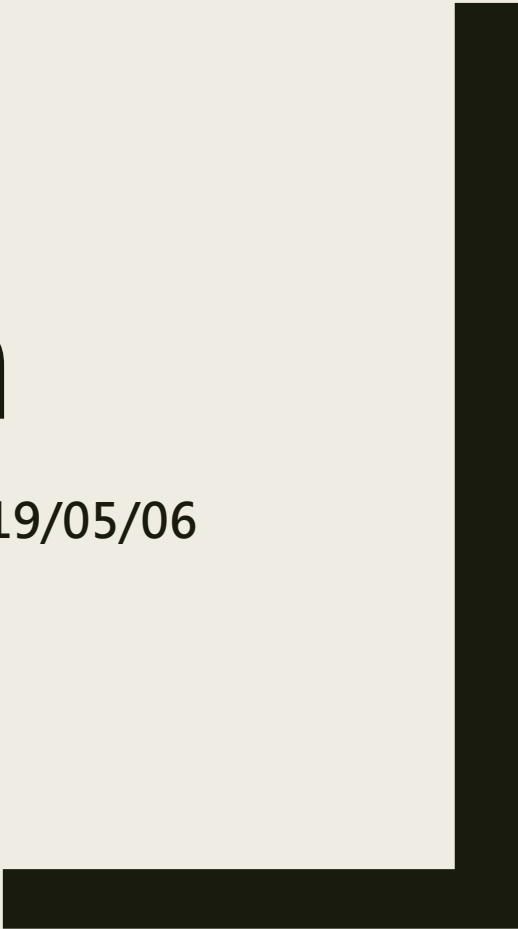




# Midterm

2019/05/06



# 考試規則

1. 可以翻閱你覺得有幫助的書(最多兩本)(教室會斷網!!)
2. 不得作弊，違者依校規論處
3. 若有格式錯誤的情形，會將該題分數  $\times 0.8$  計算
4. 本次練習都只需繳交 **Header file**  
不得更改 `main_Q1.c` 中任何內容  
繳交時請自行將 Header file 檔名改為 學號－題號  
如：0756704-1.h  
註：不需變更 `ifndef`, `define`, `include` 的檔名
5. 總共只有一次繳交機會，請務必確認格式正確後，再舉手找助教繳交。
6. 行動電子產品 (手機、平板電腦等等)請收在包包內，不要放在桌面上或使用它。

# 期中考規則

- 題目總共五題，總分採計得分最高的四題
- 助教只會解釋題目意思，或是電腦的突發狀況，例如突然關機，其他問題同學請自行處理，像是當機、無法編譯
- 考試中，請勿使用其他電子產品、儲存裝置或是配戴耳機，避免產生不必要的誤會
- 考試中，請勿交談或交換參考書籍
- 外出上廁所也請不要攜帶行動裝置
- 如有違規且不聽勸阻者，本次考試將以0分採計

# Header file

- Header file contains function declarations and macro definitions to be shared between several source files.
- For example

The image shows two code editors side-by-side. The left editor, titled 'main.c', contains the following code:

```
1  #include <stdio.h>
2  #include "add.h"
3
4  int main(void)
5  {
6      int number;
7      scanf("%d", &number);
8
9      number = do_something(number);
10
11     printf("%d", number);
12
13     return 0;
14 }
15
```

The right editor, titled '\*add.h', contains the following code:

```
1  #ifndef add_H
2  #define add_H
3
4  int do_something(int n){
5      return n + 1;
6  }
7
8  #endif
9
```

Red arrows indicate the relationship between the two files: one arrow points from the `#include "add.h"` line in `main.c` to the header file, and another arrow points from the `do_something` function call in `main.c` to the function definition in `*add.h`.

- In Header file (\*.h) , you can add any function or declaration **except** main function

# Q1 - IntegerSet

## Description

Create class **IntegerSet** for which each object can hold **integers that is smaller than the value of the set**. Represent the set internally as an array of bool values. Element  $a[i]$  is true if integer  $i$  is in the set. Element  $a[j]$  is false if integer  $j$  is not in the set. The default constructor initializes a set to the so-called “empty set,” i.e., a set for which all elements contain false.

Provide member function for the common set operations. Provide an **InsertElement** member function which takes integer  $k$  as input and set the  $a[k]$  to true.

Provide a **printSet** member function that prints a set as a list of numbers, which is separated by “\_”, in ascent order. Print only those elements that are present in the set.

Finally, please provide the **highest** and the **lowest** function that will return the **highest** element and the **lowest** element from the IntegerSet.

# Q1 - IntegerSet

## Description(cont.)

The derived class **EvenIntegerSet** should inherit from the base class **IntegerSet**. Also, provide the member function **filterSet** that will print a set as a list of only **even numbers**, which is separated by “\_”, in ascent order.

# Q1 – IntegerSet

## Description of input value

The input will ask user to enter the size(int) for the **IntegerSet** and the **EvenIntegerSet**. Then, it will ask user to enter the element for the set. And the element is **int** type and the range is  $0 \sim \text{size}-1$ .

- The total number of the input elements(count) is fixed to **10 as shown in the main file**.
- The input size should be **bigger than** the count.
- The **value** of each element should be **less than** the **value of** size.

# Q1 – IntegerSet

## Description of output value

The output will show the results from the member function **printSet**, **filterSet**, and the **highest** and **lowest function**.

- No need line feed for the **printSet** and **filterSet**.
- No need to implement operator overloading in **highest** and **lowest** function.
- The **highest** function and the **lowest** only return **int**.



# Example

## ■ Sample Input1

11

8 7 6 3 5 4 5 8 4 6

## ■ Sample Output1

3\_4\_5\_6\_7\_8\_

4\_6\_8\_

8

3

## ■ Sample Input2

12

1 1 11 3 3 3 3 1 1 1

## ■ Sample Output2

1\_3\_11\_

11

1

# Q2 – Cute Animals in the Zoo

## Description

Create a `ZooAnimal` class which contains four private variables “`name`”(char), “`weightDate`”(int), “`weight`”(int). Create the constructor that copies the character string parameter into the name field, and then assigns the two `integer` parameters to `weightDate`, and `weight` respectively. If there's no parameter provided when initializing the object, set default values of “Nameless” for `name`, 101 (January 1) for `weightDate`, and 100 for `weight`. Write a member function `reptName` that returns the `ZooAnimal`'s name. Write a member function `reptWeight` that returns the `ZooAnimal`'s weight. Write an inline function `reptWeightDate` to return `weightDate` data member. Write a member function `daysSinceLastWeighed`. This function has a single integer parameter `today` and returns an integer number of days since the animal was last weighted. (Assume each month has 30 days) As the animals measure their body weight at least every year, so **don't** consider the number of days is over 360 (a year, 30\*12).

# Q2 – Cute Animals in the Zoo

## Description

Write an inline function `changeWeightDate` to set the `weightDate` variable.

Write an inline function `changeWeight` to set the `weight` variable.

Create an inheritance hierarchy containing derive classes `LargeAnimal` that inherit from base class `ZooAnimal`, which includes a data member “`species`” of type `char` and a data member “`cageMinimumVolume`” of `float` type.

Write a member function `reptName` for `LargeAnimal`, try to return the object’s “`species`” instead of returning the name by the `ZooAnimal` class’ `reptName` member function.

For `LargeAnimal` class, Write a member function `reptCageMinVolume` that returns the `LargeAnimal`’s `CageMinVolume`.

# Q2 – Cute Animals in the Zoo

## Description of input value

- First value represents the animal selected
- Second value represents the date of today variable in integer format
  - E.g. 205 represents Feb 5, 1112 represents Nov 12
  - Assume there's 30 days in each month
  - Don't consider the number of days is over 360 (a year,  $30 \times 12$ ). As the cute animals in the zoo measure their weight at least every year.



# Example

## ■ Sample Input1

2 308

## ■ Sample Output1

Name: titanosaur

Weight: 90000

Weight Date: 208

Since Last Weight Date: 30

## ■ Sample Input2

3 1224

## ■ Sample Output2

Name: Lin\_Wang

Species: Mammal

Cage Minimum Volume: 28.5

Weight: 5500

Weight Date: 1225

Since Last Weight Date: 359

Reweigh...

Weight: 5000

Weight Date: 1224

# Q3 – Convert polar coordinate

## Description

Create a class named “**Transform**” which contains a function “set\_polar” which converts the rectangular coordinate value into polar coordinate value. The x, y coordinates will be double type, in range -500~+500 .The formula for finding polar coordinates is:

$$c = \sqrt{x^2 + y^2} \quad q = \arctan\left(\frac{y}{x}\right)$$

Use the concept you learn in **pointer**. You don't need to return or print **anything** out. Create a class named “**Distance**” which contains a function “get\_distance” which *computes* and *returns* the distance between two coordinates(x,y)and (i,j). [Hint] Feel free to use **atan()** in math.h or any other function to compute the coordinates.

# Example

## ■ Sample Input1

3 4 0 0

## ■ Sample Input2

-300 -400 300 400

## ■ Sample Output1

5\_0.927295

5

## ■ Sample Output2

500\_0.927295

1000

# Q4 – Really Simple Vector

## Description

Create a simple template class name `SimpleVector` which simulates different vector types and contains functions below:

- Constructor: set the initial vector. If the vector length is not assigned, then set the maximum capacity of vector to 5.
- Destructor: delete the vector.
- `my_empty`: return 1 if the vector element is empty, otherwise return 0.
- `my_size`: return the number of elements that exist value in the vector.
- `my_reverse`: reverse all the elements in the vector by their positions.
- `my_clear`: set all existing elements to `'\0'` that means no value.
- `my_extend`: increases the capacity to “two times of the current capacity” of the dynamically created array.
- `my_push_back`: assign the value to the last element of the array with the user input value. If the vector’s size isn’t enough, then extend the array.
- `my_pop_back` : pop out the last element of the vector.



# Q4 – Really Simple Vector

## Description

- `my_insert`: insert an element with value and index provided by user. Extend the array if the vector's size isn't enough. If the index input is out of the vector's size, then don't do anything.
- `my_set`: replace an element with value and index provided by user. If the index input is out of the vector's size, then don't do anything.
- `print`: display all the elements in the vector separated by “\_”.

Besides, try to overload the operator `[]` that returns the value of the vector for the specific index. E.g. `sample[5]` returns the fifth index of the sample vector. If the input index is out of the vector length, then return the last element of the vector.

# Q4 – Really Simple Vector

## Description of input value

Your program is supposed to be able to handle “ int ” and “ double ” format

- First value represents the datatype of test data :  
if it's 1, the input will be integer; otherwise, double
- Then use the command below to manipulate the vector until command=0.

Command	Description
1	my_push_back
2	my_insert
3	my_set
4	my_reverse
5	my_pop_back
6	my_clear
0	Print the info and quit

# Example

## ■ Sample Input1

1  
1 1 0  
1 2 0  
1 3 0  
1 4 0  
2 22 2  
3 33 3  
5 0 0  
4 0 0  
0 0 0

## ■ Sample Output1

33\_22\_2\_1\_  
33\_22\_2\_1\_  
Not Empty

## ■ Sample Input2

2  
1 1 0  
1 1.5 0  
6 0 0  
1 -99.9 0  
0 0 0

## ■ Sample Output2

-99.9\_  
-99.9\_  
Not Empty

# Q5 – Personal information

## Description

In this part, you will need to construct a multiple inheritance hierarchy as the following image. Declare a parent class named “**Person**” which contains all the basic features of person, such as name, age and sex, we will need to initial these data members. Next, derive two classes “**Student**” and “**Teacher**” inherit from base class Person. Class Student includes a data member score in float type and in class Teacher includes a data member title in string type, we will need to initial these data members. However, in both classes you also need to implement a **display** function to print all the data member in following format:

[Student] Order in “name sex age score”. [Teacher] Order in “name sex age title”.

Add “\_” (an underline) between each output value.

Need line feed in the end(需要换行).

# Q5 – Personal information

## Description

Use multiple inheritance to declare a class named “**Graduate**” which derives by both Student and Teacher classes, in this class contains a data member wage in float type. In this part you will need to overload increment (++) operator to increase one amount of wage in Graduate object, and also decrement (--) operator to decrease one amount of wage in Graduate object. **You must implement in both prefix and postfix expression.**

Last, please overload (<<) in order to get the output in the following format:

Order in “name sex age score title wage”. Add “\_” (an underline) between each output value. **No need line feed in the end(不需要換行).**

**[Remind]** If your output format in this question is wrong, you may not get any scores, so please make sure it's correct.

# Q5 – Personal information

## Description of input value

- Sex will be 'f' and 'm' in char type.
- Title will be 'professor', 'assistant' in string type.
- Age will be in range 15~105 in int type.
- Score will be in range 0~100 in double type.
- Wage will be in range 0~10000 in double type.

# Example

## ■ Sample Input1

Alice f 18 98.6

Bob m 27 professor

Cathy f 24 assistant 87.8 159.7

## ■ Sample Input2

Dorothy f 17 83.9

Eric m 49 professor

Fade m 100 assistant 98 9996

## ■ Sample Output1

Alice\_f\_18\_98.6

Bob\_m\_27\_professor

Cathy\_f\_24\_87.8\_assistant\_159.7

Cathy\_f\_24\_87.8\_assistant\_158.7

Cathy\_f\_24\_87.8\_assistant\_159.7

Cathy\_f\_24\_87.8\_assistant\_159.7

Cathy\_f\_24\_87.8\_assistant\_158.7

## ■ Sample Output2

Dorothy\_f\_17\_83.9

Eric\_m\_49\_professor

Fade\_m\_100\_98\_assistant\_9996

Fade\_m\_100\_98\_assistant\_9995

Fade\_m\_100\_98\_assistant\_9996

Fade\_m\_100\_98\_assistant\_9996

Fade\_m\_100\_98\_assistant\_9995

Q&A