 & j==0) //Mantissa
{
    lans = lans*2;
    n++;
    if (n==le)
        j=1;
}

```

```

while (e<0 & j==0) //Mantissa
{
    lans = lans*0.5;
    n--;
    if (n==le)
        j=1;
}

```

```
}

double lfinal = (1 + lrx) * lans;
if (la[0]==1) //Sign bit
    lfinal = lfinal *-1;
printf("%lf\n",lfinal);

return 0 ;

}
```

Compilation:

```
gcc hw6_2.c -o hw6_2
```

Execution:

```
./hw6_2
```

Input:

1. Float
2. Binary
3. Binary 64-bit

Output:

```
Input the float number:
(1. binary number)
Input binary number to convert float number:
(2. float)
Input binary number to convert double number:
(3. double)
```

error message:

進行 32-BIT 到 64-BIT 修改時忘記把 127 改成 1023,導致錯誤一直出現,修改後無問題.

```
(1)
#include <stdio.h>
#include <float.h>
int main()
{
float a,b;
a=FLT_MIN;
b=1.17549435082228750796873653722224567781866555677208752
150875170627841725945472717285156050000000000000000000000
0000000000e-38f;
if(a==b)
    printf("Yes");
else
    printf("No");

    return 0;
}
```

```
1.1754943508222875079687365372222456778186655567720875215  
087517062784172594547271728515605000000000000000000000000  
00000000e-38f 為最小的 floating point number.
```

(3) 因為程序沒有給予要精確到哪個位置(精確度),所以得此結果.