

```
1) char outstring[5] = "Γαα";
   int index = 0;
   while (outstring[index] != '\0')
   { outstring[index] = 'A';
     index++; }
```

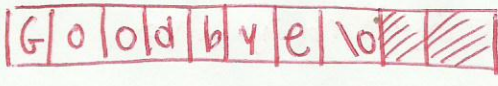
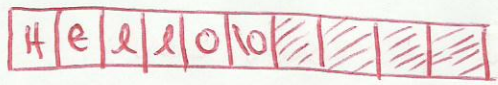
outstring[0] = 'Γ'
outstring[1] = 'α'
outstring[2] = 'α'
outstring[3] = '\0'
outstring[4] = '\0'

outstring[0] ≠ '\0' outstring[0] = 'A' 0++ = 1	outstring[1] ≠ '\0' outstring[1] = 'A' 1++ = 2	outstring[2] ≠ '\0' outstring[2] = 'A' 2++ = 3
outstring[3] ≠ '\0' outstring[3] = 'A' 3++ = 4	outstring[4] ≠ '\0'	

Αρα, outstring[5] = { 'A', 'A', 'A', 'A' }.

```
2) char astring[10] = "Hello";
   char anotherstring[10] = "Goodbye";
   if (strcmp(astring, anotherstring))
       cout << "strings NOT same";
   else
       cout << "strings are same";
```

⊕ arrival pointer over (astring == anotherstring)



```
3) char mystring[10] = "d0bed0";
   cout << strlen(mystring);
```

Επισημαίνεται το πλήθος των χαρακτήρων δίχως το '\0' δηλ το αποτέλεσμα που θα εμφανιστεί είναι : 6.

4) Ποια είναι ισοδύναμα μεταξύ τους;

- i) `char stringVar[10] = "Γεια";`
- ii) `char stringVar[10] = { 'r', 'ε', 'ι', 'α', 'ιο' }`
- iii) `char stringVar[10] = { 'r', 'ε', 'ι', 'α' }`
- iv) `char stringVar[5] = "Γεια";`
- v) `char stringVar[] = "Γεια";`

Το (i), (ii) ισοδύναμα.

Το (iii), (iv) και (v) ισοδύναμα.

5) Υπάρχει λάθος;

```
char stringVar[] = "Γεια";  
strcat(stringVar, "_και_αριο.");  
cout << stringVar;
```

Υπάρχει λάθος και αυτό είναι ότι ο πίνακας `stringVar` έχει 5 θέσεις και τον σπενδώνει με πίνακα

που έχει 11 θέσεις άρα η `stringVar[] = "Γεια"` δεν είναι αποτελεσματική

μετάλη για να δέχεται τον "_και_αριο."

```
6) char song[10] = "I did it";  
char fsong[90];  
strcpy(fsong, song);  
strcat(fsong, "my way!");  
cout << fsong << endl;
```

θα εμφανίσει:

I did it my way!

Διότι έγινε με τον `strcpy` εναλλαγή μερών των πινάκων άρα η `fsong` είναι αποτελεσματική για να δέχεται τον "my way"