指针的进阶：

指针笔试题：

int main()

{

int a[4] = { 1, 2, 3, 4 };

int \*ptr1 = (int \*)(&a + 1);

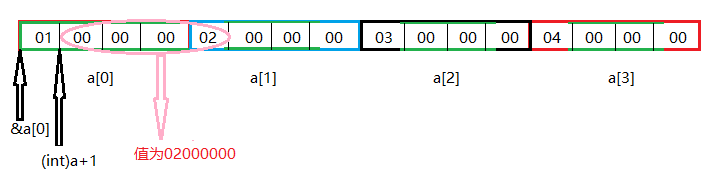
int \*ptr2 = (int \*)((int)a + 1);

printf("%x,%x", ptr1[-1], \*ptr2);

system("pause");

return 0;

}



int main()

{

int a[5][5];

int(\*p)[4];

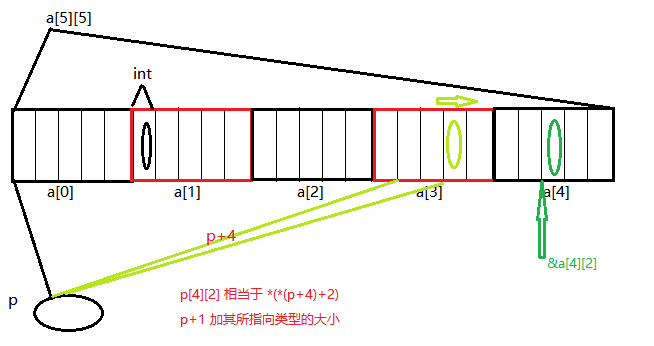
p = a;

printf("%p,%d\n", &p[4][2] - &a[4][2], &p[4][2] - &a[4][2]);

system("pause");

return 0;

}



int main()

{

char \*a[] = { "work", "at", "alibaba" };

char\*\*pa = a;

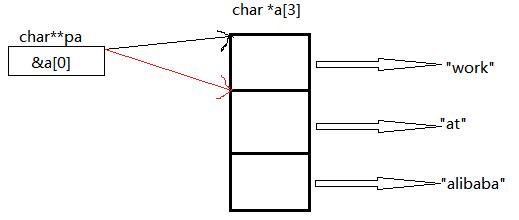
pa++;

printf("%s\n", \*pa);

system("pause");

return 0;

}



int main()

{

char \*c[] = { "ENTER", "NEW", "POINT", "FIRST" };

char\*\*cp[] = { c + 3, c + 2, c + 1, c };

char\*\*\*cpp = cp;

printf("%s\n", \*\*++cpp);

printf("%s\n", \*--\*++cpp + 3);

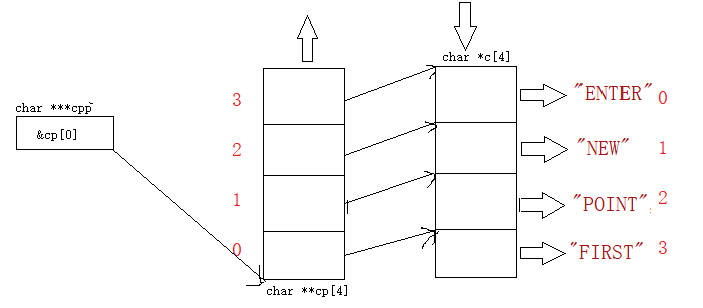
printf("%s\n", \*cpp[-2] + 3);

printf("%s\n", cpp[-1][-1] + 1);

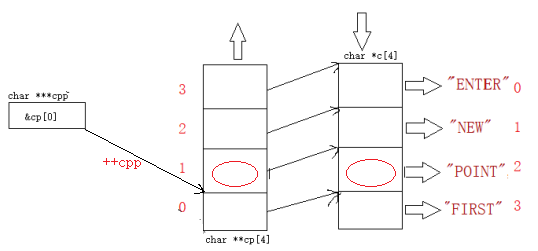
system("pause");

return 0;

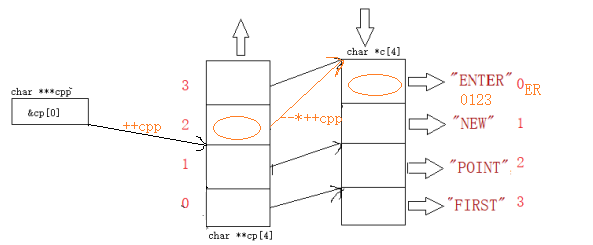
}



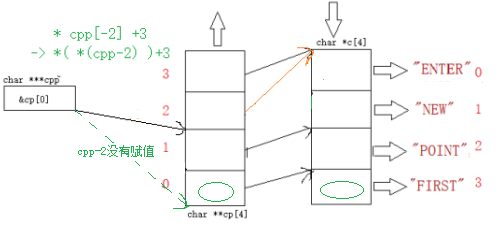
\*\*++cpp：



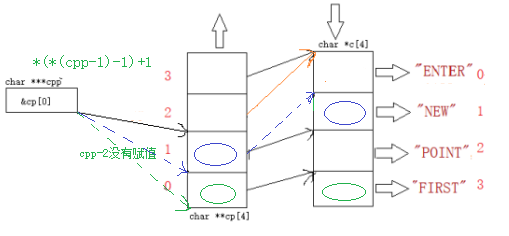
\*--\*++cpp + 3：



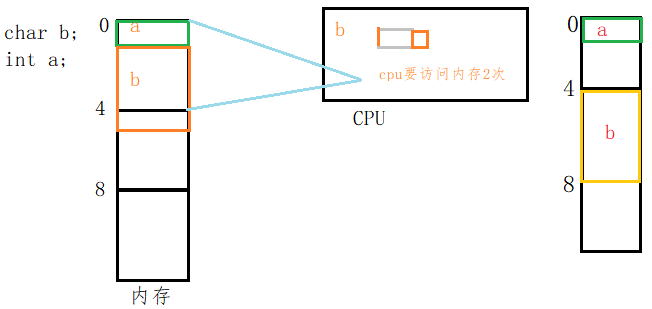
\*cpp[-2] + 3：



cpp[-1][-1] + 1：



内存对齐：



struct B //16

{

char a;

char b;

double c;

};

struct A

{ //偏移量：最后的位置

double a; //0:8

char b[3]; //8:11

char \*c; //12:16

struct B d[3]; //16:64

double e; //64:72

long long \*f[3]; //72:84

char g; //84:85->88

};最后结果：88