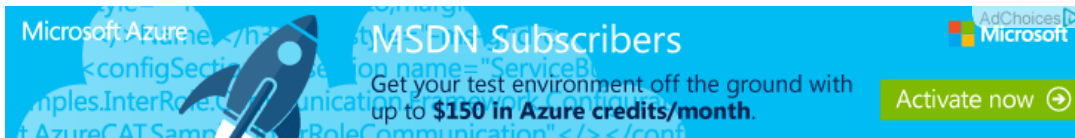


Stack Overflow is a question and answer site for professional and enthusiast programmers. It's 100% free, no registration required.

Take the 2-minute tour ×

How would you sort 1 million 32-bit integers in 2MB of RAM?



Please, provide code examples in a language of your choice.

Update: No constraints set on external storage.

Example: Integers are received/sent via network. There is a sufficient space on local disk for intermediate results.

[algorithm](#) [language-agnostic](#) [google-moderator](#)

edited Sep 25 '08 at 19:04

asked Sep 25 '08 at 15:53



[J.F. Sebastian](#)

116k 22 191 310

4 Smells like Homework – [Omar Kooheji](#) Sep 25 '08 at 15:59

Homework questions are OK as long as they are general enough where others can find the solution valuable. – [Outlaw Programmer](#) Sep 25 '08 at 16:02

I would have said that getting someone to answer your homework for you defeated the purpose. – [Omar Kooheji](#) Sep 25 '08 at 16:04

I thought newbie (self-research) questions are allowed. It seems I was mistaken. – [J.F. Sebastian](#) Sep 25 '08 at 18:48

I think was many reacts to in that respect is the sentence "Please provide code examples". If it's self-research, you don't need others to do it for you. – [Lasse V. Karlsen](#) Feb 12 '09 at 8:30

show 5 more comments

11 Answers

Split the problem into pieces small enough to fit into available memory, then use [merge sort](#) to combine them.

answered Sep 25 '08 at 15:57



[moonshadow](#)

30.1k 5 45 92

2 probably best solution, exsept you also want to have enough working space in memory to sort them... – [Omar Kooheji](#) Sep 25 '08 at 16:04

1 I'm interested in code examples (I've already read theoretical aspects in Knuth) – [J.F. Sebastian](#) Sep 25 '08 at 18:52

add a comment

Work on work you love. From home.



 **stackoverflow**careers

A bubble sort is the wrong way to go!

edited Jul 2 '09 at 9:26

answered Sep 25 '08 at 16:31



Loofer

3,183 2 25 61

2 then why did you provide it – Alcott Sep 15 '11 at 14:10

4 @alcott he was making a joke i think it was something obama said – Josh Sherick Sep 28 '11 at 0:45

[add a comment](#)

Sorting a million 32-bit integers in 2MB of RAM using Python by Guido van Rossum

answered Oct 23 '08 at 7:38



J.F. Sebastian

116k 22 191 310

6 That's suspiciously specific :) – skaffman Jul 2 '09 at 9:34

[add a comment](#)

You need to provide more information. What extra storage is available? Where are you supposed to store the result?

Otherwise, the most general answer: 1. load the first half of data into memory (2MB), sort it by any method, output it to file. 2. load the second half of data into memory (2MB), sort it by any method, keep it in memory. 3. use merge algorithm to merge the two sorted halves and output the complete sorted data set to a file.

answered Sep 25 '08 at 15:57



zvrba

14.9k 2 26 50

I've updated the question. – J.F. Sebastian Sep 25 '08 at 19:06

[add a comment](#)

1 million 32-bit integers = 4 MB of memory.

You should sort them using some algorithm that uses external storage. Mergesort, for example.

answered Sep 25 '08 at 15:55



gabr

20.9k 7 44 101

[add a comment](#)

This [wikipedia article on External Sorting](#) have some useful information.

answered Sep 25 '08 at 16:38



chakrit

32.1k 12 92 126

[add a comment](#)

Dual tournament sort with polyphased merge

```
#!/usr/bin/env python
import random
from sort import Pickle, Polyphase

nrecords = 1000000
available_memory = 2000000 # number of bytes
    #NOTE: it doesn't count memory required by Python interpreter

record_size = 24 # (20 + 4) number of bytes per element in a Python list
heap_size = available_memory / record_size
p = Polyphase(compare=lambda x,y: cmp(y, x), # descending order
              file_maker=Pickle,
              verbose=True,
              heap_size=heap_size,
              max_files=4 * (nrecords / heap_size + 1))

# put records
maxel = 1000000000
for _ in xrange(nrecords):
    p.put(random.randrange(maxel))

# get sorted records
last = maxel
for n, el in enumerate(p.get_all()):
    if el > last: # elements must be in descending order
        print "not sorted %d: %d %d" % (n, el, last)
        break
    last = el

assert nrecords == (n + 1) # check all records read
```

[edited Sep 25 '08 at 18:43](#)

[answered Sep 25 '08 at 18:33](#)



[J.F. Sebastian](#)

116k 22 191 310

[add a comment](#)

- Um, store them all in a file.
- Memory map the file (you said there was only 2M of RAM; let's assume the address space is large enough to memory map a file).
- Sort them using the file backing store as if it were real memory now!

[answered Feb 12 '09 at 8:09](#)



[Adam Hawes](#)

4,248 1 10 23

[add a comment](#)

Here's a valid and fun solution.

Load half the numbers into memory. Heap sort them in place and write the output to a file. Repeat for the other half. Use external sort (basically a merge sort that takes file i/o into account) to merge the two files.

Aside: Make heap sort faster in the face of slow external storage:

- Start constructing the heap before all the integers are in memory.
- Start putting the integers back into the output file while heap sort is still extracting elements

[answered Jan 16 '13 at 5:43](#)



[AtlasMeh-ed](#)

226 2 5

[add a comment](#)

As people above mention type int of 32bit 4 MB.

To fit as much "Number" as possible into as little of space as possible using the types int, short and char in

C++. You could be slick (but have odd dirty code) by doing several types of casting to stuff things everywhere.

Here it is off the edge of my seat.

anything that is less than $2^8(0 - 255)$ gets stored as a char (1 byte data type)

anything that is less than $2^{16}(256 - 65535)$ and $> 2^8$ gets stored as a short (2 byte data type)

The rest of the values would be put into int. (4 byte data type)

You would want to specify where the char section starts and ends, where the short section starts and ends, and where the int section starts and ends.

answered Sep 25 '08 at 16:28



J.J.

3,561 1 12 26

Wouldn't help much. Or wouldn't help at all if all numbers are bigger than 65535. – [gabr](#) Sep 25 '08 at 16:35

1 You'd waste more space keeping track of types than you'd save! – [Adam Hawes](#) Feb 12 '09 at 8:08

[add a comment](#)

No example, but [Bucket Sort](#) has relatively low complexity and is easy enough to implement

answered Sep 25 '08 at 20:28



Harald Scheirich

7,598 12 39

[add a comment](#)

Not the answer you're looking for? Browse other questions tagged [algorithm](#)

[language-agnostic](#) [google-moderator](#) or [ask your own question](#).