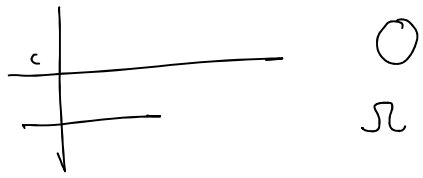
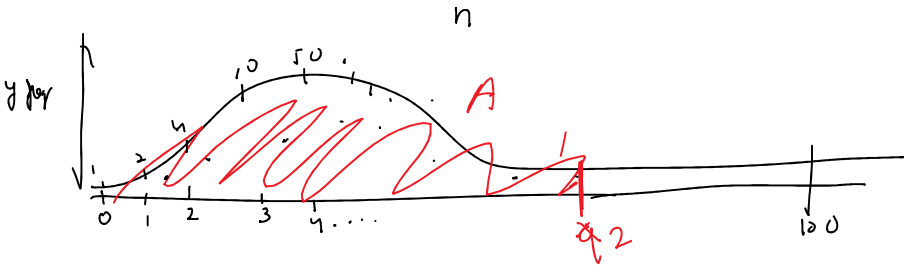


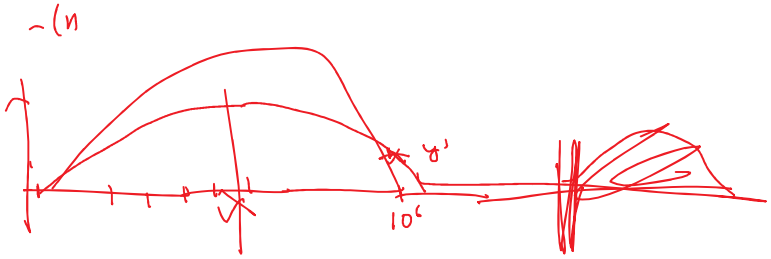
$f(n) = 2 + 3l = 2 + 3n \rightarrow O(n)$

$l \hookrightarrow n$



(A)

Total ans



{ }

$K + C \cdot \log n = f(n)$

$\hookrightarrow O(\quad n \qquad f(n) = 1 + 2(1 + \log \frac{n}{2})$

$f(n) = 1 + 2n \rightarrow O(n)$

```
while (i<=n){
  i++;
}
```

```
w( ){
 i = i x 2 i
}
```

```
while (i<=n){
  i = i+2
}
```

```
w( ){
 i = i x 3 }
```

$f(n) = 1 + 2 \cdot l$

```
while (  ){
  i = i+2
  i = i+3
}
```

$1 + 2l$ = $f(n)$

```
w( ){
  i = i x 2
  i = i x 3
}
```

```
while ( ) {
```

```
w( ) {
```

l	i
1	1
2	K
3	K^2
l	K^{l-1}

$K^{l-1} <= n$

```
while ( ) {
  i = i + k }
```

```
w( ) {
  i = i x K
}
```

$K^{l-1} <= n$

$l - 1 = \log_K n + 1$

$f(n) = 1 + 2(1 + \log_K n)$

$O(\log_K n)$

$\log_a C$

$\frac{a}{b} = 0$

```
while (n>0){
   n = n / 2;
}
```

l	n
1	n
2	$n/2$
3	$n/2^2$
4	$n/2^3$
l	$n/2^{l-1}$ = 0 = 1

$n = 2^{l-1}$

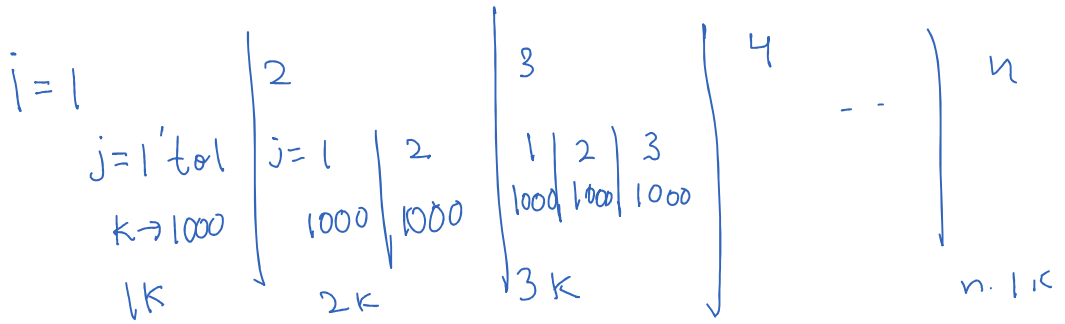
$\log_2 n = l$

```
for( i=0; i^2 <= n; i++){
   Syso(i);
}
```

$\sqrt{n} = n^{1/2}$

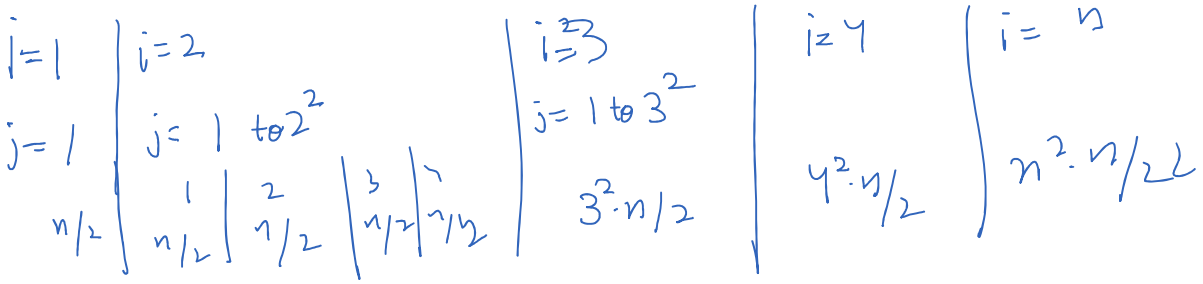
$n^{-1} = \frac{1}{n}$

$(\sqrt{n}+1) \cdot C_1 + C_2 = f(n)$



$1K(1+2+3+\cdots n) \Rightarrow 1000 \cdot \frac{n\cdot(n+1)}{2}$

$O(n^2)$



$n/2 + 4\cdot n/2 + 3^2\cdot n/2 \cdots + n^2\cdot n/2$

$n/2\,(1+2^2+3^2+\cdots n^2)$

$n/2 \cdot \frac{n\cdot(n+1)\,(2n+1)}{6}$

$O(n^3)$

$\left(1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}\cdots\frac{1}{n}\right)$

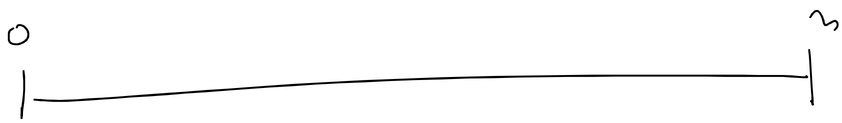
$\log n$

$i = 1, 2, 3, \ldots n$

$n + n/2 + n/3 \cdots n/n$

$$n\left(1+\frac{1}{2}+\frac{1}{3}+\frac{1}{4}+\cdots\frac{1}{n}\right)=n\log n$$

$$\int\frac{1}{x}$$



n/K

$O(1)$, $O(\log n)$, $O(\sqrt{n})$, $O(n)$, $O(n\log n)$

$O(n^2)$, $O(n^2\log n)$, $O(n^3)$, $O(c^n)$, $O(n!)$, $O(n^n)$

$c^n \leq n!$, n^n

$(c\cdot c\cdot c\cdots c)$

$(1\cdot 2\cdot 3\cdot c\cdot(c+1)\,(c+2)\cdots n)$

1) HW 4.4 H2

① Max Subarray Sum

Q Max Sub array sum

$n \le 10^5$

$O(n^3)$

$10^{15} / 10^9 = 10^6$

$O(n^2)$

$O(10^{10})$

$10^{10} / 10^9 \approx 10$

$O(n)$

10^5

	W	Best	
Bubble	$O(n^2)$	$\Omega(n^2) \to \Omega(n)$	
Selec	$O(n^2)$	$\Omega(n^2)$	
Ins	$O(n^2)$	$\Omega(n)$	

Given : 1 * x^n+2*x^(n-1)+3*x^(n-2)... + nx;
solve, return number
n=3;
x=7;

$1 \cdot (7)^3 + 2 \cdot (7)^2 + 3 \cdot 7^1$

Q → $1 \cdot x^n + 2 \cdot x^{n-1} + 3 \cdot x^{n-2} + 4 \cdot x^{n-3} + \dots + nx^1$

$n \to 3$, $x = 7$

$O(n)$

$7^3 + 2 \cdot 7^2 + 3 \cdot 7^1$

$343 + 2 \times 49 + 21$

98

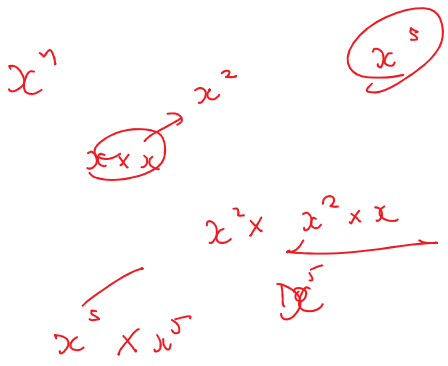
462

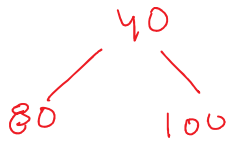
$n \to 2$
$x \to 1$

$1 \cdot (1)^2 + 2(1)$

$\Rightarrow$

x^5





(Key, Value) →

unique

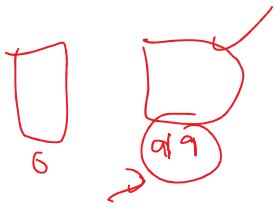
array → 0. n-1

A → 10
B → 20
C → 15
D → 10
E → 20

Key →	A	B	C	D	E
Value	10	20	15	10	20

Search → O(n), Key ∝ Value
Add → O(1)
Remove → O(n)
Update → O(n)

Has
Search → Key O(1), Update O(1)
Add → O(1)
Remove → O(1)




```
int[] arr1 = {30, 20, 40, 50, 70, 30, 20, 20, 50, 50 };
int[] arr2 = {30, 80, 30, 20, 20, 20, 90, 50, 20 };
```

30, 20, 50, 20, 20/50

int[] arr1 = { 30, 20, 40, 50, 70, 30, 20, 20, 50, 50 };

int[] arr2 = { 50, 80, 30, 20, 20, 20, 90, 50, 20 };

number of string concatination for string s to be a subsequence of string t

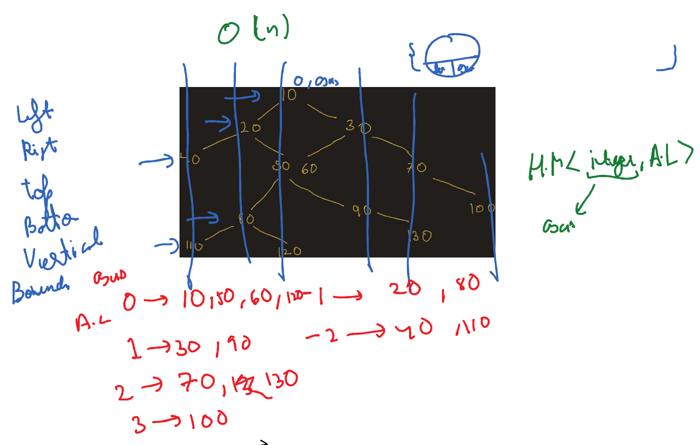
From
<https://docs.google.com/spreadsheets/d/14W2tmEZ1X2pr1e9kVV1cu74wBapshhURPyUBdKx0OQM/edit#gid=1635212751>

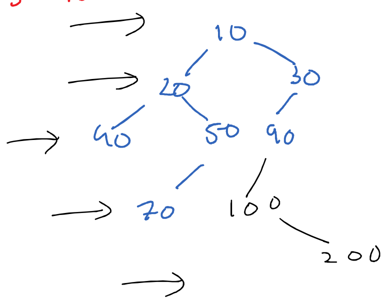
You are given an array of Integers in no particular order. Write a Program to find the longest possible sequence of consecutive numbers using the numbers from the array.

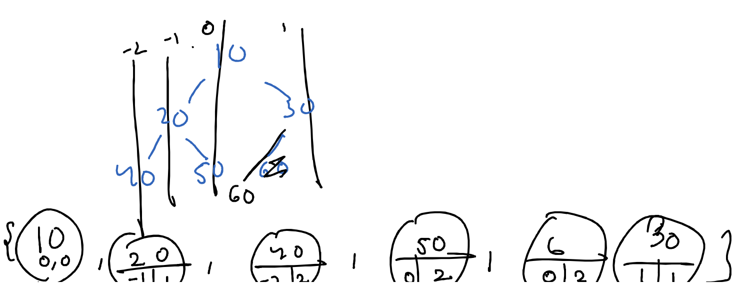
Target O(n)

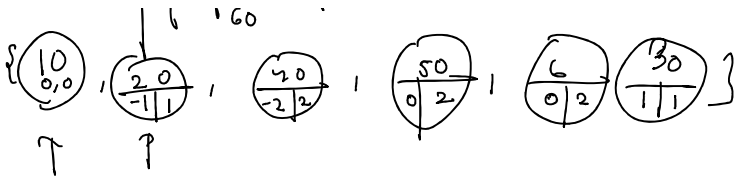
Input : [2,12,9,16,10,5,3,20,25,11,1,8,6]

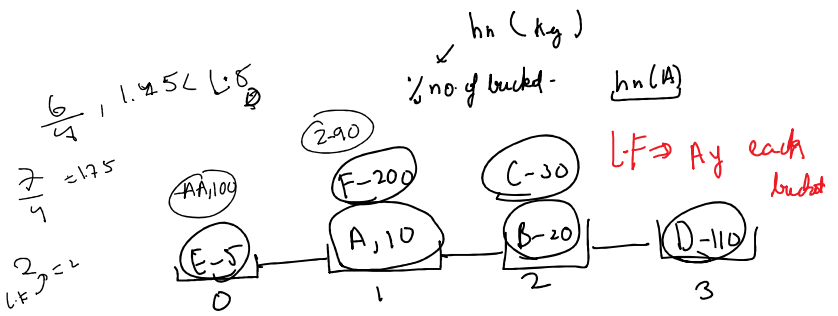
Output : [8,9,10,11,12]
Input : [15,13,23,21,19,11,16]
Output : [15,16]

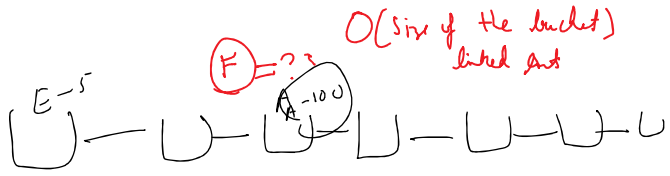












avg size of each bucket ← L.F ?? , collision resolve ?? → chain.

th ?? , default bucket } 16

Arrays ??