Which graphics card in laptop?

Linux: lspci lgrep VGA
System >Preferences> Hardware Info
lshw
nvidia-smi

Mac: About This Mac>More Info>System Report>Graphics/Display

Windows: Device Manager; Device Adapter
or Right click on desktop; Properties>Settings Tab //
Advanced>Adaptor Tab
msinfo32
dxdiag

To find NVIDIA card **compute capability:**

https://developer.nvidia.com/cuda-gpus
Click on your GPU card to find # of compute cores

Text editors and command lines

Textwrangler, Texteditor, gedit, kate, notepad++

Syntax highlighting, text format

Not Word, Pages, or other formatting word processor

IDE

Xcode

Visual Studio

Eclipse

NetBeans

.....

C++

Derived from C starting 1979; backward compatible
Easier than C
C and C++ are the most popular computing languages

More features in C++: object oriented programming
inheritance, exception handling
classes

Your first C++ program

```
#include <iostream>
using namespace std;

int main()
{
    cout << "Hello World!" << endl;
    return 0;
}
```

Note that most lines end with ;

C Version

```
#include <stdio.h>
int main()
 {
   printf(“Hello, World!”);
}
```

Formatting

code can be most anywhere on a line

```
int main(0)  { bracket here or
   {
   }
```

output: \n \r return, \a alert, \b backspace, \t tab

Comments

```
//  single line comment

/*  multi-line comment
    runs until
    here */
```

Use comments liberally to document code.

- Explain inputs/outputs, requirements
- Explain methods, functions
- Explain usage

Typing of Variables

Variable type must be declared before assigning it a value
Integers (no decimals) and floats (decimals)

```
int a;
float b, c;
```

Variables have two parts: their value,
and their location in memory

Variable Types

1 byte = 8 bits; bit= binary digit (0,1)

Group	Type names*	Notes on size / precision
Character types	**char**	Exactly one byte in size. At least 8 bits.
	char16_t	Not smaller than `char`. At least 16 bits.
	char32_t	Not smaller than `char16_t`. At least 32 bits.
	wchar_t	Can represent the largest supported character set.
Integer types (signed)	**signed char**	Same size as `char`. At least 8 bits.
	signed **short** *int*	Not smaller than `char`. At least 16 bits.
	signed **int**	Not smaller than `short`. At least 16 bits.
	signed **long** *int*	Not smaller than `int`. At least 32 bits.
	signed **long long** *int*	Not smaller than `long`. At least 64 bits.
Integer types (unsigned)	**unsigned char**	(same size as their signed counterparts)
	unsigned short *int*	
	unsigned *int*	
	unsigned long *int*	
	unsigned long long *int*	
Floating-point types	**float**	
	double	Precision not less than `float`
	long double	Precision not less than `double`
Boolean type	**bool**	
Void type	**void**	no storage
Null pointer	**decltype(nullptr)**	

Bits and Bytes

1 byte = 8 bits; bit= binary digit (0,1)

Usually only need to use int, float,double, and bool (true or false)

Size	Unique representable values	Notes
8-bit	256	$= 2^8$
16-bit	65 536	$= 2^{16}$
32-bit	4 292 967 296	$= 2^{32}$ (~4 billion)
64-bit	18 446 744 073 309 551 616	$= 2^{64}$ (~18 billion billion)

int
long int, float
double

```
/*  size of variables on Macbook Pro
*/
#include <iostream>
using namespace std;

int main()
{
  int k;
  float d;
  double g;
  long l;
cout <<"size of int = " << sizeof(int) << endl;
cout <<"size of long = " << sizeof(long) << endl;
cout <<"size of float = " << sizeof(float) << endl;
cout <<"size of double = " << sizeof(double) << endl;

return 0;
}
```

Float and Double

± mantissa 2 ^ exponent (in binary)

32 bits, 24 significant bits; about 7 decimal digits

64 bits, 53 significant bits; about 16 decimal digits

Variable names

Must begin with a-z, A-Z, or _

Upper case different from lower case

Cannot be keywords:

char, break, float enum return struct, switch

long short typedef void union if int, etc

Constants

#define symbolic_name value

preprocessor in compiler does replacement

#define days_in_year 365

Operators

+ - / * % (remainder for integers, called mod)

++ - - increment/decrement operators

i++ is equal to i=i+1

i++ is the value in i before it has been incremented by 1.
++i is the value in i after it has been incremented by 1.

i=1;
j=i++; k=++i

what is i, j, and k?

Input to program: command line

```
 int main (int argc, char *argv[])
                          {
if (argc >1)
   {
     N= atol(argv[1]);  //parse command line for no. intervals
    np= atoi(argv[2]);
      printf("Number of intervals: %d; no. of processors: %d
            \n",N,np);
    }
   else
    {
    printf("be sure you had two args on command line for N
greater than 8!\n”);
   }
```

input/output

```
// C03:Guess.cpp
// Guess a number (demonstrates "while")
#include <iostream>
using namespace std;

int main() {
  int secret = 15;
  int guess = 0;
  // "!=" is the "not-equal" conditional:
  while(guess != secret) { // Compound statement
    cout << "guess the number: ";
    cin >> guess;
  }
  cout << "You guessed it!" << endl;
} ///:~
```

Loops

```
for (int i = 0; i< n; i++)
 {
    ….statements….
 }
```

Standard form

```
do
  {
    …statements
  }
   while (expression)
```

if-else

```
if(expression)
     statement
```

or

```
if(expression) {
   statement
 }
else {
     statement
 }
```

Program:

Write a C++ program that either takes in 8 numbers
from the command line or a cin command
and determines their sum.

Write these numbers to the command line, using cout