

10/28

* Data Types

- Integers
- Floats
- Chars
- etc.

* Lab 4 is posted

* Assignment #4 is posted
(End of Lab #4)
due 11/7

NOTE:

For Thursday presentations:

1. Prepare no longer than 8-9 min presentation

2. Make sure Deepak has your handout (pdf) by wed. eve.

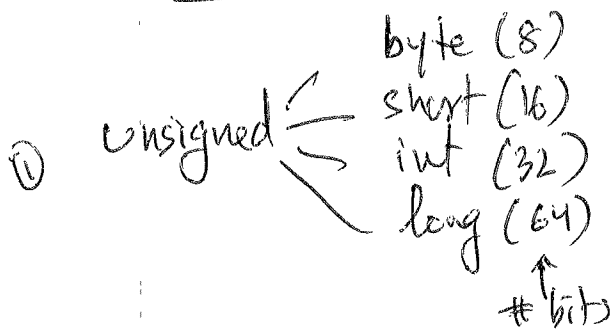
3. You are welcome to prepare a PPT presentation.

Bring your own laptop

OR send the presentation to Deepak by Wed. evening.

Integer Types

Names: C/C++/Java



Python

int

unlimited!

On computer, in Python try:

```

def fact(n):
    if n == 0:
        return 1
    else:
        return n * fact(n-1)

```

```

try:
    fact(5)
    fact(10)
    fact(50)
    fact(100)
    etc.

```

Floating Point Numbers

Names: C/C++/Java

float (32)
double (64)

#bits used

Python

float

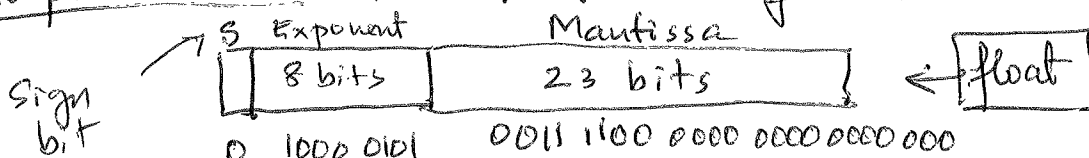
implemented using C float | what if you need higher precision??

Go

float32
float64

(2)

Representation: IEEE 754 Floating Point Standard



= 1.001111 * 2^6 = (1.001111)_2 = (79.0)_10

(1)

Most computers use Floating Point Processing Units (FPU's) to do floating point arithmetic.
Also called GPU's

Precision Issues: Cannot represent 1/3 exactly!

Try in Python: $1.0/3 = 0.3333333333333333$
16 digits

try: ~~0.1 + 0.1 + 0.1~~ 0.1 + 0.1 + 0.1
a = 0.30000000000000004
15 zeros

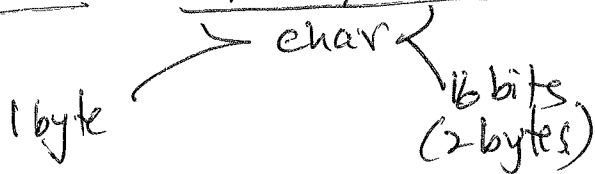
try: 1.1 + 2.2 ≠ 3.3!!
result in: 3.3000000000000003

try: 0.1 + 0.1 + 0.1 - 0.3
→ 5.55...e-17

X

Characters

Name: C/C++/Java



Go/Python

No char type
chars are represented
as strings with 1 char
e.g. "a", "A", etc

②

Representation:

basic: 7-bit ASCII code for English.

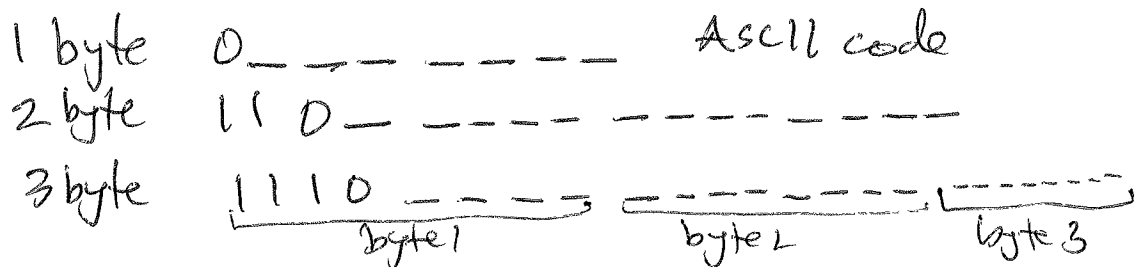
other character coding standards: ISO 10646
uses 16-bits, encodes all characters of known
human languages (including math + other symbols
~~and emojis~~)
~ 1,112,064 characters!

Unicode: expanded version of ISO 10646 (includes emojis)

UTF-8: provides backward compatibility with ASCII

encoding can be 1-6 bytes/char
uses 3 bytes for all standard ISO chars.

①



In C, char and int can be used interchangeably.

e.g. $\frac{'A'}{\text{code } 65} + 3 \Rightarrow \frac{'D'}{68}$

operations: +, -

Boolean Types

Names: C
doesn't have
any boolean type
but provides
<stdbool.h>
bool found;

Java
boolean $\left\{ \begin{array}{l} \text{false} \\ \text{true} \end{array} \right.$

Python
bool $\left\{ \begin{array}{l} \text{False} \\ \text{True} \end{array} \right.$

(2)

Operations: $\frac{1}{0}$ ~~and~~, $\&\&$, $\frac{11}{or}$
not and or

and, or, not

Complex Type

Name: C not available C++ complex
|
but has a
Complex library.

Python
complex
a = complex(2, 3)
print(a) $\rightarrow (2+3j)$

(1)

or do Floating Pt exp. example

Enumeration Types

- Enable users to define their type name and literals.

e.g. Pascal

type weekday = (Sunday, Monday, --, Saturday);

var day : weekday;

operations: succ(Monday) → Tuesday
pred(Monday) → Sunday

- can write loops:

for today := Sunday to Saturday do
begin
end;

- can be used as array indices

var attendance : array[weekday] of integer;
attendance[Sunday] := 0;

C has enumerated types

enum weekday { Sunday, Monday, --, Saturday };

weekday day = Sunday;

for (int i = Sunday; i <= Saturday; i++) {

↑
enums are like ints

Java (enumerated types are equivalent to a class)

```
enum Weekday { Sunday, _____, Saturday };
```

```
Weekday day = Weekday.Sunday;
```

```
for (Weekday d : Weekday().values()) {
```

```
    }
```

Python: No enumerated types in Python.

Q: Are enumerated types really useful???

Subranges

Pascal

```
type workday : Monday .. Friday; (* base type weekday *)  
score : 0 .. 100; (* base type integer *)
```

→ Repeat same question.

→ While they have some use they are not considered to be so useful to be included in modern PLs.