|  |
| --- |
| 得 分 |
|  |

資料結構

HWK

四則運算

B10702057盧昱達

* 程式碼

#include <iostream>

#include <fstream>

#include <sstream>

#include <string>

#define MAX\_SIZE 50

using namespace std;

class Stack

{

private:

string s[MAX\_SIZE];

double d[MAX\_SIZE];

int top;

int base;

public:

Stack();

void push(string);

void push\_d(double);

string pop();

double pop\_d();

bool is\_empty();

};

Stack::Stack()

{

top = 0;

base = 0;

}

void Stack::push(string ch)

{

s[top++] = ch;

}

string Stack::pop()

{

return s[--top];

}

void Stack::push\_d(double ch)

{

d[top++] = ch;

}

double Stack::pop\_d()

{

return d[--top];

}

bool Stack::is\_empty()

{

if (top <= base) return true;

else return false;

}

string dateinput()

{

string t;

cout << "請輸入：";

cin >> t;

return t;

}

string infix\_to\_postfix(string infix)

{

string postfix;

Stack a;

for ( int i = 0; i < infix.size(); i++)

{

string op;

op = infix[i];

switch(infix[i]){

case '0':case '1':case '2':case '3':case '4':

case '5':case '6':case '7':case '8':case '9':

postfix += infix[i];

break;

case '+':case '-':

postfix += " ";

if(a.is\_empty())

{

a.push(op);

}

else

{

postfix += a.pop() + " ";

if(!a.is\_empty())

{

postfix += a.pop() + " ";

}

a.push(op);

}

break;

case '\*':case '/':

postfix += " ";

if(!a.is\_empty())

{

string t;

t = a.pop();

if (t == "+" || t == "-")

{

a.push(t);

a.push(op);

}

else

{

postfix += t + " ";

a.push(op);

}

}

else

{

a.push(op);

}

break;

}

}

while(!a.is\_empty())

{

postfix += " " + a.pop();

}

return postfix;

}

double run\_postfix(string postfix)

{

double number, x, y;

string op, op1, op2;

Stack a;

for (int i = 0; i < postfix.size(); i++)

{

op = postfix[i];

switch(postfix[i])

{

case '0':case '1':case '2':case '3':case '4':

case '5':case '6':case '7':case '8':case '9':

number = number \* 10 + (postfix[i] - '0');

break;

case ' ':

a.push\_d(number);

number = 0;

break;

case '+':

x = a.pop\_d();

y = a.pop\_d();

a.push\_d(y + x);

i++;

break;

case '-':

x = a.pop\_d();

y = a.pop\_d();

a.push\_d(y - x);

i++;

break;

case '\*':

x = a.pop\_d();

y = a.pop\_d();

a.push\_d(y \* x);

i++;

break;

case '/':

x = a.pop\_d();

y = a.pop\_d();

if ((float)x == 0)

{

cout << "不符邏輯運算錯誤" << endl;

system("pause");

exit(1);

}

else

{

a.push\_d((float)y / x);

i++;

}

break;

}

}

return a.pop\_d();

}

int main ( void )

{

//宣告

string infix, postfix;

double out;

//讀黨

infix = dateinput();

cout << "Infix = " << infix << endl;

//後制式

postfix = infix\_to\_postfix(infix);

cout << "Postfix = " << postfix << endl;

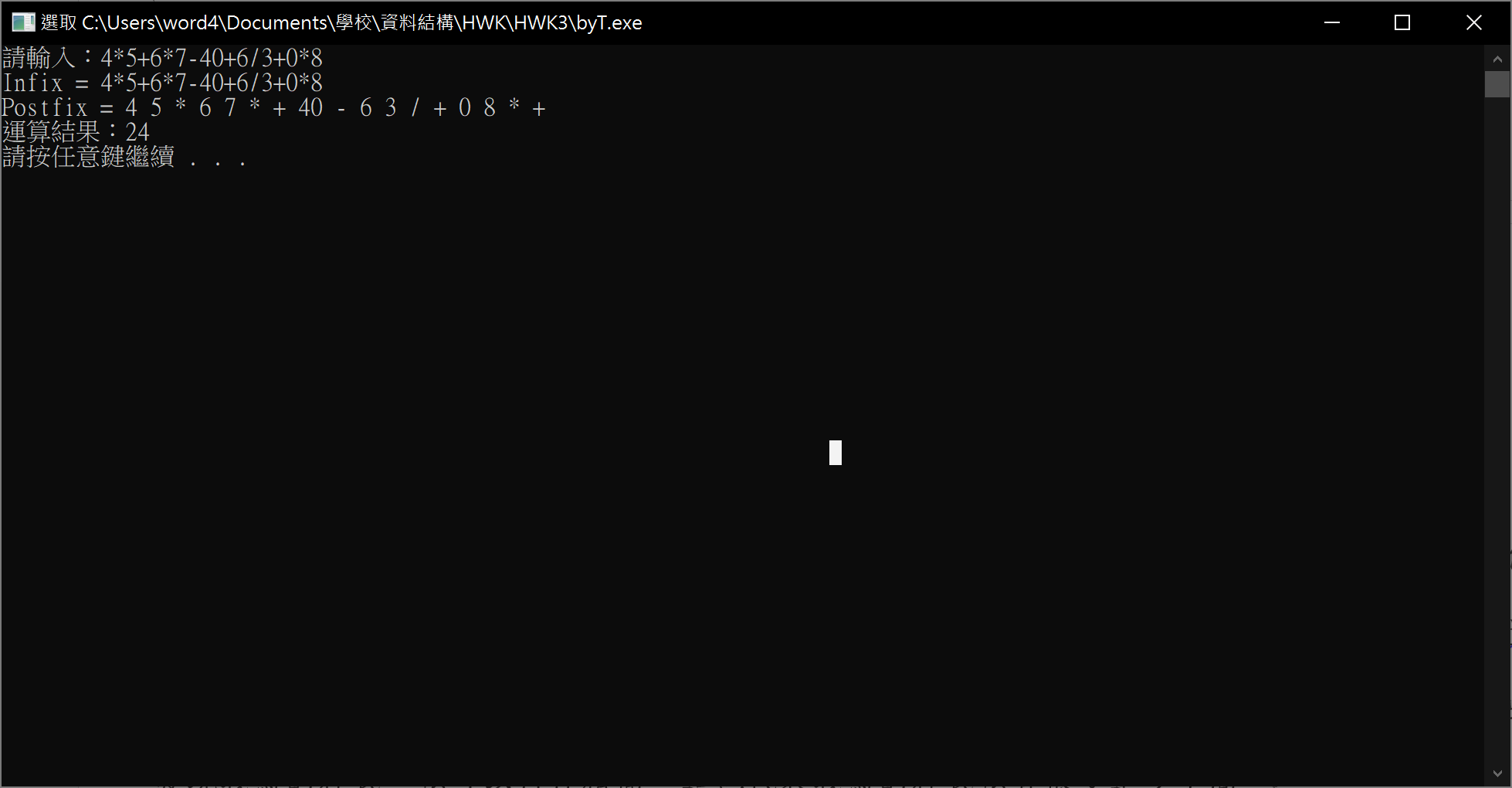
//最後運算

out = run\_postfix(postfix);

cout << "運算結果：" << out << endl;

system("pause");

* 測試結果



* 實作心得

這次實作中，原本我先使用了字串分割(strtok)寫這題，讓運算符號與數字分別儲存，實作上多了許多小步驟，雖最終也得道我要的答案，但在上了老師的課輔，收穫不少，便改採用新的寫法，在轉後置式上先讓他維持原先的資料型態，最後再用後置示轉換資料型態，並作運算，獲得更簡便的寫法，但在寫這些作業時，我會發現我寫的程式，還可以再作精簡，這也代表我寫的程式還有很多進步空間。