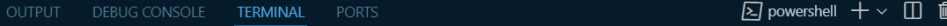


Part 1 – Data Structure and Algorithm

1. Problem 1 – Remove Duplicates

```
def remove_duplicates(array):
    unique_elements = list(set(array))
    unique_elements.sort(key=array.index)
    return len(unique_elements)

print(remove_duplicates([2, 3, 3, 3, 6, 9, 9]))
print(remove_duplicates([2, 3, 4, 5, 6, 9, 9]))
print(remove_duplicates([2, 2, 2, 11]))
print(remove_duplicates([2, 2, 2, 11]))
print(remove_duplicates([1, 2, 3, 11, 11]))
```



The screenshot shows a Windows PowerShell terminal window with a dark blue background. The title bar at the top includes tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is active), and 'PORTS'. On the right side of the title bar are icons for a powershell window, a plus sign, a dropdown arrow, a window icon, a trash icon, and ellipsis, up, and close buttons. The terminal content shows the command prompt 'PS C:\Users\tawhe\Documents\Belajar Python>' followed by the command 'python Data_Structure_and_Algorithm_Part1.py'. Below the command, the output '4', '6', '2', '2', and '4' is displayed on separate lines. A large green arrow points from the left towards the output lines. The prompt 'PS C:\Users\tawhe\Documents\Belajar Python>' is repeated at the bottom of the terminal window.

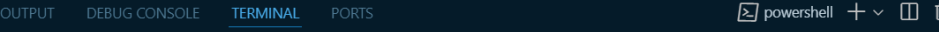
2. Problem 2 – Prima ke X

```
def is_prime(n):
    if n <= 1:
        return False
    for i in range(2, int(n**0.5) + 1):
        if n % i == 0:
            return False
    return True

def primeX(x):
    count = 0
    num = 1
    while count < x:
        num += 1
        if is_prime(num):
            count += 1
    return num
```



The screenshot shows a Windows PowerShell terminal window with a dark blue background. The title bar at the top includes tabs for 'PROBLEMS', 'OUTPUT', 'DEBUG CONSOLE', 'TERMINAL' (which is active), and 'PORTS'. On the right side of the title bar are icons for a terminal, a plus sign, a minus sign, a window icon, a trash icon, and a close button. The terminal content shows the command prompt 'PS C:\Users\tawhe\Documents\Belajar Python>' followed by the command 'python Data_Structure_and_Algorithm_Part1.py'. Below the command, the numbers '2', '11', '19', '23', and '29' are listed vertically, indicating line numbers or output. A large green arrow points from the right towards the terminal output area.



The screenshot shows a Windows terminal window with the following content:

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + - [ ] [X] ... ^ X
```

```
PS C:\Users\tawhe\Documents\Belajar Python> python Data_Structure_and_Algorithm_Part1.py
0
1
34
55
144
PS C:\Users\tawhe\Documents\Belajar Python> 
```

A large green arrow points to the number 34 in the output.

```
def is_prime(n):
    if n <= 1:
        return False
    if n <= 3:
        return True
    if n % 2 == 0 or n % 3 == 0:
        return False
    i = 5
```

```

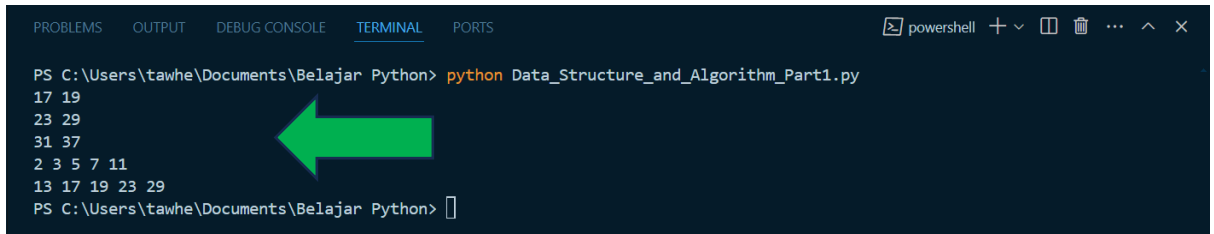
while i * i <= n:
    if n % i == 0 or n % (i + 2) == 0:
        return False
    i += 6
return True

def next_prime(start):
    prime = start
    while True:
        prime += 1
        if is_prime(prime):
            return prime

def generate_primes_grid(width, height, start):
    result = ""
    current_prime = start
    for _ in range(height):
        row = []
        for _ in range(width):
            current_prime = next_prime(current_prime)
            row.append(current_prime)
        result += " ".join(map(str, row)) + "\n"
    return result.strip()

print(generate_primes_grid(2, 3, 13))
print(generate_primes_grid(5, 2, 1))

```



```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\tawhe\Documents\Belajar Python> python Data_Structure_and_Algorithm_Part1.py
17 19
23 29
31 37
2 3 5 7 11
13 17 19 23 29
PS C:\Users\tawhe\Documents\Belajar Python>

```

5. Problem 5 – Total Maksimum dari Deret Bilangan

```

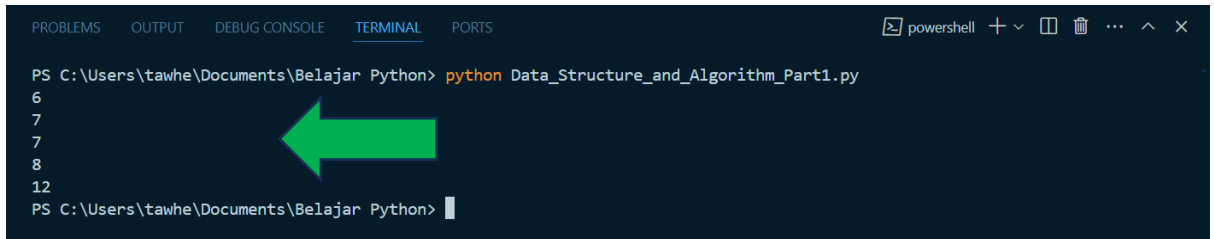
def max_sequence(arr):
    max_so_far = float('-inf')
    max_ending_here = 0

    for number in arr:
        max_ending_here += number
        if max_so_far < max_ending_here:
            max_so_far = max_ending_here
        if max_ending_here < 0:
            max_ending_here = 0

    return max_so_far

```

```
print(max_sequence([-2, 1, -3, 4, -1, 2, 1, -5, 4]))  
print(max_sequence([-2, -5, 6, -2, -3, 1, 5, -6]))  
print(max_sequence([-2, -3, 4, -1, -2, 1, 5, -3]))  
print(max_sequence([-2, -5, 6, -2, -3, 1, 6, -6]))  
print(max_sequence([-2, -5, 6, 2, -3, 1, 6, -6]))
```



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS powershell + - [ ] [X] ... ^ X  
PS C:\Users\tawhe\Documents\Belajar Python> python Data_Structure_and_Algorithm_Part1.py  
6  
7  
7  
8  
12  
PS C:\Users\tawhe\Documents\Belajar Python> |
```