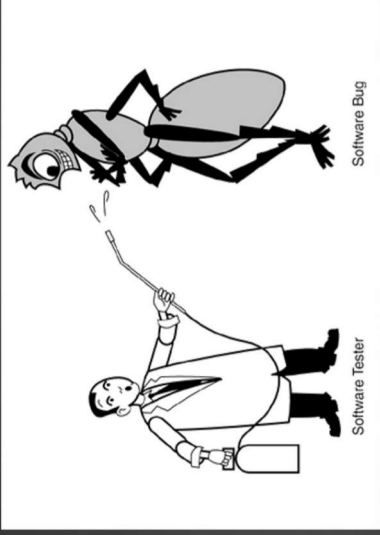


Αποσφαλμάτωση - Debugging

9



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

11

```
if (0 < a < 5) c;  
if (a == b) c;  
  
int A[10], i;  
for (i = 1; i <= 10; i++)  
    A[i] = 0;
```



Μιχάλης Ξένος, 2022

Εισαγωγικά

10

- ▶ Απόρροια επιτυχημένου ελέγχου (ελέγχου που εντόπισε ελάττωμα).
- ▶ Έλεγχος → Συστηματική διαδικασία. Γίνεται μεθοδικά.
- ▶ Αποσφαλμάτωση → Νοητική διαδικασία. Πιρισσότερο τέχνη παρά τεχνική!
- ▶ Κάποιοι είναι καλοί εκθαλαστές σφαλμάτων και άλλοι όχι.
- ▶ Η εμπειρία προγραμματισμού συμβάλει θετικά.
- ▶ Διαδικασία που μπορεί να δημιουργήσει προβλήματα στο χρονοδιάγραμμα ενός έργου.
- ▶ "Everyone knows that debugging is twice as hard as programming; so, if you are as clever as you can be when you write the program, how will you ever debug it?" - Kernighan



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

12

```
if (0 < a < 5) c;  
if (a == b) c;  
  
int A[10], i;  
for (i = 1; i <= 10; i++)  
    A[i] = 0;
```



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
if( 0 < a < 5) c;
```

```
if( a == b) c;
```

```
int A[10], i;  
for( i=1; i<=10; i++)  
    A[i]=0;
```

Αυτό μεταφράζεται ως $(a = (lb))$, και είναι καταχώρηση αντί για $(a != b)$ ή $(a == !b)$

13



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
int x = 5;  
while( x > 0 );  
x--;
```

```
int x = -5;  
while( x < 0 )  
    x--;
```

```
for( i=1; i<=10; i++)  
    printf( "%d ", i );
```



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
if( 0 < a < 5) c;
```

```
if( a == b) c;
```

```
int A[10], i;  
for( i=1; i<=10; i++)  
    A[i]=0;
```

Προσοχή στα όρια των πινάκων!
Δεν γίνεται σωστή αρχικοποίηση

14



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
int x = 5;  
while( x > 0 );  
x--;
```

Ατέμνος βρόγχος.
Αυτό το ερωτηματικό είναι το λάθος!

```
int x = -5;  
while( x < 0 )  
    x--;
```

```
for( i=1; i<=10; i++)  
    printf( "%d ", i );
```



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
int x = 5;
while( x > 0 );
x--;
```

Ατέρμονος βρόγχος; Και βέβαια όχι!

```
int x = -5;
while( x < 0 )
x--;
```

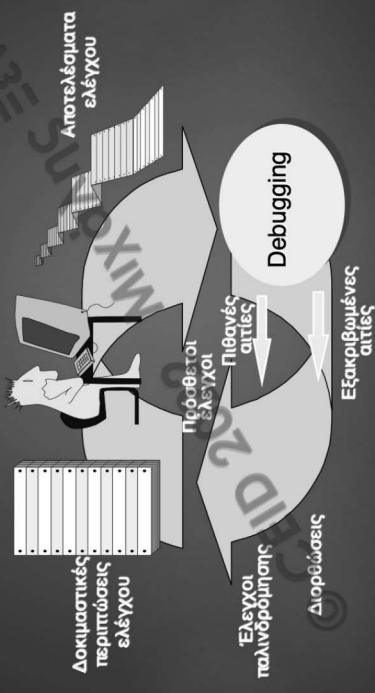
```
for(i=1; i=10; i++)
printf( "%d ", i);
```



Μιχάλης Ξένος, 2022

19

Διαδικασία Αποσφαλμάτωσης



Μιχάλης Ξένος, 2022

Για να θυμηθούμε C

```
int x = 5;
while( x > 0 );
x--;
```

Αυτός είναι ατέρμονος βρόγχος!

```
int x = -5;
while( x < 0 )
x--;
```

```
for(i=1; i=10; i++)
printf( "%d ", i);
```



Μιχάλης Ξένος, 2022

18

Δυσκολίες Εκσφαλμάτωσης

- ▶ Το σύμπλωμα και η αιτία μπορεί να βρίσκονται σε διαφορετικά τμήματα.
- ▶ Προσωρινή εξαφάνιση του σφάλματος με τη διόρθωση κάποιου άλλου άσχετου σφάλματος.
- ▶ Το σύμπλωμα μπορεί να προέρχεται από πρόβλημα στρογγυλοποίησης ή χρονισμού.
- ▶ Αναπαράγωγή των συνθηκών που δημιούργησαν το σφάλμα δύσκολη ή και αδύνατη (ειδικά σε συστήματα πραγματικού χρόνου).
- ▶ Το σφάλμα μπορεί να μην εμφανίζεται πάντα.



Μιχάλης Ξένος, 2022

20

Συνήθη λάθη σε επίπεδο υλοποίησης κώδικα

21

- K1 αλγοριθμικά και εκτελεστικά λάθη (όπως overflow, ασυμφωνία τύπων, λάθη στην ακρίβεια των δεδομένων, κτλ)
- K2 λάθη στις συνθήκες ελέγχου και λογικής (όπως λάθη στις λογικές συνθήκες ή στα όρια δομών επανάληψης)
- K3 λάθη τυπογραφικά
- K4 λάθη αρχικοποίησης
- K5 λάθη στις δομές δεδομένων (κυρίως λανθασμένος ορισμός αλλά και χειρισμός δομών όπως πίνακες, λίστες κτλ.)
- K6 λάθη στην διεπαφή μονάδων
- K7 λάθη στην τεκμηρίωση του κώδικα (μη επαρκής ή ασαφής τεκμηρίωση)
- K8 λάθη στην επικοινωνία/χρήση εξωτερικών πόρων (όπως το υλικό, εξωτερικές βάσεις δεδομένων, αρχεία κτλ.)



Μιχάλης Ξένος, 2022

Ψυχολογική Απόσταση Μεταβλητών

23

stoppt	stoppt	σχεδόν αόρατη
shiftrn	shiftm	αδιόρατη
dcount	bcount	μικρή
claims1	claims2	μικρή
product	sum	μεγάλη



Μιχάλης Ξένος, 2022

Ψυχολογικοί παράγοντες

22

Ένας προγραμματιστής κοιτά αυτό:

```
if ( x < y )
    swap = x;
    x = y;
    y = swap;
```

και βλέπει αυτό:

```
if ( x < y )
{
    swap = x;
    x = y;
    y = swap;
}
```

- ▶ Δεν υπάρχει συγκεκριμένη μεθοδολογία που να εγγυάται επιτυχία!
- ▶ Αναζήτηση επαναλαμβανόμενων δομών στα δεδομένα ελέγχου που προκαλούν το σφάλμα. Π.χ. το σφάλμα συμβαίνει όταν αναζητούμε ένα στοιχείο στο τέλος ενός πίνακα.
- ▶ Χρήση λίστας συνηθισμένων λαθών ανάλογα με το πρόβλημα.
- ▶ Επιτυχία αποσφαλμάτωσης βασίζεται σε συστηματική ανάλυση δεδομένων ελέγχου, ΕΜΠΙΝΕΥΣΗ και ΤΥΧΗ.



Μιχάλης Ξένος, 2022

Μεθοδολογίες αποσφαλμάτωσης

24



Μιχάλης Ξένος, 2022

Μεθοδολογίες αποφασιμάτων

25

- ▶ Βασικές κατηγορίες μεθοδολογιών (δίνουν κατευθυντήριες γραμμές αλλά δεν δίνουν οδηγίες για επιτυχή αποσφαλισμάτωση):
 - ▶ Κατά μέτωπο επίθεση (brute force).
 - ▶ Οπισθοδρόμηση (backtracking).
 - ▶ Εξάλειψη του αιτίου (cause elimination).
 - ▶ Επαγωγή (induction).
 - ▶ Απαγωγή (deduction).



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Τεχνικές

27

- ▶ Εκτύπωση διαγνωστικών μηνυμάτων:
 - ▶ Εκτυπώνουν τιμές μεταβλητών σε σημεία του προγράμματος.
 - ▶ Το σφάλμα βρίσκεται μεταξύ της τελευταίας ορθής και της πρώτης λανθασμένης εμφάνισης της τιμής της μεταβλητής.
 - ▶ Σε περίπτωση που το σφάλμα είναι χρονισμού, οι εντολές εκτύπωσης μπορεί να εμποδίσουν την εμφάνισή του.
 - ▶ Άχρηστες εντολές που πρέπει να αφαιρεθούν στη συνέχεια.



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση

26

- ▶ Παρακάμπτει το νοητικό σκέλος της διαδικασίας.
- ▶ Ο εκκαθαριστής ξεκινά την αποσφαλισμάτωση χωρίς να έχει ιδέα πού μπορεί να βρισκείται το λάθος.
- ▶ Πολύ διαδεδομένη μέθοδος αλλά χρήσιμη μόνο για μικρά προγράμματα.
- ▶ Σε σύνθετα προϊόντα λογισμικού πρέπει να χρησιμοποιείται μόνο αν αποτύχουν όλες οι άλλες μέθοδοι.
- ▶ Καταιγισμός από δεδομένα που πρέπει να αναλυθούν για τον εντοπισμό του σφάλματος.
- ▶ Έχουμε πάντα τον κίνδυνο να χτυπήσουμε σε τσίχο!!!



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Τεχνικές

28

- ▶ Εκτύπωση διαγνωστικών μηνυμάτων - Παράδειγμα. Συνάρτηση που μετρά το πλήθος των αριθμητικών χαρακτήρων σε αλφαριθμητικό. Δεν λειτουργεί σωστά όταν ο πρώτος χαρακτήρας είναι αριθμός:

```
int digits(char str[])  
{  
    int i=0, no_digits=0;  
    while (str[i]!='\0')  
    {  
        if (isdigit(str[i]))  
            no_digits++;  
        i++;  
    }  
    return no_digits;  
}
```

printf("i=%d\t no_digits=%d", i,
no_digits);
Ορθή αρχικοποίηση αλλά στην πρώτη
επανάληψη του while το i είναι 1 και
όχι 0.
ΑΠΑ...



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Τεχνικές

29

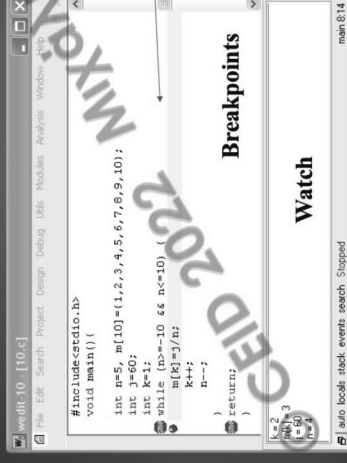
- ▶ Αντίγραφα περιεχομένων μνήμης (Dumps)
 - ▶ Λεπτομερής περιγραφή της μνήμης σε κάποιο σημείο εκτέλεσης.
 - ▶ Χρησιμοποιούνταν ΚΑΠΟΤΕ για αποσφαλμάτωση προγραμμάτων σε Assembly.
 - ▶ Τα αρχεία **'core'** του UNIX είναι αρχεία αντιγράφων μνήμης.
- ▶ Επιλεκτική ιχνήλαση μεταβλητών (trace):
 - ▶ Λεπτομερής καταγραφή όλων των διαδοχικών καταστάσεων επιλεγμένων μεταβλητών.



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Σημεία διακοπής

31



Εντολή υπό εκτέλεση



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Τεχνικές

30

- ▶ Σημεία διακοπής (breakpoints):
 - ▶ Διακοπή της εκτέλεσης και μεταφορά ελέγχου στον εκκαθαριστή (που μπορεί έπειτα να τρέξει το πρόγραμμα γραμμή-γραμμή).
 - ▶ Εξέταση ή/και τροποποίηση τιμών των μεταβλητών του προγράμματος.
- ▶ Σημεία διακοπής υπό συνθήκη (conditional breakpoints):
 - ▶ Η ροή του προγράμματος διακόπτεται όταν ικανοποιείται κάποια συνθήκη επί των μεταβλητών.



Μιχάλης Ξένος, 2022

Κατά Μέτωπο Επίθεση - Τεχνικές

32

- ▶ Σημεία διακοπής - Παρόδειγμα (το προηγούμενο)

```
int digits(char str[])  
{  
    int i=0, no_digits=0;  
    while (str[i++]!='\0')  
    {  
        if (isdigit(str[i]))  
            no_digits++;  
        i++;  
    }  
    return no_digits;  
}
```

Breakpoint στην αρχή του while
ώστε να ελέγχουμε τις τιμές των
μεταβλητών.



Μιχάλης Ξένος, 2022

Οπισθοδρόμηση (backtracking)

33

- ▶ Ξεκινάμε από το σημείο που παρατηρήθηκε το σφάλμα.
- ▶ Ιχνηλατούμε την εκτέλεση του κώδικα προς τα πίσω μέχρι να βρούμε το ελάττωμα που προκάλεσε το σφάλμα.
- ▶ Χρήσιμη μέθοδος για μικρού μεγέθους προγράμματα.
- ▶ Βήματα:
 - ▶ Ξεκινάμε με δεδομένο το αποτέλεσμα.
 - ▶ Τρέχουμε το πρόγραμμα προς τα πίσω και υπολογίζουμε την τιμή των μεταβλητών σε προηγούμενα σημεία.
 - ▶ Χρησιμοποιούμε εντολές εκτύπωσης διαγνωστικών μηνυμάτων για να επιβεβαιώσουμε τους υπολογισμούς μας.
 - ▶ Εντοπίζουμε την περιοχή του κώδικα (βέλπιστα τη γραμμή του κώδικα) όπου υπάρχει το σφάλμα.



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Περιπτώσεις Ελέγχου

35

- ▶ Για τη δημιουργία ορθών υποθέσεων σχετικά με το σφάλμα πρέπει να λαμβάνουμε υπόψη διαφορετικές περιπτώσεις ελέγχου.
 - ▶ Περιπτώσεις ορθής αλλά και λανθασμένης εκτέλεσης του προγράμματος.
- ▶ Πχ. αν στο πραγματοποιούμενο πρόγραμμα που αναζητά πλήθος αριθμητικών χαρακτήρων σε ένα αλφαριθμητικό είχαμε μόνο περιπτώσεις ελέγχου όπου ο πρώτος χαρακτήρας ήταν πάντα αριθμός, θα πιστεύαμε ότι το πρόγραμμα δεν λειτουργεί ποτέ ορθά.
- ▶ Πρόβλημα για την ανάπτυξη της πιθανότερης υπόθεσης.



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Cause Elimination

34

- ▶ Τα σφάλματα εντοπίζονται και διορθώνονται ακολουθώντας μια βηματική διαδικασία επίλυσης προβλημάτων.
- ▶ Απαιτεί μελέτη των αποτελεσμάτων των ελέγχων πριν από την προσπάθεια διόρθωσης.
 - ▶ Πότε το πρόγραμμα λειτουργεί ορθά και πότε λανθασμένα.
 - ▶ Κάτω από ποιες συνθήκες (hardware, software) προκύπτει σφάλμα και κάτω από ποιες όχι.
- ▶ Έλεγχος πότε και κάτω από ποιες συνθήκες εμφανίζεται το λάθος.
- ▶ Νοητική διαδικασία (εξχνίαση του εγκλήματος...).



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Επαγωγή (Induction)

36

- ▶ Διαδικασία:
 - ▶ Προσδιορισμός του προβλήματος (μέσω των αποτελεσμάτων των ελέγχων).
 - ▶ Ανάπτυξη κάποιας υπόθεσης (αναφορικά με το σφάλμα). Αρχικά η πιο πιθανή υπόθεση.
 - ▶ Έλεγχος της υπόθεσης.
 - ▶ Αν η υπόθεση επαληθευτεί, επίλυση του προβλήματος.
 - ▶ Διαφορετικά μεταβαίνουμε στο βήμα 2.



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Επαγωγή – Παράδειγμα

37

```
int digits(char str[])
{
    int i=0, no_digits=0;
    while (str[i++]!='\0')
    {
        if (isdigit(str[i]))
            no_digits++;
    }
    return no_digits;
}
```

- ▶ Δοκιμαστικές Περιπτώσεις:
 - ▶ abc123 → Λειτουργεί ορθά... Επιστρέφει 3
 - ▶ 123abc → Λειτουργεί λανθασμένα... Επιστρέφει 2
 - ▶ 1a2b3c → Λειτουργεί λανθασμένα... Επιστρέφει 2
 - ▶ a1b2c3 → Λειτουργεί ορθά... Επιστρέφει 3
- ▶ Πιθανότερη υπόθεση:
 - ▶ Δεν εξετάζει τον πρώτο χαρακτήρα.
 - ▶ Γιατί;
 - ▶ Λάθος αρχικοποίηση.
 - ▶ Λάθος στην προσαύξηση του i.



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Απαγωγή – Παράδειγμα

39

```
int digits(char str[])
{
    int i=0, no_digits=0;
    while (str[i++]!='\0')
    {
        if (isdigit(str[i]))
            no_digits++;
    }
    return no_digits;
}
```

- ▶ Δοκιμαστικές Περιπτώσεις:
 - ▶ abc12 → Λειτουργεί ορθά... Επιστρέφει 2
 - ▶ 123abc → Λειτουργεί λανθασμένα... Επιστρέφει 2
 - ▶ 1a2b3c → Λειτουργεί λανθασμένα... Επιστρέφει 2
 - ▶ a1b2c3 → Λειτουργεί ορθά... Επιστρέφει 3
- ▶ Πιθανότερες υποθέσεις:
 - ▶ Δεν λειτουργεί για όριο αριθμού αριθμητικών χαρακτήρων. Απορρίπτεται από την 4η δοκιμή.
 - ▶ Δεν εξετάζει τον πρώτο χαρακτήρα. Εξηγεί τα δεδομένα.
 - ▶ Εκλείπουνσι:
 - ▶ Λάθος αρχικοποίηση.
 - ▶ Λάθος προσαύξηση.



Μιχάλης Ξένος, 2022

Εξάλειψη του Αιτίου – Απαγωγή (Deduction)

38

- ▶ Διαδικασία:
 - ▶ Απαρίθμηση πιθανών υποθέσεων.
 - ▶ Απόρριψη των υποθέσεων με βάση τα αποτελέσματα των ελέγχων.
 - ▶ Αν δεν μείνει καμία πιθανή υπόθεση συλλογή και άλλων δεδομένων.
 - ▶ Εκλείπουνσι της πιθανής υπόθεσης.
 - ▶ Αν η υπόθεση επαληθευτεί, επίλυση του προβλήματος.
 - ▶ Διαφορετικά συλλέγουμε επιπλέον δεδομένα και μεταβαίνουμε στο βήμα 1.



Μιχάλης Ξένος, 2022

Οδηγίες Διόρθωσης Σφάλματος

40

- ▶ Μη βιάζεστε να διορθώσετε το λάθος.
 - ▶ Ερευνήστε και άλλα λάθη στην ίδια περιοχή.
- ▶ Εντοπίστε και άλλα σημεία που μπορεί να εμφανίζεται το ίδιο λάθος.
- ▶ Ποιο άλλο λάθος μπορεί να δημιουργηθεί με τη διόρθωση του υπάρχοντος;
- ▶ Τι θα μπορούσατε να κάνετε για να αποφύγετε το σφάλμα; Συμβάλει στη διόρθωση της διαδικασίας και όχι μόνο του σφάλματος.
- ▶ Δημιουργήστε λίστα με τα σημαντικά σφάλματα που κάνετε ώστε να ανατρέχετε σε αυτά όταν ψάχνετε για πιθανές υποθέσεις.



Μιχάλης Ξένος, 2022

Έλεγχος Παλινδρόμησης

41

- ▶ Τι είναι;
 - ▶ εκ νέου εκτέλεση των δοκιμαστικών περιπτώσεων που αποκάλυψαν το σφάλμα.
- ▶ Γιατί γίνεται; Όστε να βεβαιωθεί ο εκκαθαριστής ότι:
 - ▶ δεν παρουσιάζεται πια το ίδιο σφάλμα,
 - ▶ δεν εμφανίστηκαν νέα σφάλματα ως αποτέλεσμα της επέμβασής του στον κώδικα.
- ▶ Εάν επιτύχει -> επανάληψη εκσφαλμάτωσης

Διαδικαστικά

Τα λέμε στις εξετάσεις

Προσοχή: Μάλλον θα γίνουν μόνο σε μια αίθουσα!

Αφού το μάθημα το παρακολουθούν 15-20 άτομα γιατί να χρειαστεί και άλλη, σωστά;

Άρα δείτε προσεκτικά ανακοίνωση και μην έρθετε όπου εσείς διαλέξετε!!