



Python i Pygame

Programiranje arkadnih igrica u Pythonu koristeći Pygame

Sortiranje



Uvod u sortiranje



- Sortiranje je preslagivanje elemenata nekog niza tako da su poredani po veličini
- Ovdje ćemo spomenuti dva algoritma za sortiranje:
 - Seleksijsko (Selection sort)
 - Krećemo od prvog elementa niza, zatim skeniramo niz do kraja da nađemo najmanji element i njima zamijenimo mjesta. Nakon toga ponavljamo postupak sa drugim elementom niza.
 - Umetanjem (Insertion sort)
 - Krećemo od prvog elementa na desno. Ako je slijedeći element niza manji od nekog od prethodnih, zamijenimo im mjesta.
- Prvo trebamo naučiti kako zamijeniti vrijednosti dva elementa niza. Za to koristimo privremenu varijablu:

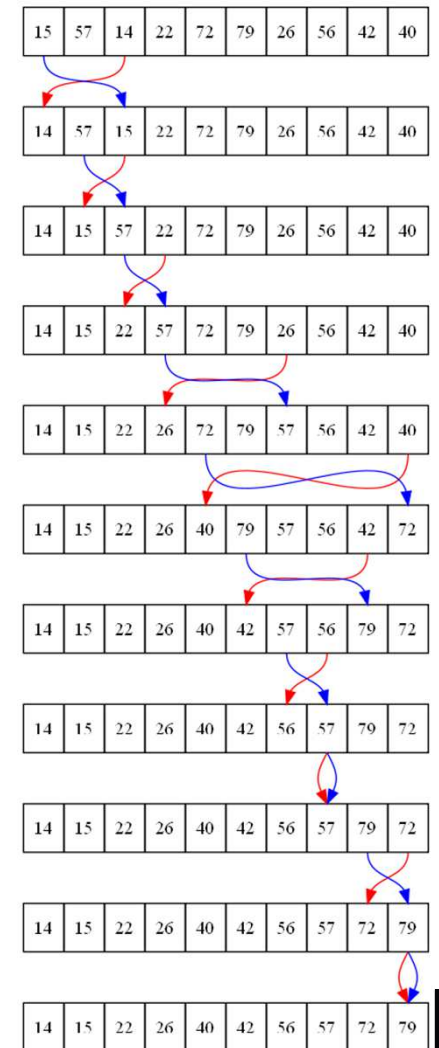
```
# Zamjena vrijednosti elemenata my_list[0] i my_list[2]
temp = my_list[0]
my_list[0] = my_list[2]
my_list[2] = temp
```

Selekcijsko sortiranje



- Primjer funkcije koja radi selekcijsko sortiranje, poziva se iz glavnog programa, ulazni parametar je početni niz a vraća sortirani niz

```
def selection_sort(my_list):  
    # Funkcija za sortiranje liste korištenjem selekcijskog  
    # sortiranja  
    # Petlja koja prolazi kroz cijelu listu  
    for cur_pos in range(len(my_list)):  
        # Traži najmanji broj, počni od trenutne pozicije  
        min_pos = cur_pos  
        # Provjeri od trenutne pozicije u desno do kraja liste  
        for scan_pos in range(cur_pos + 1, len(my_list)):  
            # Da li je trenutna pozicija najmanja?  
            if my_list[scan_pos] < my_list[min_pos]:  
                # Ako da, zapamti je  
                min_pos = scan_pos  
        # Zamijeni trenutnu poziciju i najmanju poziciju  
        temp = my_list[min_pos]  
        my_list[min_pos] = my_list[cur_pos]  
        my_list[cur_pos] = temp  
    # Vraća sortirani niz  
    return(my_list)
```

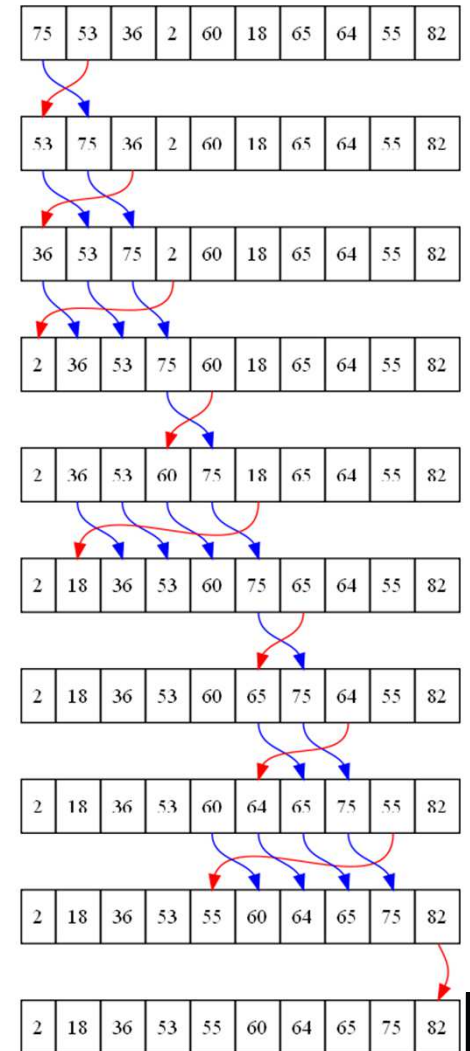


Sortiranje umetanjem



- Primjer funkcije koja radi sortiranje umetanjem, poziva se iz glavnog programa, ulazni parametar je početni niz a vraća sortirani niz

```
def insertion_sort(my_list):  
    # Funkcija za sortiranje liste korištenjem sortiranja  
    umetanjem  
    # Kreni od drugog elementa niza (pos 1)  
    # i umetni ga u listu  
    for key_pos in range(1, len(my_list)):  
        # Procitaj vrijednost trenutnog elementa  
        key_value = my_list[key_pos]  
        # Kreni od trenutnog elementa u lijevo prema pocetku  
        # liste  
        scan_pos = key_pos - 1  
        # Provjeravaj svaki elemente dok ne stignes do pocetka  
        # liste ili ne nadjes manji element  
        while (scan_pos >= 0) and (my_list[scan_pos] >  
key_value):  
            # Postavi trenutni element ispred tekuceg u petlji  
            my_list[scan_pos + 1] = my_list[scan_pos]  
            # pomakni se za jedno mjesto u lijevo  
            scan_pos = scan_pos - 1  
        # Postavi trenutnu vrijednost na pravu lokaciju  
        my_list[scan_pos + 1] = key_value  
    # Vрати sortirani niz  
    return(my_list)
```



Kviz



- Slijedi link ispod:

<http://programarcadegames.com/quiz/quiz.php?file=sorting&lang=en>



Kviz odgovori



- P1: How many lines of code are normally used to swap two values?
 - 5
 - 2
 - 4
 - 3
- P2: What is key in writing code to properly swap two values?
 - Using a variable to temporarily hold one of the values while swapping.
 - Make sure you use the == operator rather than the = operator.
 - Using the swap operator.

Kviz odgovori, nastavak



- P3: In the selection sort, what does the outside loop to?
 - Count the number of items in the list.
 - Finds the smallest value in the list.
 - Selects the next element that we will be placing the smallest remaining value into.
- P4: In the selection sort, what does the inside loop to?
 - Selects the next element that we will be placing the smallest remaining value into.
 - Finds the smallest value in the list.
 - Count the number of items in the list.



Kviz odgovori, nastavak



- P5: In the insertion sort, what does the outside loop to?
 - Finds the smallest value in the list.
 - Slides an element into a sorted position.
 - Selects the next element to be slid into a sorted position.
- P6: In the insertion sort, what does the inside loop to?
 - Selects the next element to be slid into a sorted position.
 - Finds the smallest value in the list.
 - Slides an element into a sorted position.



Kviz odgovori, nastavak



- P7: If the selection sort and insertion sort run in n^2 time, what is n ?
 - The size of each element.
 - The number of lines of code.
 - The time it takes to sort in milliseconds.
 - The number of elements to sort.
- P8: If the selection sort and insertion sort run in n^2 time, what does that mean if I have a problem size of 100 ($n = 100$) and increase it by ten times to $n = 1000$?
 - The 1,000 elements will take about 10 times longer to sort than a list of 100 elements.
 - The 1,000 elements will take about 1,000 times longer to sort than a list of 100 elements.
 - The 1,000 elements will take about 4 times longer to sort than a list of 100 elements.
 - The 1,000 elements will take about 2 times longer to sort than a list of 100 elements.
 - The 1,000 elements will take about 100 times longer to sort than a list of 100 elements.



Kviz odgovori, nastavak



- P9: What type of list does the insertion sort work particularly well on?
 - A list that is in reverse order.
 - A randomly sorted list.
 - A list that already close to being in order.