

1. 右側 F() 函式執行後，輸出為何？

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

此程式做什麼事？

當 a == 2 && b == 3 時

```
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);
            }
        }
    }
}
```

a = 2
c = 2
t = item[2] = '3'
b = 3
if ('1' < '3')
c = b = 3
t = item[3] = '1'

2. 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';
}
```

3. 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

$G(3) = K(a, 3)$
 $= K(a, 2) + a[3]$
 $= K(a, 1) + a[2] + a[3]$
 $= K(a, 0) + a[1] + a[2] + a[3]$
 $= K(a, -1) + a[0] + a[1] + a[2] + a[3]$
 $= 0 + 5 + 4 + 3 + 2$

```
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);
}
```

4. 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;
float f = 3;
int val = f/a;
val = ?

整數除法
 $b/a = 3/2 = 1$
 $c/b = 4/3 = 1$
 $d/b = 5/3 = 1$

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

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

5. 右側程式碼執行後輸出結果為何？

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

i = 0, 輸出 a[0], a[8],
 i = 1, 輸出 a[1], a[7],
 i = 2, 輸出 a[2], a[6],
 i = 3, 輸出 a[3], a[5],
 i = 4, 輸出 a[4], a[4] 的陣列元素內容

```
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]);
```

a[0] 與 a[8] 對調
 a[1] 與 a[7] 對調
 ...
 a[7] 與 a[1] 對調
 a[8] 與 a[0] 對調

6. 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) $a < 0$

$F(7)$
 $= F(5) + F(4)$
 $= F(3) + F(2) + F(2) + F(1)$
 $= F(1) + F(0) + F(0) + F(-1) + F(0) + F(-1) + F(-1) + F(-2)$

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

if判斷式不成立時，
F函式會呼叫自己兩次

7. 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) $n^2(n+1)/2$

外面兩個迴圈執行次數如下：

$i = 1, n$ 次
 $i = 2, n - 1$ 次
 $i = 3, n - 2$ 次
...

$i = n - 1, 2$ 次
 $i = n, 1, 1$ 次

總次數為： $n + (n-1) + (n-2) + \dots + 3 + 2 + 1$
 $= n * (n+1) / 2$

最內層迴圈執行次數為 n 次
所以此三個迴圈總次數為 $= n * n * (n+1) / 2$

```
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;
```

8. 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

二分搜尋只能在已
排序好的資料進行

9. 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

79~70 會判斷為 D
60 會判斷為 F

```
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");
}
```

10. 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

此程式做什麼事？

G(p, l, r) 函數會將 p[r] 的內容改成 p[l]

p = {0, 1, 2, 3, 4}, G(p, 1, r)

執行 G(p, 0, 1):

a = K(p, l) = K(p, 0) = p[0] = 0

b = K(p, r) = K(p, 1) = p[1] = 1

if(a != b) ==> if(0 != 1) ==> p[1] = 0
==> p 陣列內容 {0, 0, 2, 3, 4}

G(p, 0, 1), p = {0, 0, 2, 3, 4}

G(p, 2, 4), p = {0, 0, 2, 3, 2}

G(p, 0, 4), p = {0, 0, 0, 3, 2}

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

陣列內容與陣列索引不一樣

```
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;
}
```

maze[i][j] ==> maze[1][1] ==> maze[3][3]

11. 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;
            }
        }
    }
}
```

4+1+3 +
1+2+2 +
3 + 2 + 2
= 20

此程式做什麼事？

12. 右側程式片段執行過程中的輸出為何？

- (A) 5 10 15 20
- (B) 5 11 17 23
- (C) 6 12 18 24
- (D) 6 11 17 22

for 迴圈執行的順序
for(A; B; D) { C; }

```
int a = 5;  
...  
for (int i=0; i<20; i=i+1){  
    i = i + a;  
    printf ("%d ", i);  
}
```

13. 若宣告一個字元陣列 `char str[20] = "Hello world!"`; 該陣列 `str[12]` 值為何？

字串會有結束字元 '\0'

- (A) 未宣告
- (B) \0
- (C) !
- (D) \n

14. 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

15. 右側程式片段執行過程的輸出為何？

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

```
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);
```

$arr[i+1] - arr[i-1] = 2 \Rightarrow 2 * 8 = 16$
 $arr[i] \Rightarrow 1 + 2 + \dots + 7 + 8 = 36$

16. 右列程式片段中，假設 a, a_ptr 和 a_ptrptr 這三個變數都被正確宣告，且呼叫 G() 函式時的參數為 a_ptr 及 a_ptrptr。G() 函式的兩個參數型態該如何宣告？

指標語法

- (A) (a) *int, (b) *int
(B) (a) *int, (b) **int
(C) (a) int*, (b) int*
(D) (a) int*, (b) int**

```
void G ( __ (a) __ a_ptr, __ (b) __ a_ptrptr) {
    ...
}

void main () {
    int a = 1;
    // 加入 a_ptr, a_ptrptr 變數的宣告
    ...
    a_ptr = &a;
    a_ptrptr = &a_ptr;
    G (a_ptr, a_ptrptr);
}
```

17. 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) $k > -1$

i = 0 時，要印出*****
 i = 1 時，要印出****
 i = 2 時，要印出**
 i = 3 時，不輸出任何東西。

18. 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);  
}
```

G(1) ==> 輸出1, 執行G(2), a == 1
G(2) ==> 輸出2, 執行G(3), a == 2
G(3) ==> 3, return

19. 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;
```

20. 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;
}
```

21. 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？

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

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

6 * G(a, x - 1)

- (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)

22. 如果 X_n 代表 X 這個數字是 n 進位，請問 $D02A_{16} + 5487_{10}$ 等於多少？

(A) $1100\ 0101\ 1001\ 1001_2$

$$5487_{10} = 156F_{16}$$

(B) 162631_8

(C) 58787_{16}

(D) $F599_{16}$

23. 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

$$65536 = 2^{16}$$

$$x = 3, i = 2 * 2 * 2 = 8$$

$$x = 4, i = 8 * 8 * 8 = 512$$

$$x = 5, i = 512 * 512 * 512$$

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

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

24. 右側 $G()$ 為遞迴函式， $G(3, 7)$ 執行後回傳值為何？

(A) 128

(B) 2187 $3^7=2187$

(C) 6561

(D) 1024

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

25. 右側函式若以 $\text{search}(1, 10, 3)$ 呼叫時， search 函式總共會被執行幾次？

(A) 2

(B) 3

(C) 4

(D) 5

$\text{search}(1, 10, 3)$
 $\Rightarrow \text{search}(1, 5, 3)$
 $\Rightarrow \text{search}(3, 5, 3)$
 $\Rightarrow \text{search}(4, 5, 3)$
 $\Rightarrow \text{search}(5, 5, 3)$

```
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 $\text{ceiling}(3.1)=4$ ， $\text{ceiling}(3.9)=4$ 。