國立臺北科技大學 110 學年度碩士班招生考試

系所組別：2300 資訊工程系碩士班

第二節　程式設計　試題

注意事項：

1. 本試題共六題，共 100 分。
2. 不必抄題，作答時請將試題題號及答案依照順序寫在答案卷上。
3. 全部答案均須在答案卷之答案欄內作答，否則不予計分。

Problem 1 [13%] [1-1, 1%] [1-2 ~ 1-5, each 3%]

The following Python code is binary search function. The parameters of the function are described as follows:

    (a) *data*: a sorted list to be searched.

    (b) *left*: the minimum index of the data.

    (c) *right*: the maximum index of the data.

    (d) *key*: the searching target.

The return value of the search function is the index of the *key*. If the *key* not in the *data*, the return value is "-1".

```
1   def search(data, left, right, key):
2       mid = (left+right)//2
3       if data[mid]==key:
4           return mid
5       if left==right:
6           return -1
7       if data[mid]>key:
8           return search(data, left, mid, key)
9       else:
10          return search(data, mid, right, key)
11  def f(x):
12      print(search([1, 5, 9, 14, 23, 26], 0, 5, x))
```

(1) (1%) If f(5) is executed, what is the output?

(2) (3%) If f(5) is executed, please write the line numbers of the running sequence for this code. (11, 12, 1, 2, 3, 5, _____ 3, 4)

(3) (3%) If f(26) is executed, what kind of the error is reported while this code is running?

(4) (3%) If f(26) is executed, please fix the error of this code (Line 8 and Line 10), so that the correct output should be 5.

(5) (3%) Assume the functionality of search() is correct. What is the output of the following Python code f()?

```
1   def f():
2       data = [1, 5, 9, 14, 23, 26]
3       my_dict ={}
4       my_dict[9]=search(data, 0, 5, 9)
5       my_dict[1]=search(data, 0, 5, 1)
6       my_dict[14]=search(data, 0, 5, 14)
7       print(my_dict)              # (i)(2%)
8       print(data[1:5:2])          # (ii)(1%)
```

Problem	Answer
1-1	
1-2	
1-3	
1-4	
1-5-i	
1-5-ii	

Please copy the above answer table to your answer sheet

Problem 2 [12%] [2-1, 3%] [2-2-i, 2-2-ii, each 3%] [2-3-i, 1%, 2-3-ii, 2%]

The following Python code will print the Fibonacci sequence. The output of f(8) is "0, 1, 1, 2, 3, 5, 8, 13,".

```
1   def fibonacci(n):
2       if n<=0:
3           return 0
4       elif n==1 or n==2:
5           return 1
6       else:
7           return fibonacci(n-1)+fibonacci(n-2)
8
9   def f(x):
10      i = 0
11      while i<x:
12          print(fibonacci(i), end=', ')
13          i = i + 1
```

(1) (3%) If f(8) is executed, how many times will fibonacci(4) be called?

(2) (6%) Please complete the following "Loop version" Python code of the Fibonacci function. The output of h(8) is "0, 1, 1, 2, 3, 5, 8, 13,".

   (i) (3%) Line 4: _____

   (ii) (3%) Line 5: _____

```
1   def fibonaccii(n):
2       data = [0 for i in range(0, n+1)]
3       data[1]=1
4       for i in range(_____, n+1):        # (i)
5           data[i] = _____      # (ii)
6       return data
7
8   def h(x):
9       data = fibonaccii(x)
10      for i in range(x):
11          print(data[i], end=', ')
```

(3) (3%) The following code is modified from the code of the above problem (1).

```
1   def fibonacci3(n, data):
2       if n>5: print(n,' * ', end='')
3       if n<=0: return 0
4       elif n==1 or n==2: data[n] = 1
5       elif data[n]==0:
6           data[n] = fibonacci3(n-1, data)+fibonacci3(n-2, data)
7       return data[n]
8
9   def p(x):
10      data = [0 for i in range(x+1)]
11      for i in range(x):
12          fibonacci3(i, data)
13
```

   (i) (1%) What is the output of p(8)?

   (ii) (2%) Please compare fibonacci3 with Fibonacci function according to the computing time and space.

Problem	Answer
2-1	
2-2-i	
2-2-ii	
2-3-i	
2-3-ii	

Please copy the above answer table to your answer sheet

Problem 3 [14%, each 2%]

Please trace the following C program and answer the output of each printf() statement for problems 3-1~3-7.

```c
#include <stdio.h>
#include <string.h>

int f1(int a, int b) {
    return (a||!b) + (~a&b);
}

float f2(int num) {
    float t = (float)num/2 + num%2 + 1.0/num;
    return t;
}

int f3(){
    enum FOOD {fish=-1, eggs, meat=3, milk, bean};
    return(eggs+milk)/2 > meat? fish: bean;
}

int f4(int n) {
    int sum = 0;
    for (int i=0; i<n; i++) {
        if (i < 3)
            continue;
        sum += i;
    }
    return sum;
}

int f5(char* s){
    int i=0, t=0;
    while (s[i]!='\0') {
        switch (s[i]) {
            case 's': t += 1;
                break;
            case 'c': t += 2;
                break;
            case 'e': t += 3;
                break;
            default: t += 1;
                break;
        }
        i++;
    }
    return t;
}
```

```c
int f6(int x[3], int y[3][3]) {
    int t=0, n=3;
    for (int i=0; i<n; i++)
        for (int j=0; j<n; j++)
            t += x[i]*y[i][j];
    return t;
}

char* f7(char s1[], char s2[]) {
    char s[20]="";
    strncat(s, s1, 4);
    strcat(s, s2);
    strncpy(s1, s, 10);
    return s1;
}

int main(int argc, char *argv[]) {
    int a[3][3]={{1,1,1},{2,2,2},{3,3,3}};
    int b[3]={1, 2, 2};
    char s1[]="Hello", s2[]="World!";

    printf("%d\n", f1(1, 2));          /*     Problem 3-1     */
    printf("%3.1f\n", f2(5));          /*     Problem 3-2     */
    printf("%d\n", f3());              /*     Problem 3-3     */
    printf("%d\n", f4(5));             /*     Problem 3-4     */
    printf("%d\n", f5("science"));     /*     Problem 3-5     */
    printf("%d\n", f6(b, a));          /*     Problem 3-6     */
    printf("%s\n", f7(s1, s2));        /*     Problem 3-7     */
    return 0;
}
```

Problem	Answer
3-1	
3-2	
3-3	
3-4	
3-5	
3-6	
3-7	

Please copy the above answer table to your answer sheet.

Problem 4 [11%] [4-1 ~ 4-3, each 2%] [4-4 ~ 4-8, each 1%]

Please trace the following C program and fill the blanks with correct statements or write down the outputs of the statements.

```c
#include <stdio.h>
#include <stdlib.h>
typedef struct node { int data; struct node *next; } NODE;

int showTop(NODE* top) {
    return top->data;
}

void pushStack(_____ top, int data) {          /* Problem 4-1 (2 points) */
    NODE* node;

    node = (_____) malloc(sizeof(NODE));       /* Problem 4-2 (2 points) */
    if(node == NULL) {
        printf("Memory allocation failure!\n");
        exit(1);
    }
    node->data = data;
    node->next = *top;
    *top = node;
}

void showStack(NODE* top) {
    NODE* node = top;
    while(node != NULL) {
        printf("%d ", node->data);
        node = _____;                       /* Problem 4-3 (2 points) */
    }
    printf("\n");
}

int popStack(_____ top) {                       /* Problem 4-4 (1 points) */
    int value;
    NODE* node = *top;
    if(node == NULL) {
        printf("Stack is empty!\n");
        exit(1);
    }
    value = (int) (node->data);
    _____ = node->next;                         /* Problem 4-5 (1 points) */
    free(node);
    return value;
}
```

```c
int main(void) {
    NODE* stackTop=NULL;

    pushStack(&stackTop, 5);
    pushStack(&stackTop, 6);
    pushStack(&stackTop, 2);
    showStack(stackTop);                      /* Problem 4-6 (1 points) */
    printf("%d\n", popStack(&stackTop));      /* Problem 4-7 (1 points) */
    printf("%d\n", showTop(stackTop));        /* Problem 4-8 (1 points) */
    return 0;
}
```

Problem	Answer
4-1	
4-2	
4-3	
4-4	
4-5	
4-6	
4-7	
4-8	

Please copy the above answer table to your answer sheet.

Problem 5 [25%] [5-1 ~ 5-4, each 1%] [5-5 ~ 5-10, each 2%] [5-11 ~ 5-13, each 3%]

Please trace the following C++ program and answer the std::cout outputs of each statement

from problems 5-1 ~ 5-13.

```cpp
#include <iostream>
using namespace std;
class Transportation    {                          /* 交通工具 Abstract Class */
public:
        ~Transportation(){cout << "~Transportation()" << endl;};
        virtual void move() = 0;
};
class Car: public Transportation {                 /* 車 */
private:
        int wheel;                                 /* 車輪數量 */
public:
        ~Car(){cout << "~Car()" << endl;};
        void setWheel(int value){ wheel = value; };
        int    getWheel(){ return wheel; };
        void move(){ cout << "run" << endl; };
        virtual void printTopSpeed() { cout << "100" << endl; };
};
class Sedan: public Car {                           /* 四門轎車 */
public:
        Sedan(){ setWheel(4); };
        Sedan(Sedan& other){ setWheel(other.getWheel()); };
        ~Sedan(){ cout << "~Sedan()" << endl; };
        void move(){ cout << "run slow" << endl; };
        void printTopSpeed() { cout << "200" << endl; };
};
class Coupe: public Car {                           /* 雙門轎跑車 */
public:
        Coupe(){ setWheel(2); };
        Coupe(Coupe& other){ setWheel(other.getWheel()); };
        ~Coupe(){ cout << "~Coupe()" << endl; };
        void move(){ cout << "run fast" << endl; };
        void printTopSpeed() { cout << "300" << endl; };
};
class AirPlane: public Transportation {             /* 飛機 */
public:
        ~AirPlane(){ cout << "~AirPlane()" << endl; };
        void move(){ cout << "fly" << endl; };
        void printTopSpeed() { cout << "900" << endl; };
};
class Jet {                                         /* 噴射引擎飛機 */
public:
        Jet(){};
        ~Jet(){ cout << "~Jet()" << endl; };
        void move(){ cout << "fly fast" << endl; };
        virtual void printTopSpeed() { cout << "1000" << endl; };
};
class FlyingCar: public Car, public AirPlane {      /* 飛天車 */
public:
```

```cpp
        FlyingCar(){ setWheel(4); };
        ~FlyingCar(){ cout << "~FlyingCar()" << endl; };
        void move(){cout << "fly slow" << endl;};
        virtual void printTopSpeed() { cout << "500" << endl; };
};
int main(){
    Sedan        sedan;
    Coupe        coupe;
    Jet          jet;
    FlyingCar flyingCar;
    sedan.move();              /* Problem 5-1 (1 points) */
    coupe.move();              /* Problem 5-2 (1 points) */
    flyingCar.move();          /* Problem 5-3 (1 points) */
    jet.move();                /* Problem 5-4 (1 points) */
    Car      *p1, *p2;
    Sedan    *p3;
    AirPlane *p4;
    p1 = new Car();   p2 = new Sedan(sedan);   p3 = new Sedan();    p4 = new FlyingCar();
    p1->printTopSpeed();       /* Problem 5-5   (2 points) */
    p2->printTopSpeed();       /* Problem 5-6   (2 points) */
    p3->printTopSpeed();       /* Problem 5-7   (2 points) */
    p4->printTopSpeed();       /* Problem 5-8   (2 points) */
    coupe.printTopSpeed();     /* Problem 5-9   (2 points) */
    jet.printTopSpeed();       /* Problem 5-10 (2 points) */
    delete p1;                 /* Problem 5-11 (3 points) */
    delete p2;                 /* Problem 5-12 (3 points) */
    delete p3;                 /* Problem 5-13 (3 points) */
    delete p4;
    return 0;
}
```

Problem	Answer
5-1	
5-2	
5-3	
5-4	
5-5	
5-6	
5-7	
5-8	
5-9	
5-10	
5-11	
5-12	
5-13	

Please copy the above answer table to your answer sheet

注意：背面尚有試題

Problem 6 [25%]

Trace following C++ program and fill the blanks with correct statements or parameters for 6-1 to 6-5 (2% each), 6-6 to 6-9 (3% each), and 6-11 (1%). Then, write down the outputs for 6-10 and 6-12 (1% each).

```cpp
#include <iostream>
#include <string>
#define PI 3
using namespace std;
class MathVector {
private:
    double *_content;
    int _size;
public:
    MathVector (){    }
    MathVector (double* input, int size = 0) {
        _content = new double[size];
        _size = size;
        for (int i = 0;i < _size;i++) {
            _content[i] = input[i];
        }
    }
    MathVector (_____input) {              /* Problem 6-1 (2 points) */
        _content = new double[input._size];
        _size = input._size;
        for (int i = 0;i < _size;i++) {
            _content[i] = input._content[i];
        }
    }
    ~ MathVector(){
        if(_content != NULL) {
            _____;                     /* Problem 6-2 (2 points) */
        }
    }
    MathVector & operator = (_____input) {     /* Problem 6-3 (2 points) */
        if(_content != NULL) {
            delete [] _content;
        }
        _____;                     /* Problem 6-4 (2 points) */
        _size = input._size;
        for(int i=0;i<_size;i++){
            _content[i] = input._content[i];
        }
        return _____;                      /* Problem 6-5 (2 points) */
    }
    double & component(int index) const {
        return _content[index - 1];
    }
};
class Shape{
protected:
    MathVector _center;
public:
    Shape(MathVector center){
        _center = center;
    }
    virtual double perimeter() _____;        /* Problem 6-6 (3 points) */
    _____~Shape() {        }                  /* Problem 6-7 (3 points) */
};
class Circle _____ {                   /* Problem 6-8 (3 points) */
private:
    double _radius;
public:
    Circle(double radius, MathVector center) : _____ { /* Problem 6-9 (3 points) */
        _radius = radius;
    }
    ~Circle() override {        }
    double perimeter() const override {
        return 2*_radius*PI;
    }
};
int main(){
    double y[2] = {1, 90};
    MathVector output = MathVector(y, 2);
    cout << output.component (2) << endl;        /* Problem 6-10 (1 points) */
    Shape* output2 = new _____(5 , output);     /* Problem 6-11 (1 points) */
    cout << output2->perimeter() << endl;        /* Problem 6-12 (1 points) */
    delete output2;
    return 0;
}
```

Problem	Answer
6-1	
6-2	
6-3	
6-4	
6-5	
6-6	
6-7	
6-8	
6-9	
6-10	
6-11	
6-12	

Please copy the above answer table to your answer sheet