Phill C++-2

3/8/2023

procedure programming

Function 函式

y= f(x)

- no input, no output
- single input, no output
- multiple inputs, single out

No input, no output

```
#include <iostream>
using namespace std;

void say_hello(){ //宣告, 放進倉庫, 不會被執行
  cout <<" hello"<<endl;
}


int main()  //主函式
{
    say_hello(); //呼叫
    return 0;
}
```

```
#include <iostream>
using namespace std;

void say_hello(string name){ //參數 argument
  cout <<" hello,"<<name << endl;
}


int main()  //主函式
{
    say_hello("Phill"); //呼叫
    //cout << name;
```

```
    return 0;
}
```

multiple inputs, no output

```
#include <iostream>
using namespace std;

void say_hello(string greeting, string name){ //參數 arguments
  cout <<greeting<<","<<name << endl;
}


int main()  //主函式
{
    say_hello("Hi","Phill"); //呼叫
    //cout << name;
    return 0;
}
```

```
#include <iostream>
using namespace std;

int phill_sum(int a, int b){ //參數 arguments
  int c;
  c= a+b;
  return c; // 傳回去
}


int main()  //主函式
{
    int z;
    z = phill_sum(1,2);
    cout <<z;
    return 0;
}
```

函式→ 練習

- 圓 : r, pi
- 傳回面積
- 主程式裡印出結果

```
#include <iostream>
using namespace std;


double circle(int r, double pi){
  double c;
  c=r*r*pi;
  return c;
}


int main()
{
   double area;
   area=circle(2,3.1415);
   cout << area;
   return 0;

}
```

翻轉字串的函式

- input:string , output:string

```
#include <iostream>
#include <string>

using namespace std;

string reverse_string(string input_str) {
    string reverse_str;

    //去耦合 decouple
    for (int i = input_str.length() - 1; i >= 0; i--) {
        reverse_str += input_str[i];
    }


    return reverse_str;
}

int main() {
    string input_str="hello Phill";
    string reversed_str;

    reversed_str = reverse_string(input_str);
```

```
    cout << "after reverse:" << reversed_str << endl;

    return 0;
}
```

```
i= i+1 // = -> assignment operator, ==
i += 1 //遞運算
i *=1
i /=1
```

- 作業 1

  階乘 - n

  主程式 n!

  3! = 1*2*3

  n!

- 作業 2

  function

  回文 ： 對稱

  寫一個檢查回文的 function, string , return → 0 不是 1 是