Write the output of the following programs (if any). If there is an error in the program, mention the error and move on.

Tip: Use python tutor (*https://pythontutor.com/cpp.html#mode=edit*) for line-by-line execution of programs for a better understanding, however, first try to solve by yourself.

Code	Output

void foo(int* arr1, const int size, int val, int* pos) {
  if(*pos == size - 1)
  *arr1 = val;
  else {
  *arr1 = val;
  ++*pos;
  foo(arr1 + 1, size, val, pos);
  }
}
 	
int main() {
  const int size = 5;
  int arr[size] = {10, 20, 33, 0, 1};
  int pos = 0;
  foo(arr, size, 10, &pos);
  for(int i = 0; i < size; ++i)
  cout<<arr[i]<<" ";
 	
  return 0;
}
	10 10 10 10 10

void make2(int *arr, int cols) {
  arr = new int[cols];
}
void make1(int **arr, int rows, int cols) {
  arr = new int*[rows];
  make2(*arr, cols);
}
void make(int ***arr, int pages, int rows, int cols) {
  arr = new int**[pages];
  make1(*arr, rows, cols);
}
 	
int main() {
 	
  int*** arr = NULL;
  make(arr, 4, 4, 4);
	Segmentation fault in the inner most loop in the main.
	Reason, memory is being allocated within the function, however, pointers are being passed by value, thus the value of pointer in the main as well in the functions didn't update after returning back.

for(int i = 0; i < 4; ++i) {
  for(int j = 0; j < 4; ++j) {
  for(int k = 0; k < 4; ++k)
  arr[i][j][k] = i + j + k;
  }
 }
 	
 return 0;
}	

int main() {
  int num[5]= {1,2,3,4,5};
  int* p;
  p = num;
  *p = 20;
  p = &num[1];
  *(++p) = 30;
  p = num + 4;
  *p = 30;
  p = num;
  *(p + 3) = 40;
  for (int i = 1; i < 5; i++)
  cout << num[i] << " ";
  return 0;
}
	2 30 40 30

int main() {
 	
 char name[5][10] = { "Pakistan", "China", "Turkiye", "Korea",
"Japan"};
 char* ptr1 = name[0];
 cout<<ptr1<<endl;
 cout<<*ptr1<<endl;
 ptr1 = name[3];
 cout<<ptr1<<endl;
 cout<<*(ptr1 + 1)<<endl;
 cout<<ptr1+2<<endl;
 ptr1 = name[1];
 cout<<*(ptr1 + 10)<<endl;
 ptr1 = name[2];
 cout<<ptr1-10<<endl;
 	
 return 0;
}
	Pakistan
	P
	Korea
	o
	rea
	T
	China

int main() {
 	
  char* alpha, beta;
  beta = new char[5];
 	
  return 0;
}
	Syntax error
	beta is of type char. It cannot allocate memory.

int main() {
 	
  int alpha = 100, beta = 200;
  int *p = &alpha, *q = β
  p = q;
  cout<<p<<endl;
  cout<<*p<<endl;
 	
  return 0;
}
	Address of beta variable
	200

int main() {
 	
  int a = 5, b = 10, c = 15;
  int *arr[] = {&a, &b, &c};
  cout << arr[1];
 	
  return 0;
}
	Address of b

int main() {
  int i, j, var = 'A';
  for (i = 3; i >= 1; i--) {
  for (j = 0; j < i; j++) {
  if(((i+var + j))%4==0)
  continue;
  cout<<char (i+var + j);
  }
  cout<<endl;
  }
  return 0;
}
	EF
	C
	B

int main() {
  char arr[20];
  int i;
  for (i = 0; i < 10; i++)
  *(arr + i) = 65 + i;
  *(arr + i) = '\0';
	ABCDEFGHIJ

cout << arr; return 0; }	

int main() {
 	
int*** arr = new int**[5];
for(int i = 0; i < 5; ++i) {
arr[i] = new int*[5];
for(int j = 0; j < 5; ++j)
arr[i][j] = new int[5] {1, 2, 3, 4, 5};
}
 	
for(int i = 0; i < 5; ++i) {
for(int j = 0; j < 5; ++j) {
for(int k = 0; k < 5; ++k)
cout<<arr[i][j][k]<<" ";
cout<<endl;
}
}
 	
delete arr;
arr = NULL;
 	
return 0;
}
	Logical Error
	Dynamic Memory deletion done incorrectly resulting in memory leak.

int main() {
 	
int a[2][4] = {3, 6, 9, 12, 15, 18, 21, 24};
cout << *(a[1] + 2) << * (*(a + 1) + 2);
 	
return 0;
}
	2121

int main() {
 	
int* scores;
scores = new int[4]{45, 65, 77, 67};
 	
if(scores)
cout<<++*scores++<<" "<<++*scores++<<" "<<++*scores++<<" "<<++*scores++;
	46 66 78 68
	
	Segmentation Fault
	Deleting unallocated memory

```
  if(scores)
    delete scores;
  scores = NULL;

  return 0;
}
```