

Array

→ serial wise

Array is like a contiguous memory to variable
store 2 or 200.

Highest 10⁶ data can store in it.

Global variable & function 10⁸ address can store in it.

* 10 inputs

```
int a[5] = {1, 2, 3, 4, 5}
```

```
int i;
```

```
for (i=0; i<5; i++)  
{  
    printf("%d\n", a[i]);  
}
```

a[3] = {1, 2, 3}

$$1 + 2 + 3 = 6$$

————— X —————

Coaching Class-2

Thursday
28/11/19

Variable stores memory, no specific address given
 format specifiers
 $\%c, \%s, \%d \rightarrow$ format specifier

$a[s] = \{ 1, 2, 3, 4, 5 \}$

$\downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow$

(address) (address)

scan -> address access taken 2251 (012 & use 225 or 225 or)

datatype -> address memory subscript & (operator)

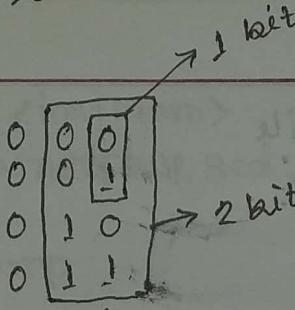
ANSWER 2251

Site: fresh2refresh.com → All the library functions are available here

data type	format specifier	size (byte)
int	$\%d$	4
float	$\%f$	4 (minimum)
double	$\%lf$	8
* long double	$\%Lf$	10
long long int	$\%lld$	8
char	$\%c$	1
Signed (can be both) Unsigned (positive)		

1 byte = 8 bit

→ 256



$$1 \text{ bit} = 2^1 = 2 \text{ characters}$$

$$2 \text{ bit} = 2^2 = 4$$

$$3 \text{ bit} = 2^3 = 8$$

$$2^4 = 16$$

$$2^5 = 32$$

$$1 \text{ byte} / 8 \text{ bit} = 2^8 = 256 \rightarrow \text{characters}$$

256 ASCII values

ASCII value of:

0 — a
(48) (57)

Difference 32

A — z
(65) (90)

a — A
(97) (122)

Ctrl + Shift + N \rightarrow New file

Ctrl + Shift + S \rightarrow save

Type Casting

(i) Implicit (Automatic) [Computer forces data type change]
[प्रोसेसर, कंप्यूटर फोर्स डेटा टाइप चेंज]

(ii) Explicit (Warning about type cast)

double a=5.00, b=2.00, c;

c = a/b;

not,

int a=5, b=2; (रज) (8N)
double c=a/b;

* It happens due to the size.

toph.com \rightarrow NGIPC \rightarrow cut piece (f. no. prob)

URI - 1040, 1021

first, start, first, second

----- X -----

Coaching Class - 3

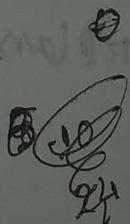
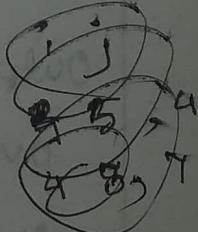
Thursday
5/12/12

00
24
600
8 00 16
00 20

- * Format specifier correct or not (in loop) छोड़े नहीं,
- * If different type of data type input from individually format specifier check space or not error,

```
for (i=0; i<=5; i++)  
{  
    i++;  
    for (j=1; j<5; j=j+3)  
    {  
        j++;  
        printf("%d\n");  
        j--;  
    }  
}
```

Q10



```

for (i=3; i<=10; i+=3)
{
    for (j=5; j<=10; j+=2)
    {
        if (i==j) continue;
        printf(i*j);
    }
}

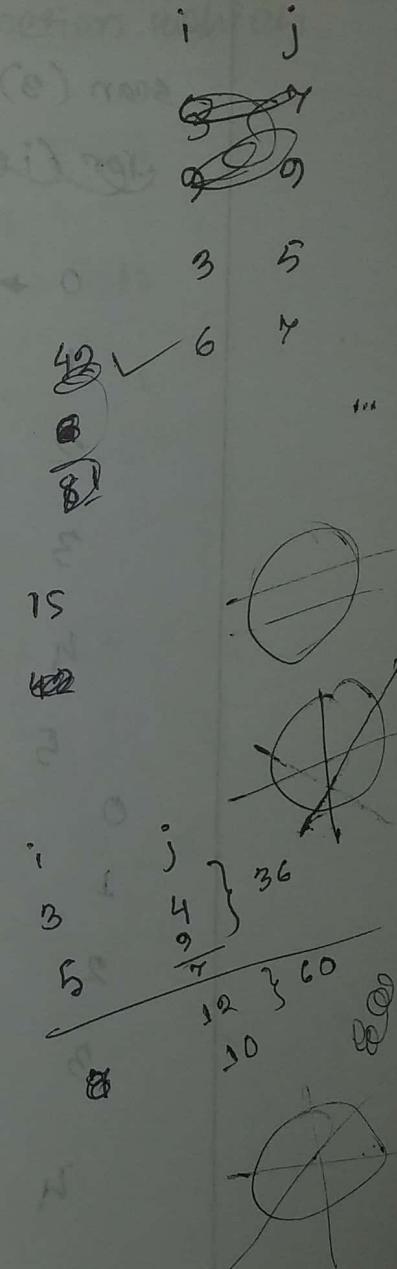
```

• At first solve inner loop

```

i=3;
while (i<=8)
{
    i+=2;
    j=4;
    while (j<=8)
    {
        j+=5;
        printf(i*j);
        j=j-2;
    }
}

```



scanf(3);

for (i=0; i<3; i++)

$$0 + \emptyset = 1$$

$$1 = 2$$

$$2 = 4$$

$$3 = 7$$

$$4 = 11$$

$$5$$

$$0 (0+1)$$

$$(0+ (2 \times 1)) = 0$$

$$(2 \times 2) = 4$$

$$(2 \times 2) = 6 + 1$$

$$(3 \times 2) = 8$$

$$(2 \times n) + 1 = 1$$

$$(2 \times n) = 2$$

$$(2 \times n) + 1 = 3$$

$$(2 \times n) + 1 = 4$$

$$(2 \times n) + 1 = 5$$

$$(2 \times n) + 1 = 6$$

$$(2 \times n) + 1 = 7$$

$$(n \times n) + 1$$

$$(2 \times 1) + 1$$

$$(2 \times 2) + 1$$

$$\frac{n(n+1)}{2} + 1$$

X

Coaching Class-4

Thursday
12/12/19

Password:
gmhs14th

URI 1047

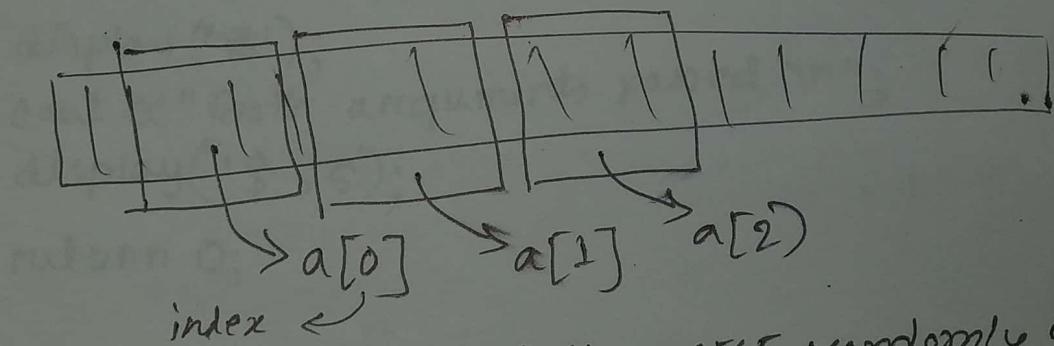
$$\begin{array}{r} 12 \quad 18 \\ 12 \quad 20 \end{array} \quad \left| \begin{array}{r} 20 \quad 20 \\ 8 \quad 15 \end{array} \right.$$

- * Start with minutes
- * Use "carry" as an extra variable

Array

Array \rightarrow sequential data structure

Ex: $a[\overset{\text{size}}{5}] = \{1, 2, 3, 4, 5\}$



* Access array serially as if, randomly or,

* $10^6 \rightarrow$ Highest size. (As local variable)

* $10^{7/8} \rightarrow$.. " (" Global ")

#Swapping:

Usage of temp variable:

* So that ~~the~~ the values don't get shuffled.

→ temp ~~will~~ initialize ~~and~~ ~~not~~ ~~2nd~~

$y = a[i + 1];$

$x = y;$ ~~only~~ ~~then~~ ~~values~~ ~~function~~ ~~2nd~~

$a[i] = x;$

Q. Two least smallest numbers:

int $a[7] = \{ 4, 6, 1, 4, 10, 14, 66 \}$

int $x = a[0];$

int $y = a[0];$

int $i;$

for ($i = 0; i < 7; i++$)

{

 if ($x > a[i]$)

{

$y = x;$

$x = a[i];$

}

 else if ($y > a[i]$)

{

$y = a[i];$

}

}

printf ("%d %d", x, y)

$a = \{1, 5, 7, 3, 9, 1, 3, 4, 7, 5\}$
~~for (i=0; i<100000; i++)~~

count ←
b[100]; a[100000];
for (i=0; i<100000; i++) b[i] = 0;
for (i=0; i<1000000; i++)
{
 b[a[i]]++;
}

6 5
0 3

* largest ascending subset

$a = \{1, 5, 7, 3, 4, 6, 9, 2, 7, 3, 6\}$

x = 0 3

y = 1;

for (i=1; i<11; i++)

{ if (a[i] < a[i-1])

(if a and first step) → body exec

{ if (y > x)

x = y;

y = 1;

}

else

y++;

}

if (y > x)

x = y;

print(x)

Coaching Class - 5

Fridays

13/12/19

String

Maximum hamming distance: \rightarrow Greeks for Greeks

Two same size वाक्य arrays, at same index वाक्य element
वर्णों की अंतरें अलग।

Example:

$$a = 2 \begin{pmatrix} 4 \\ 6 \end{pmatrix} 2 \begin{pmatrix} 4 \\ 3 \end{pmatrix} 7$$

$$b = 2 \begin{pmatrix} 6 \\ 4 \end{pmatrix} 2 \begin{pmatrix} 3 \\ 4 \end{pmatrix} 7$$

Ans: 3

Program:

```
a[n], b[n*2];
scanf("%");
for(i=0; i<n; i++)
{
    scanf(a[i]);
    b[i] = a[i];
    b[n+i] = a[i];
}
x = 0; maximum hamming  
distance
for(i=0; i<n; i++)
{
    c = 0; individual hamming distance
    for(j=i, K=0; j<i+n; j++, K++)
        if(b[j] != a[K])
            c++;
    if(c > x) x = c;
}
```

- डिस्ट्रिमन्ड अद्यता
- variable @ zero
- अपनामा होता है
- compare दोस्ती

- variable की
- अद्यता variable
- हाथ value
- input दोस्ती
- दोस्ती

* We can use this instead of gets() :

```
scanf("%[^\\n]%", *C", a);
```

* length of string in C, K = strlen(a);

" " " C++, K = a.size();

to find strlen:

```
i = 0;
while (1)
{
    if (a[i] == '\n')
    {
        j++;
    }
    else
    {
        break;
    }
    i++;
}
```

```
for (i=0; ; i++)
```

```
{
    if (a[i] != '\0')
        continue;
    else
        break;
```

```
print (del i)
```

* C to string copy str(a, b)
a++ a = a+b

* strcmp (a, b)

In
C++

```
if a > b    output → 1  
.. a < b    .. → -1  
.. a = b    .. → 0
```

* Palindrome:

```
for (i = L-1, k=0; i ≥ k; k++, i--)  
{  
    if (a[i] != a[k])  
    {  
        print (not palindrome);  
        break;  
    }  
    else  
    {  
        print (palindrome);  
    }  
}
```

UVa : 10945 (Mother Bear)

Solution:

My task:
To know more about
strcmp.

```
while (i){  
    get a;  
    if (strcmp(a, "DONE") == 0)  
        break;  
}
```

* To find ASCII value:

```
int K;  
char c = 'd';  
K = c - 'a';  
c = K + 'A';  
  
char a[10000], b[10000];  
K = 0;  
for (i = 0; i < strlen(a); i++)  
{  
    if (a[i] >= 65 && a[i] <= 90) {  
        a[i] += 32;  
        b[K] = a[i];  
    }  
    K++;  
}  
else if (a[i] >= 97 && a[i] <= 122) {  
    b[K] = a[i];  
}  
else {  
    continue;  
}
```

Coaching Class-6

Thursday
02/01/20

* Determine the winner - Hackerrank

①

$n \mod k = 0$ → Winner

If Even $k=1$, $n = \text{odd} \rightarrow A$
 $n = \text{even} \rightarrow B$

• $k=2$, $n = \text{odd} \rightarrow A$
next

K

$n \% (k+1)$
 $((k+1) \% k == 0)$

→ B

else

→ A

* Codeforces → CME

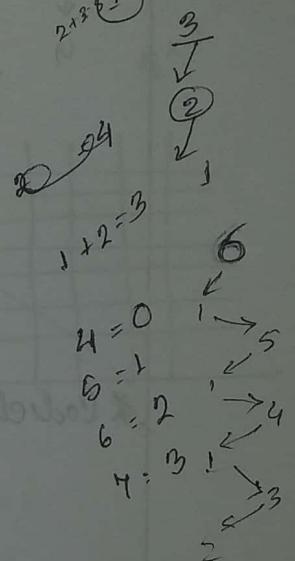
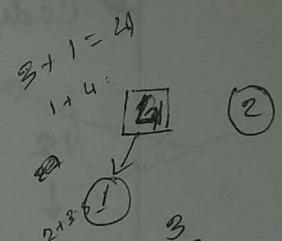
If $n < 4$

else if $n > 4$ & odd

→ 0

else if $n > 4$ & even

→ 1



* Codeforces → Chips Moving

12	3	5	15	8	4	8
↓	↓	↓	↓	↓	↓	
4	3	3	3	4	4	$\cancel{0}$

$\underbrace{\hspace{10em}}$
 $1+1+1 = 3$
 $= 3$

1 2, 3, 4
↓ ↓
1 2

2 3 → 1
3 3 → 0
0

$n = \text{odd} > \text{odd}$
 $\text{add} = \text{even}$
 $n = \text{odd} > \text{even}$
 $\text{odd} = \text{even}$

3, 4
↓
+1 - 3
3, 2

```
for (i=0; i<=10^5; i++)  
    b[i] = 0
```

2
↓ -2 0
0 1 1

* Frequency (How to find frequency)

1 % 2 → 1 } 1
1 % 3 = 2 } 1

```
> Solution
if (x>8) { continue; }
if (y>b[x])
{
    b[x] = y;
}
```

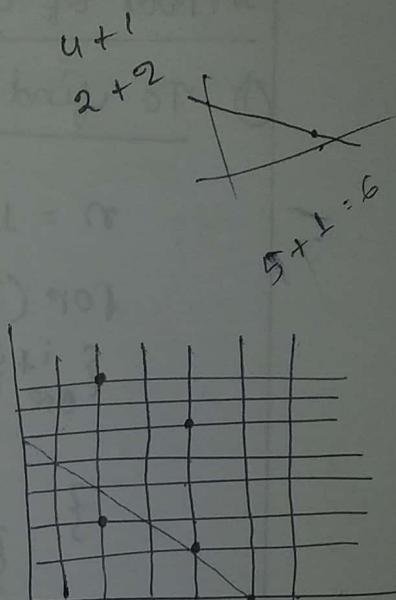
→ How to find maximum

27

* bedforces concave points:

* UVA : I love pizza

This is a frequency related problem



*

: (m) type = > i *

is (P = i) \Rightarrow (r = > i)

pr * all * not equal

$m = > 1 \times i$ यहां से कौन सी

$001 = > 1 \times 0$ $m = > 1 \times i$
 $01 = > 1 \times 1$ $(m) + 1 = > i$

Proof of contradiction :-

① To find the divisors of n :

$n = 10^{12}$

for ($i=1$; $i <= 10^6$; $i++$)

{ if ($i*i == n$) print(i)

{ else if ($n \% i == 0$) {

 print(i);

 print(n/i);

}

}

This is $\sqrt{10^{12}}$
or $\text{sqrt}(n)$

* $i <= \text{sqrt}(n);$
can be written as $i*i <= n;$

* for loop-2 loop भिन्नता वाले 10^7 तक

② Prime number यद्यपि दोनों हासिली समीकरणों की समीकरणें समान हैं।

प्रमाणित करना कि $i*i <= n$

$$\begin{array}{l|l} i*i <= n & a^2 <= 100 \\ i <= \text{sqrt}(n) & a <= 10 \end{array}$$

1. C/C++ 10⁶ മുമ്പ് എല്ലാ സംഖ്യകൾ ഒരു പേരിൽ

for (j=2 ; j<= 10⁶; j++)

{

 for (i=2; i*i <= j; i++)

{

 if (j % i == 0)

{

 flag = 1;

 break;

}

}

 if (p == 1)

{

 print (j = not prime)

}

else

{

 print (j = prime)

}

→ इसका
Sieve of Eratosthenes

Sieve of Eratosthenes : Wikipedia

int a[121];

for (i=2; i<=120; i++)

{

a[i] = 1; (1 means prime)

}

for (i=2; i*i<=120; i++)

{

if (a[i] == 1)

{

for (j = ~~i*i~~; j <= 120; j = j+i)

{

a[j] = 0; ($0 \text{ means not prime}$)

}

}

}

for (i=2; i<= n; i++)

if (a[i] == 1)

{ print (i is prime);

}

else

{ print (i is not prime);

}

→ So that the main
numbers doesn't
get cancelled out

Defn

G.C.D

$$\begin{array}{r} 2 \mid 12, 30 \\ 3 \mid 6, 18 \\ 2 \mid 2, 6 \\ \hline 2 \mid 12 \end{array}$$

$$\begin{array}{r} 2 \mid 12 \\ 2 \mid 6 \\ \hline 3 \end{array}$$

$$\begin{array}{r} 2 \mid 30 \\ 3 \mid 15 \\ \hline 5 \end{array}$$

$$\begin{array}{r} 12) 30(2 \\ 24 \\ \hline 6) 12(2 \\ 12 \\ \hline x \end{array}$$

$$\begin{array}{r} 10) 11(1 \\ 10 \\ \hline 1) 10(10 \\ 10 \\ \hline x \end{array}$$

55, 10, 5

$$\begin{array}{r} 55) 105 \mid \\ \overline{55}) 55(1 \\ 50 \\ \hline 5) 50(50 \\ 50 \\ \hline x \end{array}$$

G.C.D

$$\begin{array}{r} 3) 11(3 \\ 9 \\ \hline 2) 31(1 \\ 2 \\ \hline 1) 2(2 \\ 2 \\ \hline x \end{array}$$

51911

$$\begin{array}{r} 5 \\ 4) 15(1 \\ 4 \\ \hline 1 \end{array}$$

To find GCD:

```
int a, b;
if (a % b == 0)
    print(b);
```

t = b;

else

b = a % b;

a = t;

Code:

```
scanf ("%d %d", &a, &b);
while (a % b != 0)
{
    t = b;
    b = a % b;
    a = t;
}
printf ("%d\n", b);
```

L.C.M : $(a \times b) / \text{GCD of } (a, b)$

*** LCM, GCD, prime, prime factors \rightarrow imp. for contest

For prime factors:

```
for (i=2; i*i <= n; i++)
{
    while (n % i == 0 && n > 1)
    {
        printf ("%d", i);
        n = n / i;
    }
}
```

— X —

$$\begin{array}{r} 2 \\ 3 \\ \hline 12, 30 \\ 6, 15 \\ \hline 2, 5 \end{array}$$

$$\begin{array}{r} 2 \\ 2 \\ 3 \\ \hline 12, 30 \\ 4, 10 \\ \hline 2, 5 \end{array}$$

$$\begin{array}{r} 2 \\ 3 \\ 2 \\ \hline 6, 32 \\ 16, 40 \\ 2, 20 \\ \hline 1, 10 \end{array}$$

$$\begin{array}{r} 5 \\ \hline 5, 30 \\ 1, 6 \end{array}$$

$$\begin{array}{r} 2 \\ 2 \\ 2 \\ \hline 8, 12 \\ 4, 6 \\ 2, 3 \\ \hline 1, 12 \\ 2, 6 \\ 3 \end{array}$$

$$\begin{array}{r} 2 \\ 2 \\ \hline 12 \\ 6 \\ 3 \end{array}$$

Linear & Binary Search

Apply binary search array to search for an element's position (PPTD),

$\text{flag}_1 = 0, \text{flag}_2 = 0$

if array element $\text{flag}_1 = 1, \text{flag}_2 = 1$

if ($\text{flag}_1 == 1 \text{ && } \text{flag}_2 == 1$)

{

print (not found)

}

Codeforces: Choose two numbers

Binary Search

Array size = 10^6

elements value = 10^{18}

Test case / Question = 10^5

* This process is so fun! Try these out every time.

```

for(i=0; i<=10^6; i++) { b[i]=0 } b is given array.
for(i=0; i<n; i++)
{ cin >> a[i];
  b[a[i]] = 1;
}

for(i=0; i<q; i++)
{
  cin >> x;
  if(b[x] == 1)
  {
    cout << "found";
  }
  else
  {
    cout << "not found";
  }
}

```

* Avoid nested loop to avoid TLE.

* lightOJ Calm Down

* Consideration upto 10^6 index.

444

Binary Search (Book)

Example:

* Element to find must sorted 200 205, will sorted
for linear search.

A[] =	1	3	7	9	13	15	17
index	0	1	2	3	4	5	6

$$\text{mid index } 0+8 = 8/2 = \underline{\underline{4}} = 3$$

$$A[3] = 9$$

find 14?

if $14 < A[3]$

{ last \Rightarrow not change 204

repeat

}

else $14 > A[3]$

{

if \Rightarrow not change 205

repeat

}

return 2020 - error 205

106 1

103 2

1500 3

750 4

375 5

187 6

93 7

46 8

23 9

11 10

" "

loop 13 2025 100p 60000 - 2041

Code:

```
long long int l, r, m;  
int a=0, r=n-1;  
while (l <= r)
```

{

 m = (l + r)/2;

 if (a[m] == k)

 { print ("K is found");

 }

 else if (a[m] < k)

 {

 l = m + 1;

 }

 else

 { r = m - 1;

}

}

for (n=0, n<m,

* 2/3 sorted or 2/3rd element exist sort will take O(n),
then Binary search apply O(log n) time,

* Imp functions: upper_bound()
to find index lower_bound()

value for \rightarrow ^{last} index \rightarrow search
the first index for
 \rightarrow value \rightarrow ^{first} index \rightarrow search

* light0] 1088

$l = 0, r = n - 1, h = 0 - 1$

while ($l <= r$)

{

$m = (l+r)/2;$

if ($a[m] >= k$ && ($m == 0$ || $a[m-1] <= k$))

~~else~~ $n = m;$ break;

else if ($a[m] < k$)

$l = m + 1;$

else

$r = m - 1;$

}

if ($n == 0$)

$h = m;$

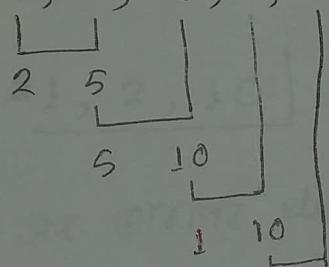
Coaching Class-9

Fridays
10/01/20

lightoj : Calm Down 1072

Bubble Sort

$$a[] = \{ 5, 2, 10, 1, 15, 12, 13, 12 \}$$



2 by 2 by 2 compare sort 8C4

Code: `for(i=0; i<n; i++)`

{

`for(j=0; j<n-i-1; j++)`

{

`if(a[j] > a[j+1])`

{

`t = a[j];`

`a[j] = a[j+1];`

`a[j+1] = t;`

}

}

Selection Sort

$a[] = \{ 2, 10, 1, 5, 3, 7, 2, 0 \}$

Search for the smallest element, at position 2 & value for swap 2nd 2nd.

Code:

```
for(i=0; i < n-1; i++)  
{  
    k = a[i];  
    n = i;  
    for(j=i+1; j < n; j++)  
    {  
        if(a[j] < k)  
        {  
            k = a[j];  
            n = j;  
        }  
    }  
    a[n] = a[i];  
    a[i] = k;  
}
```

Insertion Sort

$$a[] = \{2, 10, 1, 5, 3, 7, 2, 0\}$$

ଆମେ 2-ରୁ ଅଗ୍ରତା ଏଣ୍ଟମ୍ କିମ୍ବା sorted କିମ୍ବା କିମ୍ବା check କରିବା
 ଆମେ 10-ରୁ ଅଗ୍ରତା 1-ରୁ ଅଗ୍ରତା " " " " " " " " " "
 ଆମେ 1-ରୁ " " " " " " " " " "
 ଆମେ 5-ରୁ 10-ରୁ sorted କିମ୍ବା, ତାହା ଏଣ୍ଟମ୍ କିମ୍ବା sorted କିମ୍ବା

$$a[] = \{\underline{2}, 1, 10, 5, 3, 7, 2, 0\}$$

ଆମେ 5-ରୁ ଅଗ୍ରତା ଏଣ୍ଟମ୍ କିମ୍ବା sorted କିମ୍ବା check କରିବା,
 ଆମେ 5-ରୁ sorted କିମ୍ବା, ତାହା 5-ରୁ sorted କିମ୍ବା

$$a[] = \{1, 2, 5, \underline{10}, 3, 7, 2, 0\}$$

Continue କରିବା ଏହି ସମ୍ପଦ.

```
for(i=0; i<n-1; i++)
{
    j=i;
    while(j>0 && a[j-1] > a[j])
    {
        t=a[j];
        a[j]=a[j-1];
        a[j-1]=t;
        j--;
    }
}
```

Algorithm

}



2nd (Coaching Class - 10)
2D Dimentional Array

Thursday
16/01/20

```
sum = 0
for (i=0; i<3; i++)
{
    for(j=0; j<3; j++)
    {
        printf("%d\n", a[i][j]);
        sum = a[i][j] + sum;
    }
    printf("\n", sum);
    sum = 0;
}
```

HackerRank: 2D Array DS

Compare process for greatest value:
if ($s > m$) → smallest integer (MIN-INT)
 $m = s;$

Compare process for smallest value:
if ($s < m$) → greatest integer (MAX-INT)
 $m = s;$

0.00
111
222

0
0
0

1
1
2

2
2

Codeforces: Plus from Picture

0 0 0
0 1 1
~~a[i-1][j]~~ a[i-1][j] a[i-1][j]
a[i][j-1] a[i][j] a[i][j+1]
a[i][j-1]
print (Yes)

different find the total number of '#'
else
print (No)

```
for(k=i-1; k>=0; k--)  
{ if(a[k][j] == '#')  
    { s++; s++;  
    } else break;
```

Codeforces: Tiling Challenge

Toph Maze Ball Game

Coaching Class-11

Thursday
23/01/20

Number System

* Negative modulus $-8 \div 3$?

$$\Rightarrow 3 \times (-3) = -9 \quad (\text{because } -9 > -8)$$

$$\begin{array}{r} \overset{-3}{\overline{) -8}} \\ +9 \\ \hline -1 \end{array}$$

Decimal $\left[\begin{array}{l} \rightarrow \text{Binary} \\ \rightarrow \text{Octal} \\ \rightarrow \text{Hexa-decimal} \end{array} \right]$ Just base theor
divide 84(10) 2(4),

Binary
Octal
Hexa \rightarrow Decimal }

$$1^3 2^2 2^1 8^0 2^1 8^{-2} 2^{-3} \\ \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \quad \downarrow \\ 2 \times 8^2 \quad 8 \times 8^1 \quad 8 \times 8^{-2} \quad 2 \times 8^{-3} \\ 128 \times 8^3 \quad 2 \times 8^1 \quad 2 \times 8^{-1}$$

* Always converted to decimal in problem solving

Process :-

11-20-19 (continued)

* How to find digits from a number:-

$$t = 1234$$

while ($t \geq 0$)

{

$$a[i] = t \% 10;$$

$$t = t / 10;$$

$i++$;

}

① $123 \rightarrow t$

$$\begin{array}{r} 123 \\ 10 \sqrt{ } \\ 123 \\ -120 \\ \hline 3 \end{array}$$

④ → digit

② $12 \rightarrow t$

$$\begin{array}{r} 12 \\ 10 \sqrt{ } \\ 12 \\ -10 \\ \hline 2 \end{array}$$

③ → digit

③ $1 \rightarrow t$

$$\begin{array}{r} 1 \\ 10 \sqrt{ } \\ 1 \end{array}$$

② → digit

④ $0 \rightarrow t$

$$\begin{array}{r} 0 \\ 10 \sqrt{ } \\ 0 \end{array}$$

① → digit

* Decimal → Octal ?

~~123~~

$$t = 24$$

$$a[i] = t \% 8;$$

$$t = \cancel{\cancel{0}} \cancel{0} + t / 8;$$

$i++$;

$$\begin{array}{r} 24 \\ 8 \sqrt{ } \\ 3 \quad 0 \\ -0 \\ \hline 3 \end{array}$$

→ 30

$$\begin{array}{r} 24 \\ 8 \sqrt{ } \\ 24 \\ -24 \\ \hline 0 \end{array} \rightarrow 8$$

t/8

Same for any decimal no.

* digit more than 10^18 means string.

$$\begin{array}{r} 56 \\ -8 \\ \hline 48 \end{array}$$

② \rightarrow last digit even

③ \rightarrow sum of digit - sum of digits is 3 or 9 or divisible by 3

④ \rightarrow last digit 0 or 5.

⑤ \rightarrow even, 3 factor or not

⑥ \rightarrow sum, a factor or not

* for finding sum:

difference between character & int

$a = "123456789"$

for ($i=0; i < d; i++$) {

$s = s + (a[i] - 48);$

}

* To find the number from the string:

$s = 01234567890$

for ($i=0; i < 10; i++$) {

$s = s * 10;$

$s = s + a[i] - 48;$

} $s = s * 10;$

$s = a[0] - 48$

for ($i=1; i < 10; i++$) {

$s = s * 10;$

$s = s + a[i] - 48;$

}

Addition
&
Subtraction
of string

2 2 2 2 2 2 2 2

$$9+8+7+6+5+4+3+2+1$$

*Problem SPOJ: last digit (LAST DIGIT)

To find the last digit of a^b :

$n = \text{index} \Rightarrow$ of series of powers of a

$$\therefore [(b \% n) - 1] \rightarrow \text{index}$$

[2] [3]

2	3
4	9
8	7
6	1
2	3
4	9
8	7
6	1

for ($i=1; i<=200; i++$)

{

$s = s \times i;$

while ($s \% 10 == 0$) {

$t++;$

$s = s / 10;$

}

$s = s \% 10;$

$a[i] = t;$

}

Next : Oyler's Pie Function

X

Coaching Class-12

Thursday
30/01/20

ASCTI Take:

$$A = 65$$

$$a = 97$$

We can write

$$a[K] = \text{something} \dots ;$$

$$K++ ;$$

as,

$$a[K++] = \text{something} \dots ;$$

*Prove of contradiction (Part of number theory)

Types of variables:

- (1) local (inside the function only)
- (2) Global
- (3) Static

Static variable:

Syntax:

static datatype variable = constant;

Recursion

Condition for stoping is called

"Base Case"

Coaching Class - 13

Bitwise Operators

Thursday
6/02/20

Codechef: Xor Again

\sim → NOT Operator

$\&$ → AND Operator

$|$ → OR Operator

\wedge → XOR Operator

$>>$ → Right Shift

$<<$ → Left Shift

$$\begin{array}{r} 00000100 \\ 11111011 \\ + 1 \\ \hline 11111100 \end{array}$$

concept
of binary
representation

1 bit $\rightarrow 2^1$

8 bit or ~~8 bytes~~ 1 byte $\rightarrow 2^8$

NOT:

• NOT of 4 is -5. [$4+1=5$ and negative of 5 is -5]

NOT of -5 is 4 [$-5+1=-4$ and positive of -4 is 4]

Shift Operators:

Left shift operator moves all bits left.

Right shift operator moves all bits right.

$3 >> 1$

Ex: 0 0 0 0 0 0 1 1 \rightarrow after shifting 2 bits

$\swarrow \searrow \swarrow \searrow \swarrow \searrow \swarrow \searrow$

existing 5 bits are lost

new 5 bits are created

Ex: $15 >> 3$

00001111
00000001 ↗
convention =
Final =

0	0	0	0	1	1	1	1
				0	0	0	1
0	0	0	0	0	0	0	1

Ex: $3 << 2$

00000011
00001100 ↗

* left shift 2 bits : $3 << 2 = 12$ [$\begin{array}{l} 3 \times 2 = 6 \\ 6 \times 2 = 12 \end{array}$]

* Right shift 2 bits : $15 >> 3 = 1$ [$\begin{array}{l} 15 / 3 = 5 \\ 5 / 3 = 1 \end{array}$]

But 1 is 2^3 : $1 << 3 = 8$ then $2^3 = 8$

* 5 factors -6 ??
These are for positive and negative signs

- < [0] 0 0 0 0 1 0 1 ↗ 5
+ < [1] 1 1 1 0 1 0 ↗
0 0 0 0 0 1 | 0 → -6
→ 1 has same 2 bits, change 2nd 2 bits

* How to not get garbage value for bigger no.?
→ 1 << 5 & then 1 LL << 5
→ long long

AND: $100 \rightarrow 5$

$$\begin{array}{r} 101 \\ 100 \\ \hline 100 \end{array} \rightarrow 4$$

$100 \rightarrow 4$

$a = 4$

$b = 5$

$c = a \wedge b$

$c = a \wedge b$

$c = a \vee b$

OR: $100 \rightarrow 5$

$$\begin{array}{r} 101 \\ 101 \\ \hline 101 \end{array} \rightarrow 6$$

$101 \rightarrow 6$

XOR: $\text{odd} \oplus \text{even} \rightarrow 1$

$\text{even} \oplus \text{odd} \rightarrow 0$

$$\begin{array}{r} 100 \\ 101 \\ \hline 001 \end{array}$$

$$\begin{array}{r} 100 \\ 101 \\ \hline 001 \end{array}$$

$$\begin{array}{r} 001 \\ 100 \\ \hline 101 \end{array}$$

* How to find odd or even?

Is Y odd or even?

$$Y \& 1 \rightarrow \begin{array}{r} 111 \\ 001 \\ \hline 001 \end{array} \rightarrow 1$$

if ($Y \& 1 == 1$)

{ odd (2nd or last bit is 1 or 0 if odd)

} else

{ even }

000

$001 \rightarrow 1$

010

011

100

101

110

$111 \rightarrow Y$

1 2 3 4 5

1 3 6 10 15

Coaching Class-14

Saturday
08/02/20

* Pre-calculation technique

```
ios_base::sync_with_stdio(0);  
cin.tie(0);  
cout.tie(0);
```

To minimize time on Codeforces, for cin and cout

* XOR is important for bitwise operation

Codeforces: Make Good

* Find subset through arrays by bitwise operators

{a, b, c}

return value, if sum XOR of all elements is 0

value from arr,

Example: 3 XOR 0 → 3

8 4 2 1
↑ ↑ ↑ ↑
1 1 1 1
0 0 0 0 } zero
2 1 0 0 } 2100
count
214 at

a	b	c
↑	↑	↑
0	0	0
0	0	1
0	1	0
0	1	1
1	0	0
1	0	1
1	1	0
1	1	1

$\rightarrow \{0\}$
 $\rightarrow \{c\}$
 $\rightarrow \{b\}$
 $\rightarrow \{a, c\}$
 $\rightarrow \{a, b\}$
 $\rightarrow \{a, b, c\}$

Coaching Class - 15

Recursion

Mondays
10/02/20

* for blankspace in string:

scanf("%d\n");

visualgo.net

* GCD

$$6) 16(2$$

$$\frac{12}{4}$$

$$4) 6(1$$

$$\frac{4}{0}$$

$$\text{GCD} \leftarrow \overbrace{(2)}^{2}) 4 \quad (2) \cancel{\text{GCD}}$$

* GCD function in C++:

-- gcd();

* int gcd(int x, int y)

{

 if (x == 0) return y;

 if (y == 0) return x;

 return gcd(~~(x, y, y)~~,
 (x % y, y));

}

If $x = 6$
 $y = 16$

① 6) 16(2
 12
 —
 4

② $x = 4$
 $y = 6$

4) 6(1
 4
 —
 2

Coaching Class-16

Recursion

Wednesday
12/02/20

GCD:

```

if (x == 0)
{
    print(y);
}
else
{

```

$$y = x;$$

$$x = y \% x;$$

* Recursion is also called "stack function".

$$\textcircled{1} \quad 16$$

$$\textcircled{2} \quad 1 + (16/2)$$

$$1 + 8 = 9$$

$$\textcircled{3} \quad 1 + (9/2)$$

$$1 + 4 = 5$$

$$\textcircled{4} \quad 1 + (5/2)$$

$$1 + 2 = 3$$

$$\textcircled{5} \quad 1 + (3/2)$$

$$1 + 1 = 2$$

$$\textcircled{6} \quad 1 + (2/2)$$

$$= 1 + 1 = 2$$

$$\textcircled{7} \quad 1 + (2/2)$$

$$= 1 + 1 = 2$$

$$\textcircled{8} \quad 1 + (2/2)$$

$$= 1 + 1 = 2$$

$$\textcircled{2} \quad 16(8,$$

$$\frac{16}{0} \rightarrow 1$$

$$\textcircled{3} \quad 2(4$$

$$\frac{2}{0} \rightarrow 2$$

~~1 + 1 = 2~~

$$\textcircled{4} \quad 4(2$$

$$\frac{4}{0} \rightarrow 2$$

$$\textcircled{5} \quad 2(1$$

$$\frac{2}{0} \rightarrow 1$$

$$\textcircled{6} \quad 1(0$$

$$\frac{0}{1}$$

```

int calcu(int x)
{
    if(x == 1)
        return 0;
    return 1 + calcu(x/2);
}

int main()
{
    int a = 16;
    cout << calcu(a);
}

```

Solution:

$$1 + 8 + 4 + 2 + 1$$

$$= 16$$

return n & fact(n-1)

$$5 \times 4 \times 3 \times 2 \times 1$$

$$\left. \begin{array}{l} 5 \times 4 \times 3 \times 2 \times 1 \\ f(1) \\ f(2) \\ f(3) \\ f(4) \\ f(5) \end{array} \right\}$$

- ① C(1) $x=$
- ② C(2) $x=$
- ③ C(4) $x=$
- ④ C(8) $x=$
- ⑤ C(16) $x=$

* int ~~end~~^f (int a[], int n)

{

 int b;

 if (n == 1)

 { return 0; }

}

 else

 { b = f(a, n - 1); }

}

 if (b > a[n - 1])

 { return b; }

}

 else

 { return a[n - 1]; }

}

}

int main()

{

 int a[] = { 1, 2, 3, 4, 5 };

 pr(f(a, 5));

Output:

5

f(a, 1)

f(a, 2)

f(a, 3)

f(a, 4)

f(a, 5)

Thursday

13/02/20

Format specifier of address (%a) is %p.

a = 5

p = &a [a is address]

*p = a [a is value]

int main()

{

int a = 10;

int *p;

pr(a);

p = &a;

*p = 20;

pr(a);

a = 15;

pr(a);

pr(*p);

pr(&a);

pr(p);

pr(&p);

pr(a) → 10

pr(a) → 20

pr(a) → 15

pr(*p) → 2015

pr(%a) → address of a

pr(p) → &a

pr(&p) → address of p

(1-10) 1 = 10

(1-10) 2 = 20

(1-10) 3 = 15

(1-10) 4 = 2015

(1-10) 5 = address of a

(1-10) 6 = &a

(1-10) 7 = address of p

(1-10) 8 = &p

(1-10) 9 = address of &p

(1-10) 10 = &(&p)

(1-10) 11 = &(&(&p))

(1-10) 12 = &(&(&(&p)))

(1-10) 13 = &(&(&(&(&p))))

(1-10) 14 = &(&(&(&(&(&p)))))

(1-10) 15 = &(&(&(&(&(&(&p))))))

(1-10) 16 = &(&(&(&(&(&(&(&p)))))))

(1-10) 17 = &(&(&(&(&(&(&(&(&p))))))))

(1-10) 18 = &(&(&(&(&(&(&(&(&(&p))))))))

(1-10) 19 = &(&(&(&(&(&(&(&(&(&(&p))))))))

(1-10) 20 = &(&(&(&(&(&(&(&(&(&(&(&p))))))))

Scanned by CamScanner

int main()

{ int a = 10, b;

int *p;

pr(a); → 10

p = &a;

b = *p;

*p = 15;

pr(a); → 15

pr(b); → ~~10~~ 10 [Not 15, because we haven't changed the value again like $b = *p$]

pr(*p); → 15

pr(&a); → address of a

pr(&b); → " " b

pr(p); → " " a

}

int main()

{ int x = 10, y;

int *p, *q;

p = &x;

y = *p;

*p = 15;

q = 20;

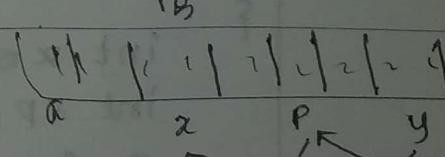
pr(x); → 15

pr(y); → 10

pr(*p); → 15

pr(*q); → 20

}



*q = NULL

This is called null pointer.

Double Pointer

```
int x = 10, *p, *q;
```

*p = x;

p = &x

*p = 10

q = &p

*q = p (which means p = &x)

**q = 10

*p

p = &q

**p = 10

```
int main()
```

{

```
    int x = 10, y = 5;
```

```
    int *p, *q, *r;
```

p = &x;

q = &p;

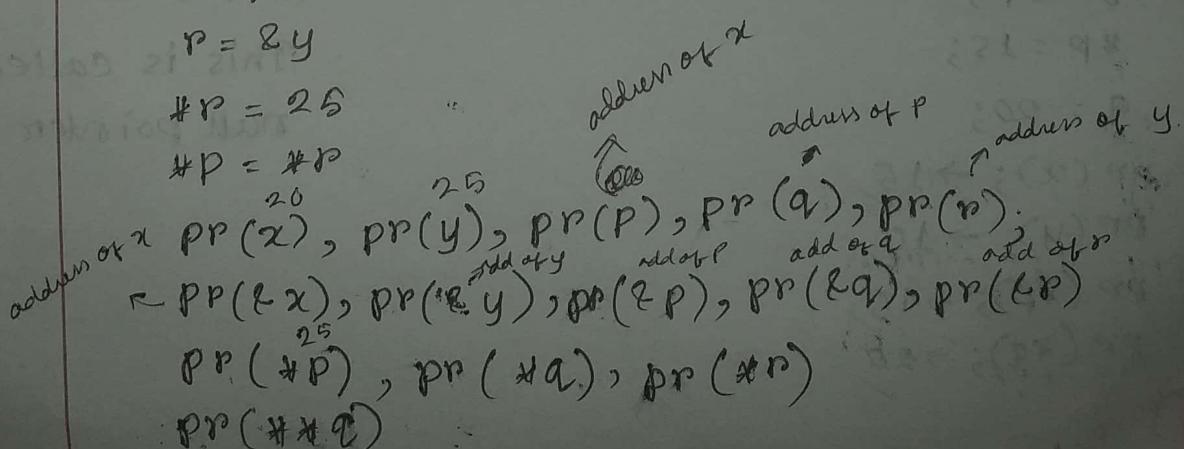
*p = 10

*x = 20

p = &y

*p = 25

*p = *p



}

* sort function in C++:

sort (a, a+n)

↓ ↓
अवधि तक (पर्याम) संख्या का विनाश

Pointers in array

int a[] = {^{a+0} 1, ^{a+1} 2, ^{a+2} 3, ^{a+3} 4, ^{a+4} 5}

P = &a[0]

= (a+0)

= a

① void add (int x, int y)

{
 x = 25, y = 35;
}

int main()

int a = 10, b = 15;

add (a, b);

pr (a, b);

}

② void add (int *x, int *y)

{
 x = 25, y = 35;
}

int main()

int a = 10, b = 15;

add (&a, &b);

pr (a, b);

}

→ Output: a = 10
 b = 15

जबकि function-में परिवर्तन
value return करते हैं

Output: a = 25
 b = 35

→ परिवर्तन value return करते हैं
जबकि function में b, address change
होता है तो value change होता है।

* I was absent in the previous three classes.

* এখনোন্নয় সম্পর্ক
Blog

* In C++, to take input for string:

`getline(cin, s);` [for full line including space]

`cin >> s;` [excluding space]

* Difference ~~between~~ among among `getline`, `scanf()`, `cin` in C++.

(i) `getline()` → Full line input ফর্মেট (including space) একটি line এ। কিন্তু ২০১৫।

(ii) `scanf()` → One word input ফর্মেট একটি line
এ। কিন্তু ২০১৫।

(iii) `cin` → One word input ফর্মেট একটি
কিন্তু variable-এর input ফর্মেট,

Standard Template Library

* `a.size()` → To find the length of the string.

* `swap(a, b)` → For swapping two strings.

* We can compare strings as well:

```
string a, b;  
a = "abb";
```

```
b = "bab";
```

```
if (a > b)
```

```
else if (b > a)
```

```
else
```

* How to sort an array:

`sort(a, a + n)` [here, n is the size of a]

* How to sort a string data type in

`sort(a, a + n)` → array ~~char~~ ~~int~~ ~~float~~ ~~double~~

~~a = "abbddeccaa"~~

→ sort by letters ~~or~~ sort by ASCII

`sort(a, a + 3)` → array ~~char~~ ~~int~~ ~~float~~ ~~double~~

* How to sort a string:

`sort(a.begin(), a.end())`

→ `sort(a.begin(), a.begin() + 3)`

- * reverse (a.begin(), a.end())
- * a.empty() → It checks whether the string is empty or not.
- * cout << a[s] → to print the sth character of the string
or a.at(s)

Vector STL

- * How to declare it:

```
vector<int / double / char / string> a;
```

```
a.pushback(1);
```

```
a.pushback(2);
```

size of vector : a.size();

```
vector<int> a(n);
```

2D vector

```
vector<vector<int>> v;
```

space

It takes $\log(n)$ time for each sorting.
Here, n is the index.

Ex:

map["Karim"] = 25

map["Rahim"] = 22

map["Abir"] = 27

After sorting, map["Abir"] will be the first element,
because "A" comes first. Then Karim and then Rahim.

#include <map>

How to declare it?

⇒ map<string, int> mp;

ex: mp["Hello"] = 25;

For unordered map:

unordered_map<string, int> mp;

mp.begin()

mp.end()

mp.size()

mp.clear()

mp.erase() Ex: mp.erase("Abir")

mp.insert(make_pair("Rahman", 50));

User input:

```
scanf("y.s", s);
scanf("%d", &x);
mp[s] = x;
```

$\{ \text{if } (\text{mp.find("Rahman"}) == \text{mp.end}()) \rightarrow$

{

pr(The key is not present);

}

Output:

* Love Extends Code } → YouTube

* HackerEarth → STL → problem

Problem:

একটি array এর কোনো element কানেক্স করা কৈ ?

a = { 10, 4, 10⁶, 25, 35, 0};

b[10⁶+1];

for(i=0; i<=10⁶; i++)

{ b[i] = 0;

}

→

a = { 10, 3, 4, 2, 10, 4, 5, 3, 5, 0 }

b[10];

for (i = 0; i < 10; i++)

{

 b[i] = 0;

}

cin >> n;

for (i = 0; i < n; i++)

{

 cin >> a[i];

***** b[a[i]]++;

}

Solution by map:

unordered_map<ll, int> b;

for (unordered_map<

 b.max_load_factor(0.25);

 b.reserve(512);

TLE

in OIISRT GYM

Codeforces (Only here)

Social Network (Hard Version)

```
int main()
{
    int n, k, i, j, p, q, h;
    cin >> n >> k;
    unordered_map<int, int> b;
    b.max_load_factor(0.25);
    b.reserve(512);
    int a[n];
    queue<int> q;
    for (i = 0; i < n; i++)
    {
        cin >> a[i];
    }
    for (i = 0; i < n; i++)
    {
        if (b[a[i]] == b.end())
        {
            q.push(a[i]);
            b[a[i]] = 1;
        }
        if (q.size() > k)
        {
            x = q.front();
            q.pop();
            b.erase(x);
        }
    }
    pr(q.size());
    vector<int> v;
```

```
while (!q.empty())
{
    x = q.front();
    q.pop();
    v.push_back(x);
}

for (i = v.size() - 1; i >= 0; i--)
{
    pr(v[i]);
}

}
```

End of coaching classes

