

hw10程式碼的思維與過程

以下是我的作業思路

我先將第一段要求先宣告出來

1. Assume there is a function declared as **(1) double power(double, int)** that calculates x^n if we call `power(x, n)`, a function declared as **(2) double multiply(double, int)** that calculate $x*n$ if we call `multiply(x, n)`, and a function declared as **(3) double divide(double, int)** that calculate x/n if we call `divide(x, n)`, where x must be double and n be integer.

程式碼如下

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<string.h>
4  #include<math.h>
5
6  /*
7   * 函式power回傳x^n
8   * 首先處理n為負數的情況，將x替換成倒數，n轉正再傳入power()
9   * 再來考慮到n=0的情況，直接回傳1
10  * 再來考慮到n=1的情況，直接回傳x本身
11  * 接下來使用遞迴計算x的n次方
12  */
13 double power(double x, int n){
14     if(n<0) return power((double)(1/x), -n);
15     if(n == 0) return 1.0;
16     if(n == 1) return x;
17     return (double)x*power(x,n-1);
18 }
19
20 // 函式multiply回傳x*n
21 double multiply(double x, int n){
22     return (double)x*n;
23 }
24
25 // 函式divide回傳x/n
26 double divide(double x, int n){
27     return (double)x/n;
28 }
```

接著按照第二三四段要求寫出powerpower()

2. Write a function **double powerpower(...)** that can compute $(x^n)^m$, $(x^n)^m$, $(x/n)^m$, where `powerpower()` must use four parameters: a pointer to function, one double and two integers.
3. Also remember to write functions `divide()`, `multiply()` and `power()`
4. use **typedef to define a new type F** which is a pointer to function

程式碼如下

```
1
2 // 首先依照要求定義好函式指標F
3 typedef double (*F)(double,int);
4
5 /*
6  * 函式powerpower()回傳所選的func(x,n)^m
7  * 接下來依照第二點宣告powerpower()
8  * 首先處理m為負數的情況，將x替換成倒數，m轉正再傳入函式中
9  * 再來考慮到m=0的情況，直接回傳1
10  * 計算func(x,n)，可能是power,multiply或divide x與n
11  * 再使用迴圈的方式計算func(x,n)^m
12  */
13 double powerpower(F func,double x,int n,int m){
14     if(m < 0)
15         return powerpower(func, (double)(1/x), n, -m);
16     if(m == 0)
17         return 1.0;
18     double result = func(x,n);
19     for(int i = 1;i<m;i++){
20         result *= func(x,n);
21     }
22     return result;
23 }
```

接著按照第五段要求寫main函式

When executing your program, you can choose the values for x, n, and m by using `argc` and `argv`.

程式碼如下

```
1
2  /*
3   * 使用argc,argv，所需參數存在argv字串陣列中
4   * 其中我另argv[1] = x
5   *      argv[2] = n
6   *      argv[3] = m
7   * 因為其為字串，在傳參時我使用atoi轉為整數、atof轉為浮點數
8   * 然後題目並無提到func的選擇方式，所以我將三種結果全部打印
9   */
10 int main(int argc, char *argv[]){
11     printf("----- powerpower(power,x,n,m) -----\\n");
12     printf("%lf\\n",powerpower(power,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
13     printf("----- powerpower(multiply,x,n,m) -----\\n");
14     printf("%lf\\n",powerpower(multiply,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
15     printf("----- powerpower(divide,x,n,m) -----\\n");
16     printf("%lf\\n",powerpower(divide,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
17 }
```

以下是完整程式碼

```
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<string.h>
4  #include<math.h>
5
6  typedef double (*F)(double,int);
7
8  double power(double x, int n){
9      if(n<0) return power((double)(1/x),-n);
10     if(n == 0) return 1.0;
11     if(n == 1) return x;
12     return (double)x*power(x,n-1);
13 }
14
15 double multiply(double x, int n){
16     return (double)x*n;
17 }
18
19 double divide(double x, int n){
20     return (double)x/n;
21 }
22
23 double powerpower(F func,double x,int n,int m){
24     if(m < 0) return powerpower(func, (double)(1/x), n, -m);
25     if(m == 0) return 1.0;
26     double result = func(x,n);
27     for(int i = 1;i<m;i++){
28         result *= func(x,n);
29     }
30     return result;
31 }
32
33
34 int main(int argc, char *argv[]){
35     printf("----- powerpower(power,x,n,m) -----\\n");
36     printf("%lf\\n",powerpower(power,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
37     printf("----- powerpower(multiply,x,n,m) -----\\n");
38     printf("%lf\\n",powerpower(multiply,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
39     printf("----- powerpower(divide,x,n,m) -----\\n");
40     printf("%lf\\n",powerpower(divide,atof(argv[1]),atoi(argv[2]),atoi(argv[3])));
41 }
```