

① Visibilité

```

1 #include <stdio.h>
2 int a;
3
4 int main(int argc, char** argv) {
5     int b;
6
7     a=0;
8     b=a;
9     printf("a: %d, b: %d\n",a,b);
10    a += 5;
11    {
12        int a;
13        printf("a: %d, b: %d\n",a,b);
14    }
15    a += 5;
16    {
17        int b;
18        b=a;
19        printf("a: %d, b: %d\n",a,b);
20        b += 5;
21        {
22            int b;
23            b=0;
24            printf("a: %d, b: %d\n",a,b);
25        }
26        printf("a: %d, b: %d\n",a,b);
27    }
28    printf("a: %d, b: %d\n",a,b);
29    return 0;
30 }

```

② Fonctions et paramètres

```

1 #include <stdio.h>
2 int a;
3
4 void affiche_ab(int b, int a) {
5     printf("a: %d, b: %d\n",a,b);
6 }
7
8 int main(int argc, char** argv) {
9     int b;
10
11    a=0;
12    b=a;
13    affiche_ab(a,b);
14    a += 5;
15    {
16        int a;
17        affiche_ab(a,b);
18    }
19    a += 5;
20    {
21        int b;
22        b=a;
23        affiche_ab(a,b);
24        b += 5;
25        {
26            int b;
27            b=0;
28            affiche_ab(a,b);
29        }
30        affiche_ab(a,b);
31    }
32    affiche_ab(a,b);
33    return 0;
34 }

```

③ Paramètres comme variables

```

1 void affiche_ab2(int a, int b) {
2     a++;
3     printf("a: %d, b: %d\n",a,b);
4 }

```

④a Passage par valeur

```

1 int max(int a, int b) {
2     return a>b ? a : b;
3 }
4 int main(int argc, char** argv) {
5     int x = 12;
6     int y = 42;
7     int z = max(x,y);
8     return 0;
9 }

```

④b Passage par valeur

```

1 void triple(int a) {
2     a = a * 3;
3 }
4 int main(int argc, char** argv) {
5     int x = 42;
6     triple(x);
7     printf("%d\n", x);
8     return 0;
9 }

```

⑤ Durée de vie

```

1 void affiche_ab3(int a, int b) {
2     int a=0;
3
4     printf("a: %d, b: %d\n",a,b);
5     a += 5;
6 }

```

⑥ Locale static

```

1 int nextInt() {
2     static int res=0;
3     res += 1;
4     return res;
5 }
6 int main(int argc, char** argv) {
7     printf("next:%d",nextInt());
8     printf("next:%d",nextInt());
9     return 0;
10 }

```

⑦ Globale static

```

1 static void affiche_ab4(int a,
2                         int b) {
3     static int a=0;
4
5     printf("a: %d, b: %d\n",a,b);
6     a += 5;
7 }

```

⑧ Fonction locale

```

1 void affiche_ab3(int a, int b) {
2     void lambda(char* fmt) {
3         printf(fmt, a, b);
4     }
5     lambda("a: %d, b: %d\n");
6 }

```

⑨ Passage par adresse

```

1 void triple(int* a) {
2     *a = *a * 3;
3 }
4 int main(int argc, char** argv) {
5     int x = 42;
6     triple(&x);
7     printf("%d\n", x);
8     return 0;
9 }

```