

시간복잡도의 이해 연습문제

아래 코드 들에 대한 시간 복잡도를 계산하시오.

함수 이름은 모두 임의로 정의된 것이므로 별다른 의미가 없습니다.

문제 1

python

```
def myfunction(arr):  
    print(arr[0])  
    print(arr[0])  
  
arr = [5, 2, 3, 4, 1]  
myfunction(arr)
```

C++

```
#include <iostream>  
  
void myfunction(int arr[]) {  
    std::cout << arr[0] << std::endl;  
    std::cout << arr[0] << std::endl;  
}  
  
int main() {  
    int arr[] = {5, 2, 3, 4, 1};  
    myfunction(arr);  
    return 0;  
}
```

문제 2

Python

```
def mytest(n):  
    for i in range(n): # n번 반복  
        print(i)      # n번 출력  
  
mytest(5)
```

C++

```
#include <iostream>

void mytest(int n) {
    for (int i = 0; i < n; ++i) { // n번 반복
        std::cout << i << std::endl; // n번 출력
    }
}

int main() {
    mytest(5);
    return 0;
}
```

문제 3

Python

```
def oktime(n):
    for i in range(1, n):
        for j in range(1, n):
            print("안녕?")

oktime(5)
```

C++

```
#include <iostream>

void oktime(int n) {
    for (int i = 1; i < n; ++i) {
        for (int j = 1; j < n; ++j) {
            std::cout << "안녕?" << std::endl;
        }
    }
}

int main() {
    oktime(5);
}
```

문제 4

Python

```
def foobar(A, n):
    total_sum = 0
```

```

    for i in range(0, n - 1):
        for j in range(i + 1, n):
            total_sum += A[i] * A[j]

    return total_sum

A = [1, 2, 3, 4, 5]
n = len(A)
result = foobar(A, n)
print("결과:", result)

```

C++

```

#include <iostream>

int foobar(int A[], int n) {
    int total_sum = 0;

    for (int i = 0; i < n - 1; ++i) {
        for (int j = i + 1; j < n; ++j) {
            total_sum += A[i] * A[j];
        }
    }

    return total_sum;
}

int foobar() {
    int A[] = {1, 2, 3, 4, 5};
    int n = sizeof(A) / sizeof(A[0]);

    int result = sample5(A, n);
    std::cout << "결과: " << result << std::endl;

    return 0;
}

```

문제 5

Python

```

import random

def helloworld(A, n):
    total_sum = 0

    for i in range(n):
        for j in range(n):

```

```

        k = max(random.sample(A, n // 2))
        total_sum += k

    return total_sum

A = [1, 2, 3, 4, 5, 6]
n = len(A)
result = helloworld(A, n)
print("결과:", result)

```

C++

```

#include <iostream>
#include <algorithm>
#include <random>

int helloworld(int A[], int n) {
    int total_sum = 0;
    std::random_device rd;
    std::mt19937 g(rd()); // 난수생성기

    for (int i = 0; i < n; ++i) {
        for (int j = 0; j < n; ++j) {
            int temp[n];
            std::copy(A, A + n, temp);
            std::shuffle(temp, temp + n, g);
            int max_val = *std::max_element(temp, temp + n / 2);
            total_sum += max_val;
        }
    }
    return total_sum;
}

int main() {
    int A[] = {1, 2, 3, 4, 5, 6};
    int n = sizeof(A) / sizeof(A[0]);

    int result = helloworld(A, n);
    std::cout << "결과: " << result << std::endl;

    return 0;
}

```

문제 6

Python

```

def niceTime(n):
    i = 1

```

```

while i < n:
    print(i)
    i = i * 2

niceTime(16)

```

C++

```

#include <iostream>

void niceTime(int n) {
    int i = 1;
    while (i < n) {
        std::cout << i << std::endl;
        i = i * 2;
    }
}

int main() {
    niceTime(16);
    return 0;
}

```

문제 7

Python

```

def doit(n):
    i = n
    while i > 1:
        print(i)
        i = i // 2

doit(16)

```

C++

```

#include <iostream>

void doit(int n) {
    int i = n;
    while (i > 1) {
        std::cout << i << std::endl;
        i = i / 2;
    }
}

```

```
int main() {  
    doit(16);  
    return 0;  
}
```

문제 8

Python

```
def foo(n):  
    for i in range(1, n):  
        j = 1  
        while j < n:  
            print(i, j)  
            j = j * 2  
  
foo(16)
```

C++

```
#include <iostream>  
  
void foo(int n) {  
    for (int i = 1; i < n; ++i) {  
        int j = 1;  
        while (j < n) {  
            std::cout << i << " " << j << std::endl;  
            j = j * 2;  
        }  
    }  
}  
  
int main() {  
    foo(16);  
    return 0;  
}
```