

## Πέτρα-Ψαλίδι-Χαρτί

Μια διαφορετική εκδοχή του γνωστού παιχνιδιού: ο υπολογιστής επιλέγει 10 κινήσεις (πέτρα ή ψαλίδι ή χαρτί) και ο παίκτης επίσης, χωρίς να γνωρίζει ο ένας τι διάλεξε ο άλλος. Οι επιλογές των δύο αντιπάλων εμφανίζονται όλες μαζί στην οθόνη και συγκρίνονται μία προς μία. Νικητής είναι όποιος έχει νικήσει στις περισσότερες επιλογές του.



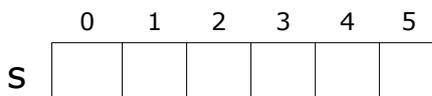
### Σημαντική Ερώτηση 1: πως θα αποθηκεύσουμε τις επιλογές των δύο αντιπάλων;

Μπορούμε να πάρουμε 10 μεταβλητές για τον υπολογιστή και 10 μεταβλητές για τον παίκτη.

Μπορούμε να χρησιμοποιήσουμε 1 συμβολοσειρά για τον υπολογιστή (με δέκα θέσεις) και 1 συμβολοσειρά για τον παίκτη (επίσης με 10 θέσεις).

Τι είναι πιο βολικό για εμάς ως προγραμματιστές;

Φανταζόμαστε τις συμβολοσειρές ως μια σειρά από διαδοχικά κουτάκια, κολλημένα το ένα στο άλλο. Όλα μαζί έχουν ένα όνομα, αυτό της συμβολοσειράς. Για να ξεχωρίσουμε το κάθε κουτάκι, χρησιμοποιούμε αριθμούς ξεκινώντας από το 0 και προχωρώντας κατά 1.



### Κώδικας - Βήμα 1ο: function που θα δημιουργεί την συμβολοσειρά με τις 10 επιλογές του παίκτη



Ανοίγουμε ένα νέο αρχείο στο IDLE. Ο πιο κάτω κώδικας δημιουργεί την συμβολοσειρά playerChoices και την επιστρέφει (με την εντολή return), για να μπορεί να χρησιμοποιηθεί και σε άλλα τμήματα του προγράμματος.

Μπορείτε να εξηγήσετε τι ακριβώς κάνει αυτή η συνάρτηση και πως δημιουργείται η συμβολοσειρά playerChoices;

Δοκιμάστε την λειτουργία της, γράφοντας στη συνέχεια ένα μικρό κυρίως πρόγραμμα:

```
playerCh = player_plays()
print(playerCh)
```

```
petrapsalidixarti.py
File Edit Format Run Options Window Help

def player_plays():
    playerChoices = " # κενή συμβολοσειρά, αρχικά

    for i in range(1, 11):
        print("Κίνηση ", i, "(Π ή Ψ ή Χ): ")
        x = input()
        playerChoices = playerChoices + x

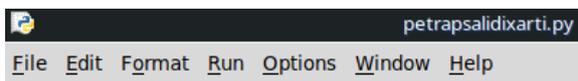
    return playerChoices
```

# 10

**Κώδικας - Βήμα 2ο:** function που θα δημιουργεί την συμβολοσειρά με τις 10 επιλογές του υπολογιστή



Ο υπολογιστής θα επιλέγει τυχαία. Χρησιμοποιήστε την εντολή `randint(1,3)` και αν έρθει 1 τότε ο υπολογιστής επιλέγει Π, αν έρθει 2 επιλέγει Ψ, αν έρθει 3 επιλέγει Χ.



```
def computer_plays():
    computerChoices = " # κενή συμβολοσειρά, αρχικά

    for i in range(1, 11):
        x = random.randint(1,3)

        if x == 1:
            computerChoices = computerChoices + 'Π'
        elif .....:
            .....
        else:
            .....

    return .....
```

Ονομάστε την συνάρτηση `computer_plays()` και τη συμβολοσειρά που θα επιστρέφει ονομάστε την `computerChoices`.

Στο δοκιμαστικό κυρίως πρόγραμμα που έχετε ήδη, προσθέστε εντολές ώστε να εμφανίζεται και η συμβολοσειρά που υπολογιστή.

**Κώδικας - Βήμα 3ο:** function που θα συγκρίνει τις συμβολοσειρές και θα εμφανίζει τον νικητή



Η function αυτή για να κάνει τη δουλειά της, πρέπει να έχει τις δύο συμβολοσειρές: τις επιλογές του παίκτη και τις επιλογές του υπολογιστή.

Άρα μέσα στην παρένθεση θα υπάρχουν δύο παράμετροι.

Δείτε πιο κάτω έναν αρχικό κώδικα για τη συνάρτηση αυτή και συμπληρώστε τον.

# 10

```
petrapsalidixarti.py - /home/eleni/Εγγραφα/PYTHON/python2025/petrapsalidixarti.py (3)
File Edit Format Run Options Window Help
def compare(pl, comp):
    player_wins = 0 # αρχικές τιμές
    comp_wins = 0
    draws = 0

    for i in range(0, 10):
        # αν ίδια κίνηση
        if pl[i] == comp[i]:
            draws = draws + 1
        # αν ισχυρότερη κίνηση παίκτη
        elif (pl[i] == 'Π' and comp[i] == 'Ψ') or (pl[i] == 'Ψ' and comp[i] == 'Π') or (pl[i] == 'Π' and comp[i] == 'Π') or (pl[i] == 'Ψ' and comp[i] == 'Ψ'):
            player_wins = player_wins + 1
        # αλλιώς, ισχυρότερη κίνηση υπολογιστή
        else:
            .....

    print("Νίκες παίκτη: ", ..... )
    print("Νίκες υπολογιστή: ", ..... )
    print("Ισοπαλίες: ", ..... )

    if .....:
        print("Νίκησε ο παίκτης!")
    else:
        print("Νίκησε ο υπολογιστής!")
```

## Κώδικας - Βήμα 4ο: Το κυρίως πρόγραμμα που θα καλεί τις functions

Πιο κάτω δίνονται δύο εκδοχές κυρίως προγράμματος. Εντοπίστε τις διαφορές τους και δείτε αν είναι ισοδύναμα ή όχι.



```
# κυρίως πρόγραμμα - εκδοχή 1η
```

```
a = player_plays()
b = computer_plays()
print("Επιλογές υπολογιστή: ", b)
compare(a, b)
```

```
# κυρίως πρόγραμμα - εκδοχή 2η
```

```
playerChoices = player_plays()
computerChoices = computer_plays()
print("Επιλογές υπολογιστή: ", computerChoices)
compare(playerChoices, computerChoices)
```

Τι θα συμβεί αν η κλήση της `compare(a, b)` γραφεί ως `compare(b, a)`; Παιζει ρόλο η σειρά των παραμέτρων (δηλ. των αντικειμένων που μπαίνουν στην παρένθεση) ;

