

# 《程式設計概要》

注意：若題目有模糊之處，請在答題時說明相關假設。

設計一個質因數分解的程式，我們收集下面的資訊：

1. 質數是指一個正整數除了1及自己以外，沒有其它的因數。
2. 一個正整數 $n$ ，若有質因數 $P \geq 0$ ，則 $P \leq \sqrt{n}$ 。

為了加速程式運作，我們做下列規劃：

1. 以陣列PrimeAry來（容量2000，由小到大依序）儲存已知的質數，PrimeNum用來說明最後一個（最大）質數的陣列位置。
  2. 陣列PrimeAry啟始時，僅儲存質數2。
- 根據上列敘述，撰寫程式回答下列問題：

- 一、(一)假設目前陣列PrimeAry儲存所有比 $n$ 小的質數，撰寫一函數IsPrime( $n$ )判別 $n$ 是否為質數。譬如：IsPrime(3)回覆True，IsPrime(4)回覆False。（15分）
- (二)撰寫一函數BuildPrime( $m$ )，使用IsPrime函數建立陣列PrimeAry，使其擁有小於或等於 $m$ 的所有質數。譬如：BuildPrime(5)將使PrimeAry陣列依序儲存2, 3, 5，而BuildPrime(10)將使PrimeAry陣列依序儲存2, 3, 5, 7。（15分）

試題評析	本題仍遵循往年試題，考的是迴圈觀念，使用的主題是質因數。這個主題在上課時就有強調其重要性。本題利用陣列增加其計算速度，而題目有指定函數規格，按照上課所教的方法即可輕鬆完成答案。
考點命中	《高點·高上程式設計講義第一回迴圈單元》，許振明編撰，頁75。

答：

(一)java語言

```
public boolean IsPrime(int n){
    int i=0;

    if(n<2) return false;
    while(PrimeAry[i]*PrimeAry[i]<=n){
        if(n % PrimeAry[i] == 0) return false;
        i++;
    }
    return true;
}
```

(二)java語言

```
public void BuildPrime(int m){
    int i, p;

    if(m<2) return;
    PrimeAry[0]=2;
    PrimeNum=0;
    if(m>2){
        for(i=3; i<=m; i+=2){
            if(IsPrime(i)) PrimeAry[++PrimeNum]=i;
        }
    }
}
```

【版權所有，重製必究！】

- 二、(一) 假設目前陣列PrimeAry儲存所有比 $n$ 小的質數，撰寫一遞迴函數 (Recursive function)  $\text{FactorTimes}(n, p)$ ，回覆正整數 $n$ 含有質數 $p$ 的因數次數。譬如： $792 = 2^3 \times 3^2 \times 11$ ， $\text{FactorTimes}(792, 2)$  回覆 3， $\text{FactorTimes}(792, 3)$  回覆 2， $\text{FactorTimes}(792, 5)$  回覆 0， $\text{FactorTimes}(792, 11)$  回覆 1。(15分)
- (二) 假設目前陣列PrimeAry儲存所有比 $n$ 小的質數，撰寫一函數PrimeFactor( $n$ ) 求出正整數 $n$ 的質因數及其因數次數，並將結果以字串傳回。譬如： $\text{PrimeFactor}(20)$  將回傳字串" $2^2 * 5$ "， $\text{PrimeFactor}(72)$  將回傳字串" $2^3 * 3^2$ "，而 $\text{PrimeFactor}(97)$  將回傳字串" $97$ "。(15分)

<b>試題評析</b>	如今年預測的題型考直接遞迴。遞迴的撰寫技巧在講義中記載的非常清楚，也符合上課中所提示的遞迴題型。因式分解的部分只要質因數觀念正確，利用迴圈就可輕易寫出程式內容。不過要注意題目有指定列印的輸出格式，小心注意即可拿高分。
<b>考點命中</b>	《高點·高上程式設計講義第一回迴圈單元與遞迴單元》，許振明編撰，頁122。

**答：**

(一) java 語言

```
public int FactorTimes(int n, int p){
    if(n<2 || p<2) return 0;

    if(n % p !=0) return 0;
    else if(n==p) return 1;
    else return(FactorTimes(n/p, p)+1);
}
```

(二) java 語言

```
public String PrimeFactor(int n){
    String str = null;
    String op = null;
    int i,a,b;

    i=0;
    while(n!=1 && i<= PrimeNum){
        if(n%PrimeAry[i]==0){
            a=PrimeAry[i];
            b=FactorTimes(n, PrimeAry[i]);
            if(str!=null) op="*";
            if(b==1){
                if(str==null) str="" + a;
                else str=str+op+a;
            }
            else{
                if(str==null) str = a+"^"+b;
                else str=str+op+a+"^"+b;
            }
            System.out.println("str: " + str);
            n/=PrimeAry[i];
        }
        i++;
    }
    return str;
}
```

【版權所有，重製必究！】

三、(一)若將上述方法（主程式除外）封裝成一個類別PrimeClass，請實踐上述規劃的1.及2.。  
（10分）

(二)使用網頁技術來建立下列系統：要求使用者輸入一個小於10000的正整數，判別它是否在要求的範圍內；該判別方法（名稱為Check\_Inp\_Val）必須在客戶端（Client）完成，不可在伺服器（Server）完成。若輸入值在範圍內，則呼叫伺服器端處理函數mainPrime；否則在網頁直接發布相關訊息，中斷處理（不呼叫處理函數mainPrime）。（20分）

(三)請撰寫處理函數mainPrime，讓系統可以求出輸入數值的質因數分解。（10分）

試題評析	今年考題然出了web程式與物件導向程式，試題題型完全在上課時完預測命中。本題答案須將前兩題所寫函數利用類別封裝起來，再加上題目所給的資料成員即可完成。至於javascript與web程式則利用(一)中所寫的類別整合進來就可輕鬆完成。今年的考題難易適中，高上同學如用功可得85以上的高分。
考點命中	《高點·高上程式設計講義第二回javascript、web單元與物件導向單元》，許振明編撰，頁32、45、63。

**答：**

(一)java語言

```
public class PrimeClass {
    int PrimeAry[] = new int[2000];
    int PrimeNum;
    public PrimeClass(){
        PrimeAry[0]=2;
        PrimeNum = -1;
    }
    public boolean IsPrime(int n){
        int i=0;
        if(n<2) return false;
        while(PrimeAry[i]*PrimeAry[i]<=n){
            if(n % PrimeAry[i] == 0) return false;
            i++;
        }
        return true;
    }
    public void BuildPrime(int m){
        int i, p;
        if(m<2) return;
        PrimeAry[0]=2;
        PrimeNum=0;
        if(m>2){
            for(i=3; i<=m; i+=2){
                if(IsPrime(i)) PrimeAry[++PrimeNum]=i;
            }
        }
    }
    public int FactorTimes(int n, int p){
        if(n<2 || p<2) return 0;
        if(n % p !=0) return 0;
        else if(n==p) return 1;
        else return(FactorTimes(n/p, p)+1);
    }
    public String PrimeFactor(int n){
        String str = null;
```

【版權所有，重製必究！】

```

String op = null;
int i,a,b;
i=0;
while(n!=1 && i<= PrimeNum){
    if(n%PrimeAry[i]==0){
        a=PrimeAry[i];
        b=FactorTimes(n, PrimeAry[i]);
        if(str!=null) op="*";
        if(b==1){
            if(str==null) str="" +a;
            else str=str+op+a;
        }
        else{
            if(str==null) str = a+"^"+b;
            else str=str+op+a+"^"+b;
        }
        System.out.println("str:" + str);
        n/=PrimeAry[i];
    }
    i++;
}
return str;
}
}

```

(二)html程式

```

<!DOCTYPE html>
<html>
<head>
<meta charset="BIG5">
<title>Insert title here</title>
</head>
<body>
<script>
function Check_Inp_Val(){
    if(document.computeNum.num.value>=10000 || document.computeNum.num.value<1)
        alert("輸入錯誤");
    else document.computeNum.submit();
}
</script>
<form name=computeNum action='prime.jsp'>
請輸入一個正整數: <input Type='text' name='num'><BR>
<input Type='button' Value='確定' onClick='Check_Inp_Val()'><br>
</form>
</body>
</html>

```

(三)

1.在類別PrimeClass內加入下列函數

```

public String mainPrime(int num){
    BuildPrime(num);
    return PrimeFactor(num);
}

```

【版權所有，重製必究！】

## 2.jsp程式

```

<% @ page language="java" contentType="text/html; charset=BIG5"
    pageEncoding="BIG5"
    import="com.abc.PrimeClass"
%>
<!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=BIG5">
<title>Insert title here</title>
</head>
<body>
<%
    int num = Integer.parseInt(request.getParameter("num"));
    PrimeClass pc = new PrimeClass();
    out.println(num + " = " + pc.mainPrime(num));
%>
</body>
</html>

```

## 3.執行結果

請輸入一個正整數:

70=2\*5\*7

\*\*\*\* 附件: 另提供第一題與第二題的C語言版本完整範例 \*\*\*\*

```

#include <stdio.h>
#include <stdbool.h>

```

```

int PrimeAry[2000]={0};
int PrimeNum=-1;
char string[2000]="";

```

```

bool IsPrime(int n){
    int i=0;

    if(n<2) return false;
    while(PrimeAry[i]*PrimeAry[i]<=n){
        if(n % PrimeAry[i] == 0) return false;
        i++;
    }
    return true;
}

```

```

void BuildPrime(int m){
    int i, p;

    if(m<2) return;
    PrimeAry[0]=2;
    PrimeNum=0;

```

【版權所有，重製必究！】

```

    if(m>2){
        for(i=3; i<=m; i+=2){
            if(IsPrime(i)) PrimeAry[++PrimeNum]=i;
        }
    }
}

void Print(void){
    int i;
    for(i=0; i<=PrimeNum; i++){
        printf("%4d", PrimeAry[i]);
        printf("\n");
    }
}

int FactorTimes(int n, int p){
    if(n<2 || p<2) return 0;
    if(n % p !=0) return 0;
    else if(n==p) return 1;
    else return(FactorTimes(n/p, p)+1);
}

char *PrimeFactor(int n){
    char op[2]="";
    int i,a,b;
    i=0;
    while(n!=1 && i<= PrimeNum){
        if(n%PrimeAry[i]==0){
            printf("%d ", a=PrimeAry[i]);
            printf("%d\n",b=FactorTimes(n, PrimeAry[i]));
            if(strcmp(string,"")) strcpy(op, "*");
            if(b==1) sprintf(string, "%s%s%d", string, op, a);
            else sprintf(string, "%s%s%d^%d", string, op, a, b);
            n/=PrimeAry[i];
        }
        i++;
    }
    return string;
}

int main(void){
    BuildPrime(100);
    Print();
    if(IsPrime(19)) printf("true\n");
    else printf("false\n");
    printf("%d\n", FactorTimes(792, 5));
    printf("%s\n",PrimeFactor(60));
    return 0;
}

```

【版權所有，重製必究！】