

1. 給定右側程式片段，哪個 n 值不會造成超過陣列 A 的存取範圍？

(A) 69
(B) 89
(C) 98
(D) 202

```
int i, n, A[100];
scanf ("%d", &n);
for (i=0; i!=n; i=i+1) {
    A[i] = i;
    i = i + 1;
}
```

2. 給定右側函式 $f()$ ，當執行 $f(10)$ 時，最終回傳結果為何？

(A) 1
(B) 3840
(C) -3840
(D) 執行時導致無窮迴圈，不會停止執行

```
int f (int i) {
    if (i>0)
        if ((i/2)%2)==0)
            return f(i-2)*i;
        else
            return f(i-2)*(-i);
        else
            return 1;
}
```

3. 給定右側程式片段，**for** 迴圈總共會執行幾次？

(A) 8
(B) 32
(C) 64
(D) 128

```
int i, j=0;
for (i=0; i<128; i=i+j) {
    j=i+1;
}
```

4. 給定右側程式，若已知輸出的結果為 [1][2][3][5][4][6]，程式中的 (?) 應為下列何者？

(A) $j < i$
(B) $j > i$
(C) $j \leq i$
(D) $j \geq i$

```
int main() {
    int i, j;
    for (i=0; i<5; i=i+1) {
        for (j=0; (?); j=j+2) {
            printf ("%d", i+j);
        }
    }
}
```

5. 給定右側程式，當程式執行完後，輸出結果為何？

- (A) 1 2 3 4 5 6 7 8
(B) 7 5 3 1 2 4 6 8
(C) 7 5 3 2 1 4 8 6
(D) 8 7 6 5 4 3 2 1

```
int A[8] = {8,7,6,5,4,3,2,1};
int main () {
    int i, j;
    for (i=0; i<8; i=i+1){
        for (j=i; j<7; j=j+1){
            if (A[j] > A[j+1]){
                A[j] = A[j] + A[j+1];
                A[j+1] = A[j] - A[j+1];
                A[j] = A[j] - A[j+1];
            }
        }
    }
    for(i=0; i<8; i=i+1){
        printf ("%d ",A[i]);
    }
}
```

6. 給定右側函式 $f()$ ，已知 $f(14)$ 、 $f(10)$ 、 $f(6)$ 分別回傳 25、18、10，函式中的(?) 應為下列何者？

- (A) $(n+1)/2$
(B) $n/2$
(C) $(n-1)/2$
(D) $(n/2)+1$

```
int f(int n) {
    if (n < 2) {
        return n;
    }
    else {
        return (n + f((?)));
    }
}
```

7. 給定右側程式，當程式執行完後，輸出結果為何？

- (A) 1
(B) 2
(C) 3
(D) 4

```
int main() {
    int a[5] = {9, 4, 3, 5, 3};
    int b[10] = {0,1,0,1,0,1,0,1,0,1};
    int c = 0;
    for (int i=0; i<5; i=i+1)
        c = c + b[a[i]];
    printf ("%d", c);
    return 0;
}
```

8. 給定右側程式片段，當程式執行完後，輸出結果為何？

(A) 9
(B) 18
(C) 27
(D) 30

```
int Q[200];
int i, val=0;
int count=0;
int head=0, tail=0;
for (i=1; i<=30; i=i+1) {
    Q[tail] = i;
    tail = tail + 1;
}
while (tail > head + 1) {
    val = Q[head];
    head = head + 1;
    count = count + 1;
    if (count == 3) {
        count = 0;
        Q[tail] = val;
        tail = tail + 1;
    }
}
printf ("%d", Q[head]);
```

9. 給定右側程式，當程式執行完後，輸出結果為何？

(函式 **f(a)** 回傳小於浮點數 **a** 的最大整數，但是回傳型態仍為浮點數。)

(A) 0.000000
(B) 1.000000
(C) 1.666667
(D) 2.000000

```
int main() {
    float x=10, y=3;
    if ((0.5*x/y - f(0.5*x/y)) == 0.5){
        printf("%f\n",f(0.5*x/y)-1);
    }
    else if ((0.5*x/y-f(0.5*x/y)) < 0.5){
        printf("%f\n",f(0.5*x/y));
    }
    else
        printf("%f\n",f(0.5*x/y)+1);
    return 0;
}
```

10. 給定右側程式，當程式執行完後，輸出結果為何？

(A) 2
(B) 3
(C) -2
(D) -3

```
void f(int x, int y) {
    int tem = x;
    x = y;
    y = tem;
}

int main() {
    int x = 2, y = 3;
    f(x, y);
    printf("%d", (x-y)*(x+y)/2);
    return 0;
}
```

11. 右側程式正確的輸出應該如下：

```
*
***
*****
*****
*****
```

在不修改右側程式之第4行及第7行程式碼的前提下，最少需修改幾行程式碼以得到正確輸出？

(A) 1
(B) 2
(C) 3
(D) 4

```
1  int k = 4;
2  int m = 1;
3  for (int i=1; i<=5; i=i+1) {
4      for (int j=1; j<=k; j=j+1) {
5          printf (" ");
6      }
7      for (int j=1; j<=m; j=j+1) {
8          printf ("*");
9      }
10     printf ("\n");
11     k = k - 1;
12     m = m + 1;
13 }
```

12. 給定一陣列 `a[10]={ 1, 3, 9, 2, 5, 8, 4, 9, 6, 7 }`，i.e., `a[0]=1, a[1]=3, ..., a[8]=6, a[9]=7`，以 `f(a, 10)` 呼叫執行右側函式後，回傳值為何？

(A) 1
(B) 2
(C) 7
(D) 9

```
int f (int a[], int n) {
    int index = 0;
    for (int i=1; i<=n-1; i=i+1) {
        if (a[i] >= a[index]) {
            index = i;
        }
    }
    return index;
}
```

13. 給定一整數陣列 $a[0]$ 、 $a[1]$ 、...、 $a[99]$ 且 $a[k]=3k+1$ ，以 $\text{value}=100$ 呼叫以下兩函式，假設函式 **f1** 及 **f2** 之 **while** 迴圈主體分別執行 $n1$ 與 $n2$ 次 (i.e, 計算 **if** 敘述執行次數，不包含 **else if** 敘述)，請問 $n1$ 與 $n2$ 之值為何？ 註： $(\text{low} + \text{high})/2$ 只取整數部分。

```
int f1(int a[], int value) {
    int r_value = -1;
    int i = 0;
    while (i < 100) {
        if (a[i] == value) {
            r_value = i;
            break;
        }
        i = i + 1;
    }
    return r_value;
}
```

```
int f2(int a[], int value) {
    int r_value = -1;
    int low = 0, high = 99;
    int mid;
    while (low <= high) {
        mid = (low + high)/2;
        if (a[mid] == value) {
            r_value = mid;
            break;
        }
        else if (a[mid] < value) {
            low = mid + 1;
        }
        else {
            high = mid - 1;
        }
    }
    return r_value;
}
```

- (A) $n1=33, n2=4$
 (B) $n1=33, n2=5$
 (C) $n1=34, n2=4$
 (D) $n1=34, n2=5$

14. 經過運算後，右側程式的輸出為何？

- (A) 1275
 (B) 20
 (C) 1000
 (D) 810

```
for (i=1; i<=100; i=i+1) {
    b[i] = i;
}
a[0] = 0;
for (i=1; i<=100; i=i+1) {
    a[i] = b[i] + a[i-1];
}
printf ("%d\n", a[50]-a[30]);
```

15. 函數 **f** 定義如下，如果呼叫 **f(1000)**，指令 **sum=sum+i** 被執行的次數最接近下列何者？

(A) 1000
(B) 3000
(C) 5000
(D) 10000

```
int f (int n) {
    int sum=0;
    if (n<2) {
        return 0;
    }
    for (int i=1; i<=n; i=i+1) {
        sum = sum + i;
    }
    sum = sum + f(2*n/3);
    return sum;
}
```

16. List 是一個陣列，裡面的元素是 element，它的定義如右。List 中的每一個 element 利用 **next** 這個整數變數來記錄下一個 element 在陣列中的位置，如果沒有下一個 element，**next** 就會記錄 -1。所有的 element 串成了一個串列 (linked list)。例如在 **list** 中有三筆資料

| 1 | 2 | 3 |
|------------|------------|------------|
| data = 'a' | data = 'b' | data = 'c' |
| next = 2 | next = -1 | next = 1 |

它所代表的串列如下圖



```
struct element {
    char data;
    int next;
}

void RemoveNextElement (element
list[], int current) {
    if (list[current].next != -1) {
        /*移除 current 的下一個 element*/
        
    }
}
```

RemoveNextElement 是一個程序，用來移除串列中 **current** 所指向的下一個元素，但是必須保持原始串列的順序。例如，若 **current** 為 3 (對應到 **list[3]**)，呼叫完 **RemoveNextElement** 後，串列應為



請問在空格中應該填入的程式碼為何？

(A) **list[current].next = current ;**
(B) **list[current].next = list[list[current].next].next ;**
(C) **current = list[list[current].next].next ;**
(D) **list[list[current].next].next = list[current].next ;**

17. 請問以 **a(13,15)** 呼叫右側 **a()** 函式，函式執行完後其回傳值為何？

- (A) 90
- (B) 103
- (C) 93
- (D) 60

```
int a(int n, int m) {
    if (n < 10) {
        if (m < 10) {
            return n + m ;
        }
        else {
            return a(n, m-2) + m ;
        }
    }
    else {
        return a(n-1, m) + n ;
    }
}
```

18. 一個費式數列定義第一個數為0 第二個數為1 之後的每個數都等於前兩個數相加，如下所示：

0、1、1、2、3、5、8、13、21、34、55、89...

右列的程式用以計算第 N 個($N \geq 2$)費式數列的數值，請問 (a) 與 (b) 兩個空格的敘述(statement)應該為何？

- | | |
|------------------------------|------------|
| (A) (a) $f[i]=f[i-1]+f[i-2]$ | (b) $f[N]$ |
| (B) (a) $a = a + b$ | (b) a |
| (C) (a) $b = a + b$ | (b) b |
| (D) (a) $f[i]=f[i-1]+f[i-2]$ | (b) $f[i]$ |

```
int a=0;
int b=1;
int i, temp, N;

...
for (i=2; i<=N; i=i+1) {
    temp = b;
    (a) _____ ;
    a = temp;
    printf ("%d\n", (b) _____ );
}
```

19. 請問右側程式輸出為何？

- (A) 1
- (B) 4
- (C) 3
- (D) 33

```
int A[5], B[5], i, c;
...
for (i=1; i<=4; i=i+1) {
    A[i] = 2 + i*4;
    B[i] = i*5;
}
c = 0;
for (i=1; i<=4; i=i+1) {
    if (B[i] > A[i]) {
        c = c + (B[i] % A[i]);
    }
    else {
        c = 1;
    }
}
printf ("%d\n", c);
```

20. 給定右側 **g()** 函式，**g(13)** 回傳值為何？

- (A) 16
- (B) 18
- (C) 19
- (D) 22

```
int g(int a) {
    if (a > 1) {
        return g(a - 2) + 3;
    }
    return a;
}
```

21. 定義 **a[n]** 為一陣列(array)，陣列元素的指標為 0 至 n-1。若要將陣列中 **a[0]** 的元素移到 **a[n-1]**，右側程式片段空白處該填入何運算式？

- (A) **n+1**
- (B) **n**
- (C) **n-1**
- (D) **n-2**

```
int i, hold, n;
...
for (i=0; i<=____ ; i=i+1) {
    hold = a[i];
    a[i] = a[i+1];
    a[i+1] = hold;
}
```


22. 給定右側函式 **f1()** 及 **f2()**。 **f1(1)** 運算過程中，以下敘述何者為錯？

- (A) 印出的數字最大的是 4
- (B) **f1** 一共被呼叫二次
- (C) **f2** 一共被呼叫三次
- (D) 數字 2 被印出兩次

```
void f1 (int m) {
    if (m > 3) {
        printf ("%d\n", m);
        return;
    }
    else {
        printf ("%d\n", m);
        f2(m+2);
        printf ("%d\n", m);
    }
}

void f2 (int n) {
    if (n > 3) {
        printf ("%d\n", n);
        return;
    }
    else {
        printf ("%d\n", n);
        f1(n-1);
        printf ("%d\n", n);
    }
}
```

23. 右側程式片段擬以輾轉除法求 **i** 與 **j** 的最大公因數。請問 **while** 迴圈內容何者正確？

- (A) **k = i % j;**
i = j;
j = k;
- (B) **i = j;**
j = k;
k = i % j;
- (C) **i = j;**
j = i % k;
k = i;
- (D) **k = i;**
i = j;
j = i % k;

```
i = 76;
j = 48;
while ((i % j) != 0) {
    _____
    _____
    _____
}
printf ("%d\n", j);
```

24. 右側程式輸出為何？

- (A) bar: 6
bar: 1
bar: 8
- (B) bar: 6
foo: 1
bar: 3
- (C) bar: 1
foo: 1
bar: 8
- (D) bar: 6
foo: 1
foo: 3

```
void foo (int i) {
    if (i <= 5) {
        printf ("foo: %d\n", i);
    }
    else {
        bar(i - 10);
    }
}

void bar (int i) {
    if (i <= 10) {
        printf ("bar: %d\n", i);
    }
    else {
        foo(i - 5);
    }
}

void main() {
    foo(15106);
    bar(3091);
    foo(6693);
}
```

25. 若以 `f(22)` 呼叫右側 `f()` 函式，總共會印出多少數字？

- (A) 16
- (B) 22
- (C) 11
- (D) 15

```
void f(int n) {
    printf ("%d\n", n);
    while (n != 1) {
        if ((n%2)==1) {
            n = 3*n + 1;
        }
        else {
            n = n / 2;
        }
        printf ("%d\n", n);
    }
}
```

26. 右側程式執行過後所輸出數值為何？

- (A) 11
- (B) 13
- (C) 15
- (D) 16

```
void main () {  
    int count = 10;  
    if (count > 0) {  
        count = 11;  
    }  
    if (count > 10) {  
        count = 12;  
        if (count % 3 == 4) {  
            count = 1;  
        }  
        else {  
            count = 0;  
        }  
    }  
    else if (count > 11) {  
        count = 13;  
    }  
    else {  
        count = 14;  
    }  
    if (count) {  
        count = 15;  
    }  
    else {  
        count = 16;  
    }  
  
    printf ("%d\n", count);  
}
```

27. 右側程式片段主要功能為：輸入六個整數，檢測並印出最後一個數字是否為六個數字中最小的值。然而，這個程式是錯誤的。請問以下哪一組測試資料可以測試出程式有誤？

- (A) 11 12 13 14 15 3
(B) 11 12 13 14 25 20
(C) 23 15 18 20 11 12
(D) 18 17 19 24 15 16

```
#define TRUE 1
#define FALSE 0
int d[6], val, allBig;

...
for (int i=1; i<=5; i=i+1) {
    scanf ("%d", &d[i]);
}
scanf ("%d", &val);
allBig = TRUE;
for (int i=1; i<=5; i=i+1) {
    if (d[i] > val) {
        allBig = TRUE;
    }
    else {
        allBig = FALSE;
    }
}
if (allBig == TRUE) {
    printf ("%d is the smallest.\n", val);
}
else {
    printf ("%d is not the smallest.\n", val);
}
}
```

28. 右側為一個計算 n 階層的函式，請問該如何修改才會得到正確的結果？

```
1. int fun (int n) {
2.     int fac = 1;
3.     if (n >= 0) {
4.         fac = n * fun(n - 1);
5.     }
6.     return fac;
7. }
```

- (A) 第 2 行，改為 `int fac = n;`
(B) 第 3 行，改為 `if (n > 0) {`
(C) 第 4 行，改為 `fac = n * fun(n+1);`
(D) 第 4 行，改為 `fac = fac * fun(n-1);`

29. 右側 **f()** 函式執行後所回傳的值為何？

- (A) 1023
- (B) 1024
- (C) 2047
- (D) 2048

```
int f() {
    int p = 2;
    while (p < 2000) {
        p = 2 * p;
    }
    return p;
}
```

30. 右側 **f()** 函式 (a), (b), (c) 處需分別填入哪些數字，方能使得 **f(4)** 輸出 2468 的結果？

- (A) 1, 2, 1
- (B) 0, 1, 2
- (C) 0, 2, 1
- (D) 1, 1, 1

```
int f(int n) {
    int p = 0;
    int i = n;
    while (i >= (a) ) {
        p = 10 - (b) * i;
        printf ("%d", p);
        i = i - (c) ;
    }
}
```

31. 右側 **g(4)** 函式呼叫執行後，回傳值為何？

- (A) 6
- (B) 11
- (C) 13
- (D) 14

```
int f (int n) {
    if (n > 3) {
        return 1;
    }
    else if (n == 2) {
        return (3 + f(n+1));
    }
    else {
        return (1 + f(n+1));
    }
}

int g(int n) {
    int j = 0;
    for (int i=1; i<=n-1; i=i+1) {
        j = j + f(i);
    }
    return j;
}
```

32. 右側 **Mystery()** 函式 **else** 部分運算式應為何，才能使得 **Mystery(9)** 的回傳值為 **34**。

- (A) **x + Mystery(x-1)**
(B) **x * Mystery(x-1)**
(C) **Mystery(x-2) + Mystery(x+2)**
(D) **Mystery(x-2) + Mystery(x-1)**

```
int Mystery (int x) {
    if (x <= 1) {
        return x;
    }
    else {
        return _____ ;
    }
}
```

33. 右側 **F()** 函式執行後，輸出為何？

- (A) 1 2
(B) 1 3
(C) 3 2
(D) 3 3

```
void F( ) {
    char t, item[] = {'2', '8', '3', '1', '9'};
    int a, b, c, count = 5;
    for (a=0; a<count-1; a=a+1) {
        c = a;
        t = item[a];
        for (b=a+1; b<count; b=b+1) {
            if (item[b] < t) {
                c = b;
                t = item[b];
            }
        }
        if ((a==2) && (b==3)) {
            printf ("%c %d\n", t, c);
        }
    }
}
```

34. 右側 **switch** 敘述程式碼可以如何以 **if-else** 改寫？

- (A) **if (x==10) y = 'a';**
 if (x==20 || x==30) y = 'b';
 y = 'c';
(B) **if (x==10) y = 'a';**
 else if (x==20 || x==30) y = 'b';
 else y = 'c';
(C) **if (x==10) y = 'a';**
 if (x>=20 && x<=30) y = 'b';
 y = 'c';
(D) **if (x==10) y = 'a';**
 else if(x>=20 && x<=30) y = 'b';
 else y = 'c';

```
switch (x) {
    case 10: y = 'a'; break;
    case 20:
    case 30: y = 'b'; break;
    default: y = 'c';
}
```

35. 給定右側 G(), K() 兩函式，執行 G(3) 後所回傳的值為何？

- (A) 5
- (B) 12
- (C) 14
- (D) 15

```
int K(int a[], int n) {
    if (n >= 0)
        return (K(a, n-1) + a[n]);
    else
        return 0;
}

int G(int n){
    int a[] = {5,4,3,2,1};
    return K(a, n);
}
```

36. 右側程式碼執行後輸出結果為何？

- (A) 3
- (B) 4
- (C) 5
- (D) 6

```
int a=2, b=3;
int c=4, d=5;
int val;

val = b/a + c/b + d/b;
printf ("%d\n", val);
```

37. 右側程式碼執行後輸出結果為何？

- (A) 2468975319
- (B) 1357924689
- (C) 1234567899
- (D) 2468513799

```
int a[9] = {1, 3, 5, 7, 9, 8, 6, 4, 2};
int n=9, tmp;

for (int i=0; i<n; i=i+1) {
    tmp = a[i];
    a[i] = a[n-i-1];
    a[n-i-1] = tmp;
}

for (int i=0; i<=n/2; i=i+1)
    printf ("%d %d ", a[i], a[n-i-1]);
```

38. 右側函式以 F(7) 呼叫後回傳值為 12，則 <condition> 應為何？

- (A) $a < 3$
- (B) $a < 2$
- (C) $a < 1$
- (D) $a < 0$

```
int F(int a) {
    if ( <condition> )
        return 1;
    else
        return F(a-2) + F(a-3);
}
```

39. 若 n 為正整數，右側程式三個迴圈執行完畢後 a 值將為何？

- (A) $n(n+1)/2$
(B) $n^3/2$
(C) $n(n-1)/2$
(D) $n^2(n+1)/2$

```
int a=0, n;  
...  
for (int i=1; i<=n; i=i+1)  
    for (int j=i; j<=n; j=j+1)  
        for (int k=1; k<=n; k=k+1)  
            a = a + 1;
```

40. 下面哪組資料若依序存入陣列中，將無法直接使用二分搜尋法搜尋資料？

- (A) a, e, i, o, u
(B) 3, 1, 4, 5, 9
(C) 10000, 0, -10000
(D) 1, 10, 10, 10, 100

41. 右側是依據分數 s 評定等第的程式碼片段，正確的等第公式應為：

90~100 判為 A 等
80~89 判為 B 等
70~79 判為 C 等
60~69 判為 D 等
0~59 判為 F 等

這段程式碼在處理 0~100 的分數時，有幾個分數的等第是錯的？

- (A) 20
(B) 11
(C) 2
(D) 10

```
if (s>=90) {  
    printf ("A \n");  
}  
else if (s>=80) {  
    printf ("B \n");  
}  
else if (s>60) {  
    printf ("D \n");  
}  
else if (s>70) {  
    printf ("C \n");  
}  
else {  
    printf ("F\n");  
}
```


42. 右側主程式執行完三次G()的呼叫後,p 陣列中有幾個元素的值為 0?

- (A) 1
- (B) 2
- (C) 3
- (D) 4

```
int K (int p[], int v) {
    if (p[v]!=v) {
        p[v] = K(p, p[v]);
    }
    return p[v];
}

void G (int p[], int l, int r) {
    int a=K(p, l), b=K(p, r);
    if (a!=b) {
        p[b] = a;
    }
}

int main (void) {
    int p[5]={0, 1, 2, 3, 4};
    G(p, 0, 1);
    G(p, 2, 4);
    G(p, 0, 4);
    return 0;
}
```

43. 右側程式片段執行後,count 的值為何?

- (A) 36
- (B) 20
- (C) 12
- (D) 3

```
int maze[5][5]= {{1, 1, 1, 1, 1},
                  {1, 0, 1, 0, 1},
                  {1, 1, 0, 0, 1},
                  {1, 0, 0, 1, 1},
                  {1, 1, 1, 1, 1}};

int count=0;
for (int i=1; i<=3; i=i+1) {
    for (int j=1; j<=3; j=j+1) {
        int dir[4][2] = {{-1,0}, {0,1}, {1,0}, {0,-1}};
        for (int d=0; d<4; d=d+1) {
            if (maze[i+dir[d][0]][j+dir[d][1]]==1) {
                count = count + 1;
            }
        }
    }
}
```

44. 假設 x, y, z 為布林(boolean)變數，且 $x=TRUE$, $y=TRUE$, $z=FALSE$ 。請問下面各布林運算式的真假值依序為何？(TRUE 表真，FALSE 表假)

- $!(y \ || \ z) \ || \ x$
- $!y \ || \ (z \ || \ !x)$
- $z \ || \ (x \ \&\& \ (y \ || \ z))$
- $(x \ || \ x) \ \&\& \ z$

- (A) TRUE FALSE TRUE FALSE
(B) FALSE FALSE TRUE FALSE
(C) FALSE TRUE TRUE FALSE
(D) TRUE TRUE FALSE TRUE

45. 右側程式片段執行過程的輸出為何？

```
int i, sum, arr[10];
for (int i=0; i<10; i=i+1)
    arr[i] = i;
sum = 0;
for (int i=1; i<9; i=i+1)
    sum = sum - arr[i-1] + arr[i] + arr[i+1];
printf ("%d", sum);
```

- (A) 44
(B) 52
(C) 54
(D) 63

46. 右側程式片段中執行後若要印出下列圖案，(a) 的條件判斷式該如何設定？

```
*****
****
**
```

```
for (int i=0; i<=3; i=i+1) {
    for (int j=0; j<i; j=j+1)
        printf(" ");
    for (int k=6-2*i; _____ (a) _____ ; k=k-1)
        printf("*");
    printf("\n");
}
```

- (A) $k > 2$
(B) $k > 1$
(C) $k > 0$
(D) $k > -1$

47. 給定右側 G () 函式，執行 G(1) 後所輸出的值為何？

- (A) 1 2 3
- (B) 1 2 3 2 1
- (C) 1 2 3 3 2 1
- (D) 以上皆非

```
void G (int a){
    printf ("%d ", a);
    if (a>=3)
        return;
    else
        G(a+1);
    printf ("%d ", a);
}
```

48. 下列程式碼是自動計算找零程式的一部分，程式碼中三個主要變數分別為 Total (購買總額)，Paid (實際支付金額)，Change (找零金額)。但是此程式片段有冗餘的程式碼，請找出冗餘程式碼的區塊。

- (A) 冗餘程式碼在 A 區
- (B) 冗餘程式碼在 B 區
- (C) 冗餘程式碼在 C 區
- (D) 冗餘程式碼在 D 區

```
int Total, Paid, Change;
...
Change = Paid - Total;
printf ("500 : %d pieces\n", (Change-Change%500)/500);
Change = Change % 500;

printf ("100 : %d coins\n", (Change-Change%100)/100);
Change = Change % 100;

// A 區
printf ("50 : %d coins\n", (Change-Change%50)/50);
Change = Change % 50;

// B 區
printf ("10 : %d coins\n", (Change-Change%10)/10);
Change = Change % 10;

// C 區
printf ("5 : %d coins\n", (Change-Change%5)/5);
Change = Change % 5;

// D 區
printf ("1 : %d coins\n", (Change-Change%1)/1);
Change = Change % 1;
```

49. 右側程式執行後輸出為何？

- (A) 0
- (B) 10
- (C) 25
- (D) 50

```
int G (int B) {
    B = B * B;
    return B;
}

int main () {
    int A=0, m=5;

    A = G(m);
    if (m < 10)
        A = G(m) + A;
    else
        A = G(m);

    printf ("%d \n", A);
    return 0;
}
```

50. 右側 G() 應為一支遞迴函式，已知當 a 固定為 2，不同的變數 x 值會有不同的回傳值如下表所示。請找出 G() 函式中 (a) 處的計算式該為何？

| a 值 | x 值 | G(a, x) 回傳值 |
|-----|-----|-------------|
| 2 | 0 | 1 |
| 2 | 1 | 6 |
| 2 | 2 | 36 |
| 2 | 3 | 216 |
| 2 | 4 | 1296 |
| 2 | 5 | 7776 |

```
int G (int a, int x) {
    if (x == 0)
        return 1;
    else
        return ____ (a) ____ ;
}
```

- (A) $((2*a)+2) * G(a, x - 1)$
- (B) $(a+5) * G(a-1, x - 1)$
- (C) $((3*a)-1) * G(a, x - 1)$
- (D) $(a+6) * G(a, x - 1)$

51. 請問右側程式，執行完後輸出為何？

- (A) 2417851639229258349412352 7
- (B) 68921 43
- (C) 65537 65539
- (D) 134217728 6

```
int i=2, x=3;
int N=65536;

while (i <= N) {
    i = i * i * i;
    x = x + 1;
}
printf ("%d %d \n", i, x);
```

52. 右側 G () 為遞迴函式，G (3, 7) 執行後回傳值為何？

- (A) 128
- (B) 2187
- (C) 6561
- (D) 1024

```
int G (int a, int x) {
    if (x == 0)
        return 1;
    else
        return (a * G(a, x - 1));
}
```

53. 右側函式若以 search (1, 10, 3) 呼叫時，search 函式總共會被執行幾次？

- (A) 2
- (B) 3
- (C) 4
- (D) 5

```
void search (int x, int y, int z) {
    if (x < y) {
        t = ceiling ((x + y)/2);
        if (z >= t)
            search(t, y, z);
        else
            search(x, t - 1, z);
    }
}
```

註：ceiling() 為無條件進位至整數位。例如 ceiling(3.1)=4, ceiling(3.9)=4。

54. 給定一個 1×8 的陣列 **A**，**A** = {0, 2, 4, 6, 8, 10, 12, 14}。右側函式 **Search(x)** 真正目的是找到 **A** 之中大於 **x** 的最小值。然而，這個函式有誤。請問下列哪個函式呼叫可測出函式有誤？

- (A) **Search(-1)**
- (B) **Search(0)**
- (C) **Search(10)**
- (D) **Search(16)**

```
int A[8]={0, 2, 4, 6, 8, 10, 12, 14};

int Search (int x) {
    int high = 7;
    int low = 0;
    while (high > low) {
        int mid = (high + low)/2;
        if (A[mid] <= x) {
            low = mid + 1;
        }
        else {
            high = mid;
        }
    }
    return A[high];
}
```

55. 給定函式 **A1()**、**A2()** 與 **F()** 如下，以下敘述何者有誤？

```
void A1 (int n) {
    F(n/5);
    F(4*n/5);
}
```

```
void A2 (int n) {
    F(2*n/5);
    F(3*n/5);
}
```

```
void F (int x) {
    int i;
    for (i=0; i<x; i=i+1)
        printf("*");
    if (x>1) {
        F(x/2);
        F(x/2);
    }
}
```

- (A) **A1(5)** 印的 '*' 個數比 **A2(5)** 多
- (B) **A1(13)** 印的 '*' 個數比 **A2(13)** 多
- (C) **A2(14)** 印的 '*' 個數比 **A1(14)** 多
- (D) **A2(15)** 印的 '*' 個數比 **A1(15)** 多

56. 右側 **F()** 函式回傳運算式該如何寫，才會使得 **F(14)** 的回傳值為 40?

- (A) $n * F(n-1)$
- (B) $n + F(n-3)$
- (C) $n - F(n-2)$
- (D) $F(3n+1)$

```
int F (int n) {
    if (n < 4)
        return n;
    else
        return _____?_____ ;
}
```

57. 右側函式兩個回傳式分別該如何撰寫，才能正確計算並回傳兩參數 **a, b** 之最大公因數 (Greatest Common Divisor) ?

- (A) **a, GCD(b, r)**
- (B) **b, GCD(b, r)**
- (C) **a, GCD(a, r)**
- (D) **b, GCD(a, r)**

```
int GCD (int a, int b) {
    int r;

    r = a % b;
    if (r == 0)
        return _____;
    return _____;
}
```

58. 若 **A** 是一個可儲存 **n** 筆整數的陣列，且資料儲存於 **A[0]~A[n-1]**。經過右側程式碼運算後，以下何者敘述不一定正確?

- (A) **p** 是 **A** 陣列資料中的最大值
- (B) **q** 是 **A** 陣列資料中的最小值
- (C) **q < p**
- (D) **A[0] <= p**

```
int A[n]={ ... };
int p = q = A[0];
for (int i=1; i<n; i=i+1) {
    if (A[i] > p)
        p = A[i];
    if (A[i] < q)
        q = A[i];
}
```

59. 若 $A[i][j]$ 是一個 $M \times N$ 的整數陣列，右側程式片段用以計算 A 陣列每一列的總和，以下敘述何者正確？

```
void main () {  
    int rowsum = 0;  
    for (int i=0; i<M; i=i+1) {  
        for (int j=0; j<N; j=j+1) {  
            rowsum = rowsum + A[i][j];  
        }  
        printf("The sum of row %d is %d.\n", i, rowsum);  
    }  
}
```

- (A) 第一列總和是正確，但其他列總和不一定正確
(B) 程式片段在執行時會產生錯誤 (run-time error)
(C) 程式片段中有語法上的錯誤
(D) 程式片段會完成執行並正確印出每一列的總和
60. 若以 $B(5, 2)$ 呼叫右側 $B()$ 函式，總共會印出幾次 “base case”？

```
int B (int n, int k) {  
    if (k == 0 || k == n){  
        printf ("base case\n");  
        return 1;  
    }  
    return B(n-1,k-1) + B(n-1,k);  
}
```

- (A) 1
(B) 5
(C) 10
(D) 19

61. 給定右側程式，其中 `s` 有被宣告為全域變數，請問程式執行後輸出為何？

- (A) 1, 6, 7, 7, 8, 8, 9
- (B) 1, 6, 7, 7, 8, 1, 9
- (C) 1, 6, 7, 8, 9, 9, 9
- (D) 1, 6, 7, 7, 8, 9, 9

```
int s = 1; // 全域變數

void add (int a) {
    int s = 6;
    for( ; a>=0; a=a-1) {
        printf("%d,", s);
        s++;
        printf("%d,", s);
    }
}

int main () {
    printf("%d,", s);
    add(s);
    printf("%d,", s);
    s = 9;
    printf("%d", s);
    return 0;
}
```

62. 右側 `F()` 函式執行時，若輸入依序為整數 0, 1, 2, 3, 4, 5, 6, 7, 8, 9，請問 `X[]` 陣列的元素值依順序為何？

- (A) 0, 1, 2, 3, 4, 5, 6, 7, 8, 9
- (B) 2, 0, 2, 0, 2, 0, 2, 0, 2, 0
- (C) 9, 0, 1, 2, 3, 4, 5, 6, 7, 8
- (D) 8, 9, 0, 1, 2, 3, 4, 5, 6, 7

```
void F () {
    int X[10] = {0};
    for (int i=0; i<10; i=i+1) {
        scanf("%d", &X[(i+2)%10]);
    }
}
```

63. 若以 **G(100)** 呼叫右側函式後，**n** 的值為何？

- (A) 25
- (B) 75
- (C) 150
- (D) 250

```
int n = 0;

void K (int b) {
    n = n + 1;
    if (b % 4)
        K(b+1);
}

void G (int m) {
    for (int i=0; i<m; i=i+1) {
        K(i);
    }
}
```

64. 若 **A[1]**、**A[2]**，和 **A[3]** 分別為陣列 **A[]** 的三個元素(element)，下列那個程式片段可以將 **A[1]** 和 **A[2]** 的內容交換？

- (A) **A[1] = A[2]; A[2] = A[1];**
- (B) **A[3] = A[1]; A[1] = A[2]; A[2] = A[3];**
- (C) **A[2] = A[1]; A[3] = A[2]; A[1] = A[3];**
- (D) 以上皆可

65. 若函式 **rand()** 的回傳值為一介於 0 和 10000 之間的亂數，下列那個運算式可產生介於 100 和 1000 之間的任意數(包含 100 和 1000)？

- (A) **rand() % 900 + 100**
- (B) **rand() % 1000 + 1**
- (C) **rand() % 899 + 101**
- (D) **rand() % 901 + 100**

66. 右側程式片段無法正確列印 20 次的 "Hi!"，請問下列哪一個修正方式仍無法正確列印 20 次的 "Hi!"？

```
for (int i=0; i<=100; i=i+5) {
    printf ("%s\n", "Hi!");
}
```

- (A) 需要將 **i<=100** 和 **i=i+5** 分別修正為 **i<20** 和 **i=i+1**
- (B) 需要將 **i=0** 修正為 **i=5**
- (C) 需要將 **i<=100** 修正為 **i<100;**
- (D) 需要將 **i=0** 和 **i<=100** 分別修正為 **i=5** 和 **i<100**

67. 若以 **F(15)** 呼叫右側 **F()** 函式，總共會印出幾行數字？

- (A) 16 行
- (B) 22 行
- (C) 11 行
- (D) 15 行

```
void F (int n) {  
    printf ("%d\n" , n);  
    if ((n%2 == 1) && (n > 1)){  
        return F(5*n+1);  
    }  
    else {  
        if (n%2 == 0)  
            return F(n/2);  
    }  
}
```

68. 給定右側函式 **F()**，執行 **F()** 時哪一行程式碼可能永遠不會被執行到？

- (A) `a = a + 5;`
- (B) `a = a + 2;`
- (C) `a = 5;`
- (D) 每一行都執行得到

```
void F (int a) {  
    while (a < 10)  
        a = a + 5;  
    if (a < 12)  
        a = a + 2;  
    if (a <= 11)  
        a = 5;  
}
```

69. 給定右側函式 **F()**，已知 **F(7)** 回傳值為 17，且 **F(8)** 回傳值為 25，請問 **if** 的條件判斷式應為何？

- (A) `a % 2 != 1`
- (B) `a * 2 > 16`
- (C) `a + 3 < 12`
- (D) `a * a < 50`

```
int F (int a) {  
    if ( _____?_____ )  
        return a * 2 + 3;  
    else  
        return a * 3 + 1;  
}
```

70. 給定右側函式 $F()$ ， $F()$ 執行完所回傳的 x 值為何？

- (A) $n(n+1)\sqrt{\lfloor \log_2 n \rfloor}$
- (B) $n^2(n+1)/2$
- (C) $n(n+1)\lfloor \log_2 n + 1 \rfloor / 2$
- (D) $n(n+1)/2$

```
int F (int n) {
    int x = 0;
    for (int i=1; i<=n; i=i+1)
        for (int j=i; j<=n; j=j+1)
            for (int k=1; k<=n; k=k*2)
                x = x + 1;
    return x;
}
```

71. 右側程式執行完畢後所輸出值為何？

- (A) 12
- (B) 24
- (C) 16
- (D) 20

```
int main() {
    int x = 0, n = 5;
    for (int i=1; i<=n; i=i+1)
        for (int j=1; j<=n; j=j+1) {
            if ((i+j)==2)
                x = x + 2;
            if ((i+j)==3)
                x = x + 3;
            if ((i+j)==4)
                x = x + 4;
        }
    printf ("%d\n", x);
    return 0;
}
```

72. 右側程式擬找出陣列 $A[]$ 中的最大值和最小值。不過，這段程式碼有誤，請問 $A[]$ 初始值如何設定就可以測出程式有誤？

- (A) {90, 80, 100}
- (B) {80, 90, 100}
- (C) {100, 90, 80}
- (D) {90, 100, 80}

```
int main () {
    int M = -1, N = 101, s = 3;
    int A[] = _____?_____;

    for (int i=0; i<s; i=i+1) {
        if (A[i]>M) {
            M = A[i];
        }
        else if (A[i]<N) {
            N = A[i];
        }
    }
    printf("M = %d, N = %d\n", M, N);
    return 0;
}
```

73. 小藍寫了一段複雜的程式碼想考考你是否了解函式的執行流程。請回答程式最後輸出的數值為何？

- (A) 70
- (B) 80
- (C) 100
- (D) 190

```
int g1 = 30, g2 = 20;

int f1(int v) {
    int g1 = 10;
    return g1+v;
}

int f2(int v) {
    int c = g2;
    v = v+c+g1;
    g1 = 10;
    c = 40;
    return v;
}

int main() {
    g2 = 0;
    g2 = f1(g2);
    printf("%d", f2(f2(g2)));
    return 0;
}
```

74. 若以 **F(5,2)** 呼叫右側 **F()** 函式，執行完畢後回傳值為何？

- (A) 1
- (B) 3
- (C) 5
- (D) 8

```
int F (int x,int y) {
    if (x<1)
        return 1;
    else
        return F(x-y,y)+F(x-2*y,y);
}
```

75. 若要邏輯判斷式 **!(X₁ || X₂)** 計算結果為真(True)，則 **X₁** 與 **X₂** 的值分別應為何？

- (A) **X₁** 為 False, **X₂** 為 False
- (B) **X₁** 為 True, **X₂** 為 True
- (C) **X₁** 為 True, **X₂** 為 False
- (D) **X₁** 為 False, **X₂** 為 True