8/10/23 Week -> 7 Date 26

Doubt class

Prime number :

(1) -> Naive

2 -> N-1

(2) -> optimize

2 -> $\leq \sqrt{N}$

Advance Method

(3) -> Sieve of Eathostherm

-> (1) -> N

↳ Square Primes

Optimize

↳ Inner Loop

-> Outer Loop

(4) -> Segmented Sieve

Spiral

8/10/23

Date 27

(ii) GCD

Euclidean's Algorithm

(1) -> gcd(a-b, b) a>b

else (b-a, a) b>a

=> (a%M * b%M)%M = (a*b)%M

(within M) * (within M)

↳ can be out of bound.

Ex: Why use Modulus ?

int a = 2^{11}

int b = 2^{13}

int c = a*b

$2^{11} * 2^{13} = 2^{24}$ => overflow

$M = 10^9$ (within)

c = ((a%M) * (b%M))%M

=> LCM * GCD = a*b

$$LCM = \frac{a*b}{HCF}$$

Spiral

8/10/23

Date 88

Exponential -> 0^b

Pointer ->

C++ -> Embedded Developed Language

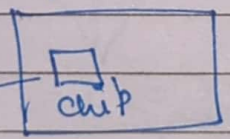
Memory

↓

Inner Level Control

System Software

Printer



SOC

(System on Chip)

↓

Isse chip pe Ram, ROM

(64MB) RAM

Isko under (Embedded software)

Memory full control

||

only C++

XOT -> Java Python

koi bhi Provide nhi karta

Spiral

Date 29

8/10/23

Python -> numpy -> array -> inner C++/C used
-> print()
-> opencv
-> very slow (Interpreted Language)

Same chip for Hardware & Software -> Embded chip

Pointer -> Saves address of other variable

* -> asterick

& -> mpercent

32 v/s 64 bit

pointer 4 8

32 bit

| 1 | 1 | 1 | 1 | 1 | 1 | 1 |

0 31

2^{32} address can be store in 4 Bit.

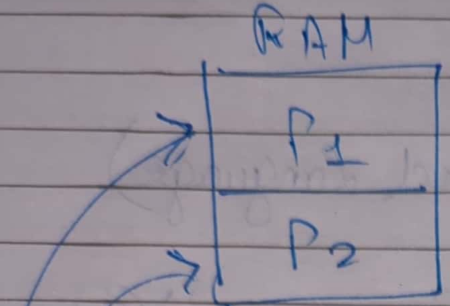
64 byte
-> 8 byte Pointer

Spiral

8/10/23 Date (90)

Wild pointer → int * ptr;

↳ Dangerous



→ chrome → int * ptr
↓
garbage value

Present here → No Problem

tou ptr
305
(Illegal memory access)

int a = *ptr

How to solve →

int * ptr = null;
int * ptr = 0;
int * ptr = nullptr

Impt {
int a=0;
if (ptr != 0)
a = *ptr;

Null pointer Deference → Null pointer Exception

Pointer Arthematics →

~~int b = 5~~
~~int *a = &b;~~
~~*a = *a + 1~~

a = 100
*ptr = &a
ptr + 1

a: [10] (104)

ptr: [104] + 1 = 108

8/10/23

Date 91

int a[5] = [1|2|3|4|5]

int * b = a; · *b -> 1

void * c = (void *)a;

||

can point to any Data;

*(b+1) -> 2

*(c+1) -> Error

void -> Data type hi Nhi pta

-> int a[5] = [1|2|3|4|5]

a = a+1

100 104 108

lost [1|2|3|4|5]

a

a

This is NOT Permitted

Not allowed

||

That's why => a = a+1

auto a = b;

Data Type dependent on b

auto a = (void *)b;

(void *)

Spiral

8/10/23 Date (92)

char * ptr = "Codehelp";
↳ Bad Practice but valid

ptr[5] = 'f'; Stack & Constant

↳ error/undefine/did not compile

ex ⇒ int a[5] = {1, 2, 3, 4, 5}

Pointer to an Integer → int * ptr = a; ✓

&a → ✗

Array of Pointers

int (* ptr)[5] = &a;

&a and a → Print → address
because print implement
hai else

```
int x = 10;
int * ptr = &x;
cout << * ptr + 2*(* ptr) - 3*(* ptr);
```

No error
10 + 2*10 - 3×10 = 0
20

Spiral

8/10/23 Date (93)

x = 5, y = 10, z = 15

int * arr[] = ~~{x, y, z}~~ {&x, &y, &z}

cout << * arr[1] << endl;

→ 10

Swap with pointers

x = 5, y = 10

swap(&x, &y)

void swap(int *x, int *y)
{
int temp = *x
*x = *y
*y = temp;
}

( ) -> Impt

Doubt -> int (*p)(char*) → function pointer

ex- int len(char* str){
return strlen(str)
}

main(){
char str[] = "codehelp";
cout << len(str
}

fP(str) ← (calling this)

OS call → address