

義守大學 93 學年度研究所碩士在職專班考試試題

系所別	資訊工程學系碩士在職專班	考試日期	93/4/24
考試科目	程式設計	總頁數	2

※此為試題卷，請將答案填寫在答案卷內，未寫於答案卷內者，不予計分。

※不可使用計算機

1. 何謂 **static memory allocation**？何謂 **dynamic memory allocation**？今若以 **dynamic memory allocation** 方式要求 100 個整數空間，則在 C 語言中該寫甚麼指令？若使用 C++ 語言該寫甚麼指令？ 10%

2. 在物件導向程式設計中，一個 class 的 members 宣告可分為 **public**, **protected**, 和 **private** 三類，試述這三類 members 在存取(access)上有何區別？為何要做如此區分，其功用為何？ 10%

3. 下列為一遞迴(recursive)函數，若在主程式中呼叫 **rec_prn(4)**，則其輸出結果為何？ 10%

```
void rec_prn(int n)
{
    if (n > 0) {
        rec_prn(n-1);
        printf("%d ", n);
        rec_prn(n-1);
    }
}
```

4. 下列程式的執行結果為何？ 10%

```
#include <iostream>
#include <string>
using namespace std;
class C {
public:
    C( const char* n ) {
        name = n;
        cout << name << " constructing.\n";
    }
    ~C() {
        cout << name << " destructing.\n";
    }
private:
    string name;
};

int main() {
    C c0( "first object" );
    {
        C c2( "second object" );
        cout << '\n';
    }
    C* ptr = new C( "third object" );
    delete ptr;
    return 0;
}
```

義守大學 93 學年度研究所碩士在職專班考試試題

系所別	資訊工程學系碩士在職專班	考試日期	93/4/24
考試科目	程式設計	總頁數	2

※此為試題卷，請將答案填寫在答案卷內，未寫於答案卷內者，不予計分。

※不可使用計算機

5. 假設現有一 linked list，其中每一個 node(節點) 的結構如下： 30%

```
struct node
{
    int data;
    node *next;
};
```

試以 C、C++ 或 Java 語言設計一 function，用以找出該 linked list 中所有 data 值為質數的 node，並將這些質數值輸出到螢幕上。

此 function 的 function heading 宣告如下，其中 first 為一 pointer 參數，指向該 linked list 的第一個 node。

```
void print_prime(node *first)
{
    . . .
}
```

(你必須略說明你的程式做法，若使用 Java 語言，則上述語法略有差異，你可依語言做適當調整)

6. 以 C++、Java 或其他物件導向程式語言，設計一代表長方形(rectangle)的 class，該 class 中需包含長方形的長(length)和寬(width)以及該長方形的左上角 x,y 座標，另需要以下幾個 member functions: 30%

- (a). **default constructor** 用來設定 length, width 和 x,y 的初值。(設 length, width, x, y 的內定值分別為 1, 1, 0, 0)
- (b). **area function** 用來計算長方形的面積。
- (c). **set function** 用來更改 length,width 和 x,y 的值。
- (d). **perimeter function** 用來計算長方形的周長

(此題中，你只需要設計 class 的 definition 部分，並不需要設計 member functions 的 function body，另外，你需說明你的 member functions 中各個參數的意義。)