計算機程式設計 113 學年度第 2 學期小考 3 試題

系級：__________ 學號：__________ 姓名：__________

All programs have "#include <stdio.h>".

1. What compiler warnings will be generated by the following code?

__ aa=a 從不相容的指標類型 ‘char *’ 賦值給 ‘char **’ __

___(資料型別錯誤, 編譯器警告)____________________

```
void com(char *a, char **aa) {
    aa = a;
    printf("%s\n", aa);
}
int main() {
    char *a = "hello";
    char *aa;
    com(a, aa);
    return 0;
}
```

2. Fill in the space to complete the code.

(1) ___static____ (2) ____*s2_______ (3)_____ NULL_____

Output

```
00
22
5A
```

```
#include <string.h>
char *myStrtok(char *s1, const char *s2) {
    ___(1)___ char *p;
    if (s1 != NULL) p = s1;
    if (p == NULL || *p == '\0') return NULL;
    char *start = p;
    while (*p != '\0' && *p != *s2) p++;
    if (*p == ___(2)___) {
        *p = '\0';
        p++;
    } else
        p = NULL;
    return start;
}
int main() {
    char str[] = "00:22:5A";
    char *delim = ":";
    char *pch;
    pch = myStrtok(str, delim);
    while (pch != ___(3)___) {
        printf("%s\n", pch);
        pch = myStrtok(NULL, delim);
    }
    return 0;
}
```

3. Output after executing the following code.

(1) Line1: ___1, 2_________ (2) Line2: ____2,2,3,4,______

(3) Line3: ___2, 2_________ (3) Line4: ____2,2,3,4,______

```
void print(int a[], int size) {
    int i = 0;
    for (i = 0; i < size; i++) {
        printf("%d,", a[i]);
    }
    printf("\n");
}
void test1() {
    int a[] = {1, 2, 3, 4}, b[] = {0, 0, 0};
    int *p;
    p = a;
    b[0] = (*(p++))++;
    printf("%d, %d\n", b[0], *p);
    print(a, 4);
}
void test2() {
    int a[] = {1, 2, 3, 4}, b[] = {0, 0, 0};
    int *p;
    p = a;
    b[0] = ++(*(p++));
    printf("%d, %d\n", b[0], *p);
    print(a, 4);
}
int main() {
    test1();
    test2();
    return 0;
}
```

4. Output after executing the following code.

(1) ____A B______ (2) _____C B_______ (3)____B B_______

```
void swap1(char *str1, char *str2) {
    char *temp = str1;
    str1 = str2;
    str2 = temp;
}
void swap2(char **str1, char **str2) {
    char *temp = *str1;
    *str1 = *str2;
    *str2 = temp;
}
void swap3(char **str1, char *str2) {
    char *temp = *str1;
    *str1 = str2;
    str2 = temp;
}
int main() {
    char *str1 = "A", *str2 = "B", *str3 = "C";
    swap1(str1, str2);
    printf("%s %s\n", str1, str2);  //(1)
    swap2(&str2, &str3);
    printf("%s %s\n", str2, str3);  //(2)
    swap3(&str1, str3);
    printf("%s %s\n", str1, str3);  //(3)
    return 0;
}
```

5. Output after executing the following code.

(1) ____Taiwn____ (2) _____Taipwan___ (3)____ Tech____

```
#include <string.h>
int main() {
    char str1[] = "Taipei Tech";
    char str2[] = "Taipei Tech";
    char *str3 = "Taipei Tech";
    str1[3] = 'w';
    str1[4] = 'n';
    str1[5] = '\0';
    char *str4 = str2;
    strcpy(&str4[4], "wan");
    printf("%s\n", str1);  //(1)
    printf("%s\n", str4);  //(2)
    printf("%s\n", str3+7); //(3)
    return 0;
}
```

6. Output after executing the following code. Assume the following memory addresses: S[0]=0x7ffe0f3ec890

(1) Line1: s[1]= 0x7ffe0f3ec ___89f___________

(2) Line2: s[1][0]= 0x7ffe0f3ec ____89f __________

(3) Line3: s[2][0]= 0x7ffe0f3ec ____8ae__________

```
int main() {
    char S[3][15] = {"Carnation", "Rose", "Lily"};
    int i;
    for (i = 0; i < 3; i++) {
        printf("S[%d]=%p\n", i, S[i]);
        printf("S[%d][0]=%p\n", i, &S[i][0]);
    }
    return 0;
}
```

7. Fill in the space to complete the code.

(1) __<____ (2) __str1+1 or ++str1__ (3)__ str2+1 or ++str2__

Output

```
1
```

```
int myStrcmp(char *str1, char *str2) {
    if (*str1 == '\0' && *str2 == '\0') {
        return 0;
    }
    if (*str1 > *str2) {
        return 1;
    } else if (*str1 ___(1)___ *str2) {
        return -1;
    }
    return myStrcmp(___(2)___, ___(3)___);
}
int main() {
    char str1[10] = "abcd";
    char str2[10] = "abcD";
    int r = myStrcmp(str1, str2);
    printf("%d\n", r);
}
```

8. Output after executing the following code.

(1) ____T_____ (2) _____O_____ (3)______BER_______

```
int main() {
    char **p;
    char *pa[3] = {"TEMP", "WORLD", "NUMBER"};
    p = pa;
    printf("%c\n", **p);                //(1)
    printf("%c\n", *(*(p + 1) + 1));    //(2)
    printf("%s\n", *(p + 2) + 3);       //(3)
}
```

9. Output after executing the following code.

(1) ________16__________ (2) ___________160_______

```
typedef struct data {
        int num;
        char c1;
        char c2[8];
} data_t;
void main() {
    data_t d1, d2[10];
    printf("%ld\n", sizeof(d1)); // (1)
    printf("%ld\n", sizeof(d2)); // (2)
}
```

10. Output after executing the following code.

(1) ___HULLO____(2) ____MAZS____ (3)_____PLXY____

```
#include <string.h>
void modifyStrings(char *p[]) {
    strcpy(*(p + 1), "WORLD");
    strcpy(*(p + 3) + 2, "XY");
    *(*(p + 2) + 2) = 'Z';
    p[0][1] = 'U';
}
int main() {
    char strings[][10] = {"HELLO", "EARTH", "MARS",
"PLUTO"};
    char *pointerArray[4] = {strings[0], strings[1], strings[2],
strings[3]};
    modifyStrings(pointerArray);
    printf("%s\n", pointerArray[0]); //(1)
    printf("%s\n", pointerArray[2]); //(2)
    printf("%s\n", pointerArray[3]); //(3)
    return 0;
}
```

11. Output after executing the following code.

(1) ____0 0 1____ (2) _____0 1 2____ (3)___0 2 3________

```
typedef union {
        int id;
        struct {
                unsigned short x : 2;
                unsigned short y : 4;
                unsigned short z : 5;
        } bits;
} number_t;
void main() {
    number_t number;
    for (number.id = 1; number.id < 12; number.id += 5) {
        printf("%d %d %d\n", number.bits.z, number.bits.y,
number.bits.x);
    }
}
```

12. Output after executing the following code.

(1) ________577________ (2) __________A___________

```
typedef enum { NUM=1, CHAR } kind_t;
typedef union {
        int a;
        char b;
} data_t;
void print(data_t d, kind_t k) {
    if (k == NUM)
        printf("%d\n", d.a);
    else
        printf("%c\n", d.b);
}
void main() {
    data_t d;
    kind_t k;
    d.a = 512;
    d.b = 'A';          // 'A' = 65
    k = NUM;
    print(d, k);        //(1)
    k = CHAR;
    print(d, k);        //(2)
}
```

13. Output after executing the following code.

(1) ______5_____________ (2) _________13____________

```
#include <string.h>
int main() {
    const char str[] = "Hello\0World\0";
    printf("%d\n", (int)strlen(str)); //(1)
    printf("%d\n", (int)sizeof(str)); //(2)
    return 0;
}
```

14. Write the data type（資料型別）of the following code.

(1) __int*_ (2) __int** or int *[6]__ (3)_ int*** or int* [4][6]__

```
int main() {
    int *x[4][6];
    // x[0][0];   (1)
    // x[0];      (2)
    // x;         (3)
    return 0;
}
```

15. Write the data type of the following code.

e.g. __char x__: x is a char

(1) __char *x[4]___: x is an array[4] of pointer to char

(2) __char *x()____: x is a function() returning pointer to char

(3) __char (*x)[3]__: x is a pointer to array[3] of char

16. Output after executing the following code.

(1) _________23__________ (2) _________16___________

```
typedef struct cow_s {
        int eye : 2;
        unsigned nose : 4;
        unsigned : 0;
        int mouth : 1;
        int : 32;
        int leg;
} cow;
typedef struct cat_s {
        unsigned char a : 3;
        unsigned char b : 3;
        unsigned char c : 5;
        unsigned char : 0;
        unsigned char : 1;
} cat;
typedef union mix_u {
        cat x;
        int y;
} mix;

int main() {
    mix value;
    value.y = 5966;
    printf("%d\n", value.x.c);
    printf("%d\n", sizeof(cow));
}
```

17. Output after executing the following code.

(1) ______34____________ (2) _________90_________

(3) ______12____________ (4) _________78__________

```
int main() {
    int a[2][3] = {12, 34, 56, 78, 90, 51};
    int (*p)[2];
    p=a;
    printf("%d\n",*(p[0]+1));    //(1)
    printf("%d\n",(*(p+1))[2]); //(2)
    printf("%d\n",(*p)[0]);      //(3)
    printf("%d\n",*(p[1]+1));    //(4)
    return 0;
}
```

18. Please describe your learning problem and how to improve it. (30 or more word will be scored)