

```
#-----
# Name:      Higher 2021 Assignment Program
# Purpose:   Working Example (parallel arrays)
#
# Author:    Mr Simpson
#-----
```

Trinket link:

<https://trinket.io/python3/7c21c32133>

```
# 1. Get qualifying athletes' data
# Data read from file into 5 parallel arrays
def getAtheleteData(entryID, location, firstName,surname,jumps):
```

```
    counter = 0
    with open("athletes.csv") as readfile:
        line = readfile.readline().rstrip("\n")
        while line:
            items = line.split(",")
            entryID[counter] = items[0]
            location[counter] = items[1]
            firstName[counter] = items[2]
            surname[counter] = items[3]
            jumps[counter] = int(items[4])

            line = readfile.readline().rstrip("\n")

            counter +=1

    return entryID, location, firstName,surname,jumps
```

```
#-----
```

```
# 2. Generate bib values and write to file using sub-string and character to ASCII conversion
```

```
def generateBibValues(entryID,location,firstName,surname):
    with open('bibValues.csv','w') as athletes:
        for loop in range(len(location)):
            bibValue = firstName[loop][0:1] + surname[loop] + str(ord(location[loop][0:1]))
            athletes.write(bibValue + "\n")
```

```
#-----
```

Program top-level design (pseudocode)

- | | |
|---|---|
| 1. Get qualifying athletes' data | OUT: entryID(), location(), forename(),
surname(), jumps() |
| 2. Generate bib values and write to new file with
entry IDs | IN: entryID(), location(), forename(), surname() |
| 3. Find the highest number of jumping jacks
completed | IN: jumps(),
OUT: maxJumps |
| 4. Display the full name of the athlete(s) who
completed the highest number of jumping jacks | IN: maxJumps, forename(), surname(), jumps() |

Running: 2021 Higher Solution - Parallel Arrays.py

```
Max Hughes completed the most jumps
Chloe McAninch completed the most jumps
Satbinder Sandhu completed the most jumps
*****
The number of finalists from Coatbridge are: 6
The number of finalists from Kirkcaldy are: 7
The number of finalists from Inverness are: 8
The number of finalists from Motherwell are: 9
```

3. Find max algorithm to find the highest number of jumps

```
def findMax(jumps):  
    maxJumps = jumps[0]  
    for loop in range(1,len(jumps)):  
        if jumps[loop] > maxJumps:  
            maxJumps = jumps[loop]  
  
    return maxJumps
```

#-----

4. Display full name of athletes with jumps that match the maximum above

```
def displayMostJumps(maxJumps,jumps,firstName,surname):  
    for loop in range(1,len(jumps)-1):  
        if jumps[loop] == maxJumps:  
            print (firstName[loop],surname[loop],"completed the most jumps")
```

#-----

#Task 1c(ii) Count finalists at each location

```
def countFinalists(location):  
    cbridge = 0  
    kcaldy = 0  
    inver = 0  
    motherw = 0  
  
    for index in range(len(location)):  
        if location[index] == "Coatbridge":  
            cbridge +=1  
  
        if location[index] == "Kirkcaldy":  
            kcaldy +=1  
  
        if location[index] == "Inverness":  
            inver +=1  
  
        if location[index] == "Motherwell":
```

```
motherw +=1
```

```
print("*****")
print("The number of finalists from Coatbridge are: ",cbridge)
print("The number of finalists from Kirkcaldy are: ",kcaldy)
print("The number of finalists from Inverness are: ",inver)
print("The number of finalists from Motherwell are: ",motherw)
```

```
#-----
```

```
## Main program
```

```
entryID = [""]* 30
```

```
location = [""]*30
```

```
firstName = [""]*30
```

```
surname = [""]*30
```

```
jumps = [0]*30
```

```
maxJumps = 0
```

```
#1. Get qualifying athletes' data
```

```
entryID,location,firstName,surname,jumps = getAtheleteData(entryID,location,firstName,surname,jumps)
```

```
#2. Generate bib values and write to new file with entry IDs
```

```
generateBibValues(entryID,location,firstName,surname)
```

```
#3. Function to find the highest number of jumping jacks completed. Returns maxJumps
```

```
maxJumps = findMax(jumps)
```

```
#4.Display the full name of the athlete(s) who completed the highest number of jumping jacks
```

```
displayMostJumps(maxJumps,jumps,firstName,surname)
```

```
#Task 1c(ii) Count occurrences for finalists in each location
```

```
countFinalists(location)
```