

國立勤益科技大學 101 學年度研究所碩士班招生筆試試題卷

所別：電子工程系

組別：多媒體遊戲組

科目：計算機程式（甲）

准考證號碼：□□□□□□□□（考生自填）

考生注意事項：

一、考試時間 100 分鐘。

二、

三、

試題一：〈 40 分〉

請在程式中宣告 a 矩陣和 b 矩陣，並寫出相乘的程式，

$$a = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}, \quad b = \begin{bmatrix} 5 & 0 & 8 \\ 2 & 1 & 7 \\ 5 & 9 & 3 \end{bmatrix}。$$

試題二：〈 40 分〉

請用亂數函數產生一長度為 10 個元素的數列，數的範圍為 0~999，並將此數列元素由小至大排序印出。

試題三：〈 40 分〉

設計一電話費的計算程式，主程式讀入每位客戶名字（以 30 個人為例）、短途通話秒數、長途通話秒數及手機通話秒數，放在結構 struct 的資料中，將結構 struct 資料傳至副程式，副程式計算每個人的通話費再傳回主程式並印出每個人的通話費，假設短途通話每分 1 元，長途通話每分 2 元，及手機通話每分 3 元。

試題四：〈 40 分〉

從鍵盤讀入一個小於十位數的整數，將此整數每一位數值取出依序放入一個整數陣列，並由高位數值依序印出，例如讀入數值 123456，則將 1 至 6 依序放入陣列中並印出。

試題五：〈 40 分〉

超級市場舉辦水果免費裝的活動，限定一個購物塑膠袋無論裝多少水果都可以免費帶回家，但是一個購物袋最多裝 8 公斤，超過 8 公斤購物袋就會破裂，則一個水果也帶不走。所以我們希望用電腦去算出購物袋在 8 公斤內能夠裝到最划算的水果，請寫一程式能夠算出怎麼樣裝水果才能夠價值最多，請顯示出所裝的水果編號，數量與總共價值多少錢，其中各種水果已事先以塑膠袋裝袋且不得拆解，袋裝水果的編號，重量與價值如下表。

編號	水果名	重量	價值
0	水梨	4 公斤	45
1	蘋果	5 公斤	57
2	橘子	2 公斤	22
3	葡萄	1 公斤	11
4	西瓜	6 公斤	65

國立勤益科技大學 101 學年度研究所碩士班招生筆試答案卷

所別： 電子工程系

組別： 多媒體遊戲組

科目： 計算機程式 (甲)

准考證號碼： (考生自填)

考生注意事項：

一、考試時間 100 分鐘。

二、

三、

試題一：〈 40 分〉

請寫出 a 矩陣和 b 矩陣相乘的程式， $a = \begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{bmatrix}$ ， $b = \begin{bmatrix} 5 & 0 & 8 \\ 2 & 1 & 7 \\ 5 & 9 & 3 \end{bmatrix}$ 。

C++

```
#include <iostream>
int a[][] = { {
    1, 2, 3 }, {
    4, 5, 6 }, {
    7, 8, 9 }
};
int b[][] = { {
    5, 0, 8 }, {
    2, 1, 7 }, {
    5, 9, 3 }
};
int void () {
for (int m = 0; m < 2; m++) {
    for (int p = 0; p < 2; p++) {
        for (int n = 0; n < 2; n++) {
            c[m][p] = c[m][p] + a[m][n] * b[n][p];
        }
    }
}
for (int i = 0; i < 2; i++) {
    for (int j = 0; j < 2; j++) {
        cout << c[i][j] << " ";
    }
}
```

```
    Cout << \n ;
return 0 ;
}
```

java

```
public class JavaHomeWork {
public static void main(String[] args) {
    int a[][] = { {
        1, 2, 3 }, {
        4, 5, 6}, {
        7, 8, 9}
    };
    int b[][] = { {
        5, 0, 8}, {
        2, 1, 7}, {
        5, 9, 3}
    };
    for (int m = 0; m < 2 ; m++) {
        for (int p = 0; p < 2 ; p++) {
            for (int n = 0; n < 2 ; n++) {
                c[m][p] = c[m][p] + a[m][n] * b[n][p];
            }
        }
    }
    for (int i = 0; i < 2 ; i++) {
        for (int j = 0; j < 2 ; j++) {
            System.out.print(c[i][j] + ",");
        }
        System.out.println("");
    }
}
}
```

試題二：〈 40 分〉

請用亂數函數產生一長度為 10 個元素的數列，數的範圍為 0~999，並將此數列元素由小至大排序印出。

```
#include "stdlib.h"
#include "stdio.h"
#include "time.h"
```

```

#define range 1000;

void swap(int *, int*);

int main(int argc, char* argv[])
{
    int ary[10];
    srand(time(NULL));

    for(int i=0; i<10; i++)
    {
        ary[i]=rand()%range;
    }

    for(int i = 10 - 1; i >= 0; i--)
    {
        for(int j = 1; j <= i; j++)
        {
            if( ary[j-1] > ary[j] )
                swap(&ary[j-1], &ary[j]);
        }
    }

    for(int i=0; i<10; i++)
    {
        printf("%5d", ary[i]);
    }
    printf("\n");
    return 0;
}

void swap(int* R, int* L)
{
    int temp;
    temp = *R;
    *R = *L;
    *L = temp;
}

java
import java.io.*;
import java.lang.Math;

public class factorial {
    public static void main(String[] args) throws IOException
    {
        int ary[] = new int[10];

```

```

for(int i=0 ; i<10 ; i++)
{
    ary[i]= (int)(Math.random()*1000);
}

for(int i = 10 - 1; i >= 0; i--)
{
    for(int j = 1; j <= i; j++)
    {
        if( ary[j-1] > ary[j] )
        {
            int temp;
            temp = ary[j-1];
            ary[j-1] = ary[j];
            ary[j] = temp;
        }
    }
}

for(int i=0 ; i<10 ; i++)
{
    System.out.print(ary[i]+" ");
}
System.out.println();
}
}

```

試題三：〈 40 分〉

設計一電話費的計算程式，主程式讀入每位客戶名字(以 30 個人為例)、短途通話秒數、長途通話秒數及手機通話秒數，放在結構 struct 的資料中，將結構 struct 資料傳至副程式，副程式計算每個人的通話費再傳回主程式並印出每個人的通話費，假設短途通話每分 1 元，長途通話每分 2 元，及手機通話每分 3 元。

```

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

```

```

struct phone{
    int s1,s2,s3,total;
    char name[5];
};

void test3(phone ss[], int size)
{
    int j;
    for(j=0;j<size;j++)
    {
        ss[j].total=ss[j].s1*1+ss[j].s2*2+ss[j].s3*3;
    }
}

int main(int argc, char *argv[])
{
    struct phone s[30];
    int i,j;
    for(i=0;i<2;i++)
    {
        printf("%d's name\n",i+1);
        scanf("%s",&s[i].name);
        printf("local\n");
        scanf("%d",&s[i].s1);
        printf("long dis\n");
        scanf("%d",&s[i].s2);
        printf("mobile\n");
        scanf("%d",&s[i].s3);
    }
    test3(s,2);
    printf("name  local long mobil total\n");
    printf("-----\n");
    for(j=0;j<30;j++)
    {
        s[j].total=s[j].s1+s[j].s2+s[j].s3;
        printf("%-5s %-6d %-6d %-6d %-6d\n",s[j].name,s[j].s1,s[j].s2,s[j].s3,s[j].total);
    }
}

```

```
    system("PAUSE");
    return EXIT_SUCCESS;
}
```

試題四：〈 40 分〉

從鍵盤讀入一個小於十位數的整數，將此整數每一位數值取出依序放入一個整數陣列，並由高位數值依序印出，例如讀入數值 123456，則將 1 至 6 依序放入陣列中並印出。

```
#include <cstdlib>
#include <iostream>
#include <stdio.h>
using namespace std;
int main(int argc, char *argv[])
{
    int a,b,sum=0,i=0,j,darray[10];
    printf("請輸入一個小於 10 位數的阿拉伯數字：");
    scanf("%d",&a);
    b=a;
    while(b!=0)
    {
        darray[i]=b%10;
        b=b/10;
        i++;
    }
    for(j=i-1;j>=0;j--)
    printf("結果：%d ",darray[j]);
    printf("\n\n");
    system("PAUSE");
    return EXIT_SUCCESS;
}
```


試題五：〈 40 分〉

超級市場舉辦水果免費裝的活動,限定一個購物塑膠袋無論裝多少水果都可以免費帶回家,但是一個購物袋最多裝 8 公斤,超過 8 公斤購物袋就會破裂,則一個水果也帶不走。所以我們希望用電腦去算出購物袋在 8 公斤內能夠裝到最划算的水果,請寫一程式能夠算出怎麼樣裝水果才能夠價值最多,請顯示出所裝的水果編號,數量與總共價值多少錢,其中各種水果已事先以塑膠袋裝袋且不得拆解,袋裝水果的編號,重量與價值如下表。

編號	水果名	重量	價值
0	水梨	4 公斤	45
1	蘋果	5 公斤	57
2	橘子	2 公斤	22
3	葡萄	1 公斤	11
4	西瓜	6 公斤	65

C++

```
#include <stdio.h>
#include <stdlib.h>
#define LIMIT 8 // 重量限制
#define N 5 // 物品種類
#define MIN 1 // 最小重量
struct body {
    char name[20];
    int size;
    int price;
};
typedef struct body object;
int main(void) {
```

```

int item[LIMIT+1] = {0};
int value[LIMIT+1] = {0};
object a[] = {"水梨", 4, 45},
             {"蘋果", 5, 57},
             {"橘子", 2, 22},
             {"葡萄", 1, 11},
             {"西瓜", 6, 67}};

int i, s;
for(i = 0; i < N; i++) {
    for(s = a[i].size; s <= LIMIT; s++) {
        int p = s - a[i].size;
        int newvalue = value[p] + a[i].price;
        if(newvalue > value[s]) { // 找到階段最佳解
            value[s] = newvalue;
            item[s] = i;
        }
    }
}
printf("物品\t 價格\n");
for(i = LIMIT; i >= MIN; i = i - a[item[i]].size) {
    printf("%s\t%d\n",
           a[item[i]].name, a[item[i]].price);
}
printf("合計\t%d\n", value[LIMIT]);
return 0;
}

```

java

```

class Fruit {
    public final String name;
    public final int size;
    public final int price;
    public Fruit(String name, int size, int price) {
        this.name = name;
        this.size = size;
        this.price = price;
    }
}

```

```

public class Knapsack {
    public static void main(String[] args) {
        final int MAX = 8;
        final int MIN = 1;
        int[] item = new int[MAX+1];
        int[] value = new int[MAX+1];
        Fruit fruits[] = {
            new Fruit("李子", 4, 45),
            new Fruit("蘋果", 5, 57),
            new Fruit("橘子", 2, 22),
            new Fruit("草莓", 1, 11),
            new Fruit("甜瓜", 6, 67));
        for(int i = 0; i < fruits.length; i++) {
            for(int s = fruits[i].size; s <= MAX; s++) {
                int p = s - fruits[i].size;
                int newValue = value[p] + fruits[i].price;
                if(newValue > value[s]) { // 找到階段最佳解
                    value[s] = newValue;
                    item[s] = i;
                }
            }
        }
        System.out.println("物品\t價格");
        for(int i = MAX; i >= MIN; i -= fruits[item[i]].size) {
            System.out.println(fruits[item[i]].name+
                "\t" + fruits[item[i]].price);
        }
        System.out.println("合計\t" + value[MAX]);
    }
}

```