

Algorithms

Some history and advice as we begin our
adventure. . . .

Advice

- ✦ Attend every class. Don't miss even one.
- ✦ Put your devices away and pay attention in class.
- ✦ Take notes in class (in pencil) on paper.
- ✦ Rewrite your class notes into a new notebook as a form of studying.
- ✦ Do every lab. Sometimes twice. Commit to GitHub.
- ✦ Work on your projects every day.

Advice

- Attend every class. Don't miss even one. Why?
 - > If you're not in class I won't have the opportunity to invite you to join a faculty research project, or the IBM Joint Study, or a hackathon, or a CCAC project, or an ECRL project, or a Summer Fellowship, or the Marist Innovation Lab.
 - > If you're not in class I will never know you well enough to write you a letter of recommendation or recommend you to an employer.

ADVICE from Dr. Helen Rothberg



- ✦ **A**ction — Do more, say less. Pursue your interests. Ask for help when you need it.
- ✦ **D**etermination — Stick to it and get things done.
- ✦ **V**ision — What are you trying to become? What do you want to do with your life? How will you get there?
- ✦ **I**ntegrity — Tell the truth all the time. Don't create or participate in drama.
- ✦ **C**ommunication — Be attentive and dig in, even if it's hard.
- ✦ **E**mpathy — Dare to care about yourself and others.

Do Not Make Excuses

- ✦ I love Chris Algozzine, but I don't give a damn about Capping.
- ✦ These people failed the class.

----- Original message -----

To: Alan Labouseur/FAC/Marist@Marist
Subject: Projects
Date: Sun, Dec 9, 2018 8:33 PM

Also, I wanted to say sorry for the subpar work. I've just been incredibly overwhelmed by capping. I enjoy having you as a professor and I don't want you to think I'm just blowing your class off, I've just been trying to keep my head above water.

----- Original message -----

To: Alan Labouseur/FAC/Marist@Marist
Subject: Projects

I chose to dedicate my productive hours towards my capping project, due to the fact that I prioritize my team's performance highly.

- ✦ Don't even mention Capping to me. Seriously.
- ✦ Work on your project and assignments **every** day.

Do Not Make Excuses

- ✦ This is not acceptable:

```
----- Original message -----
```

```
To: Alan Labouseur/FAC/Marist@Marist
```

```
Subject: RE: Projects
```

```
Hi Professor Labouseur,
```

```
I just looked at my email this morning, I'm surprised you didn't get my work. I thought I pushed it Sunday, But I now realize I didn't set upstream origin for the branch, so it didn't actually get pushed. I've just re-pushed this morning so you can review it.
```

- ✦ I did not “review it”. The grade remained the same.
- ✦ If you cannot figure out GitHub, you will not be earning a CS / ITS / Cyber / Data Science degree.

Academic Honesty

- ✦ These are individual projects. All work must be your own.
- ✦ Any violation of this will result in your **immediate** and **automatic** failure of this class.
- ✦ This is your only warning. The next time we talk about this it will be in the context of me telling you that you have failed this class.
- ✦ I am serious. Ask prior students.

Academic Honesty

- ✦ I will be checking your work against...
 - your class
 - other classes
 - past projects and...

Assignments

Embrace Computer Science scholarship and write your labs using LaTeX. **OverLeaf** is probably the easiest way to do this. I've written a **lab example you can use as a template**. There are more LaTeX resources **below**.

Assignment Zero - Limericks in LaTeX

Assignment One - Data Structures and **magic items**.

Assignment Two - Sorting **magic items**.

Assignment Three - Searching and Hashing **magic items**.

Assignment Four - **Graphs** and Trees (and **magic items to find in your BST**)

Assignment Five - Dynamic Programming (**graphs**) and Greedy Algorithms (**spice**)



Academic Honesty

- ✦ I will be checking your work against...
 - your class
 - other classes
 - past projects
 - and...
 - GitHub

```
crav12345/RavosaAlgorithmAnalysis
Algorithms4Ravosa/IntergalacticReaderRavosa.java
7 // constructing several directed, weighted graphs. The program will produce
8 // those graphs and implement the Bellman-Ford path finding algorithm on them.
-
- //We print the path in Bellman-Ford.
21 public static int formatter = 0;
22
23 public static void main(String[] args) {
-
- Python Showing the top two matches Last indexed on May 17, 2019

CameronUlrich/CMPT435Algorithms
Assignment Four/AssignmentFour.py
45 edge.v.distance = edge.u.distance + edge.weight
46 edge.v.predecessor = edge.u
47
48 def BellmanFord(self):
-
- return False
142
143
144
145
146 def main():
147 print("Bellman-Ford dynamic programming algorithm for Single Source Shortest
Path:")
-
- Python Showing the top two matches Last indexed on May 13, 2019

rsbmarist/cmpt435bradshaw
Assignment4/Main.java
13 //This while loop runs the graph reading and bellman-ford methods.
14 while(graphReader.hasNextLine()){
-
- //Run the bellman ford method on thew last graph.
63 bellmanFord(g, 1);
64
65
66
67
68 File spiceFile = new File("spice.txt");
69 Scanner spiceReader = new Scanner(spiceFile);
-
- Java Showing the top four matches Last indexed on May 10, 2019
```

```
rust-lang/rust org:rust-lang CodePoint
library/std/src/sys_common/wtf8.rs 886 lines (785 sloc) 27.43 KB
view on github.com
45 /// Format the code point as 'U-' followed by four to six hexadecimal digits.
46 /// Example: 'U+1F4A9'
47
48 impl fmt::Debug for CodePoint {
49     #[inline]
50     fn fmt(&self, formatter: &mut fmt::Formatter<'>) -> fmt::Result {
51         write!(formatter, "U{:04X}", self.value)
52     }
53 }
54
55 impl CodePoint {
56     /// Unsafely creates a new 'CodePoint' without checking the value.
57     ///
58     /// Only use when 'value' is known to be less than or equal to 0x10FFFF.
59     #[inline]
60     pub unsafe fn from_u32_unchecked(value: u32) -> CodePoint {
61         CodePoint { value }
62     }
63 }
64
65 /// Creates a new 'CodePoint' if the value is a valid code point.
66 ///
67 /// Returns 'None' if 'value' is above 0x10FFFF.
68 #[inline]
69 pub fn from_u32(value: u32) -> Option<CodePoint> {
70     CodePoint::from_u32_unchecked(value).or_else(|| None)
71 }
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

Academic Honesty

- ✦ I will be using MOSS to help detect cheaters. See theory.stanford.edu/~aiken/moss/

Moss

A System for Detecting Software Similarity

UPDATES

- Feb 1, 2018 *And even more community contributions have been added!*
- Nov 9, 2017 *More community contributions have been added below ...*
- Aug 31, 2017 *Thanks to Christophe Troestler for an [OCaml client](#) for Moss.*
- May 18, 2014 *Community contributions (including a Windows submission GUI from Shane May, thanks!) are now in their own section on this page.*
- May 14, 2014 *And here is a [Java version](#) of the submission script. Thanks to Bjoern Zielke!*
- May 2, 2014 *Here is a [PHP version](#) of the submission script. Many thanks to Phillip Rehs!*
- June 9, 2011 *There were two outages over the last couple of days that lasted no more than a hour each (I think). I've made some changes to the disk management software that should prevent these problems from recurring.*
- April 29, 2011 *There was an outage lasting a few hours today, the first since last summer, but everything is back up.*
- August 1, 2010 *Everything is back to normal.*
- July 27, 2010 *The Moss server is back on line. There may be some more tuning and possibly downtime in the coming weeks, but any outages should be brief. New registrations are not yet working, but people with existing accounts can submit jobs.*
- July 25, 2010 *As many (many!) people have noticed, the Moss server has been down for all of July. Unfortunately the hardware failed while I was away on a trip. I am hopeful it will be back up within a few days.*

What is Moss?

Moss (for a Measure Of Software Similarity) is an automatic system for determining the similarity of programs. To date, the main application of Moss has been in detecting plagiarism in programming classes. Since its development in 1994, Moss has been very effective in this role. The algorithm behind moss is a significant improvement over other cheating detection algorithms (at least, over those known to us).

Write Original Code

- ✦ Infuse your code with your personality.
 - variable names, code conventions, and style
 - structure and modularization
- ✦ Write grammatically correct comments and steep them with your personality. Be snarky, clever, and funny if you like. I want to get to know you by reading your code.
 - Comments should say things about the code that the code cannot say about itself.
- ✦ List collaborators (including AI), references, and sites consulted in a comment block at the top of each file.

What about using an AI?

- I've been teaching my teenage son to drive. We started with golf carts and then moved to a real car in empty parking lots.
- He's making incremental progress built on a solid foundation (and cart paths). But he's a long way from driving in F1.
- If I were to put him in an Aston Martin he'd surely hurt himself and others.
- AI code generators are like F1 racecars. Until you have a solid foundation you're only going to hurt yourself and others by using them. But if you feel that you're ready... DRS is enabled so go for it!



When using an AI

- As ever, anything that's not 100% yours and yours alone **must be marked and cited with sources**. Whether you get help from an online AI, a tutor in the programming lab, or a random student sleeping outside of Marion Hall, you have to acknowledge that help and cite it or it is unquestionably plagiarism. Copying and pasting from a Hall of Fame project or GitHub or some other online source is unquestionably plagiarism unless you cite it; and this is no different.
- How do you cite it? Let's ask ChatGPT . . .

When using an AI



Write a citation for ChatGPT in ACM reference format



As of my last knowledge update in September 2021, there isn't a specific ACM (Association for Computing Machinery) reference format for citing AI language models like ChatGPT, as these models are relatively new and standards may not have been established at that time. However, you can follow a general citation format in accordance with ACM guidelines, and adapt it as needed based on any new guidelines that may have emerged after my last update. Here's a general citation in ACM format:

less

Copy code

```
[Author(s)]. [Year]. "ChatGPT: A Language Model for Natural Language Underst
```

Please replace `[Author(s)]`, `[Year]`, `[Name of the Conference or Journal]`, `[Volume number](Issue number)`, `[Article number]`, `[Page numbers]`, and `[DOI if available]` with the specific information relevant to the version or publication of ChatGPT you are citing. If there are specific guidelines or recommendations from ACM for citing AI language models as of 2023, I recommend following those guidelines for the most accurate and up-to-date citation format.

When using an AI

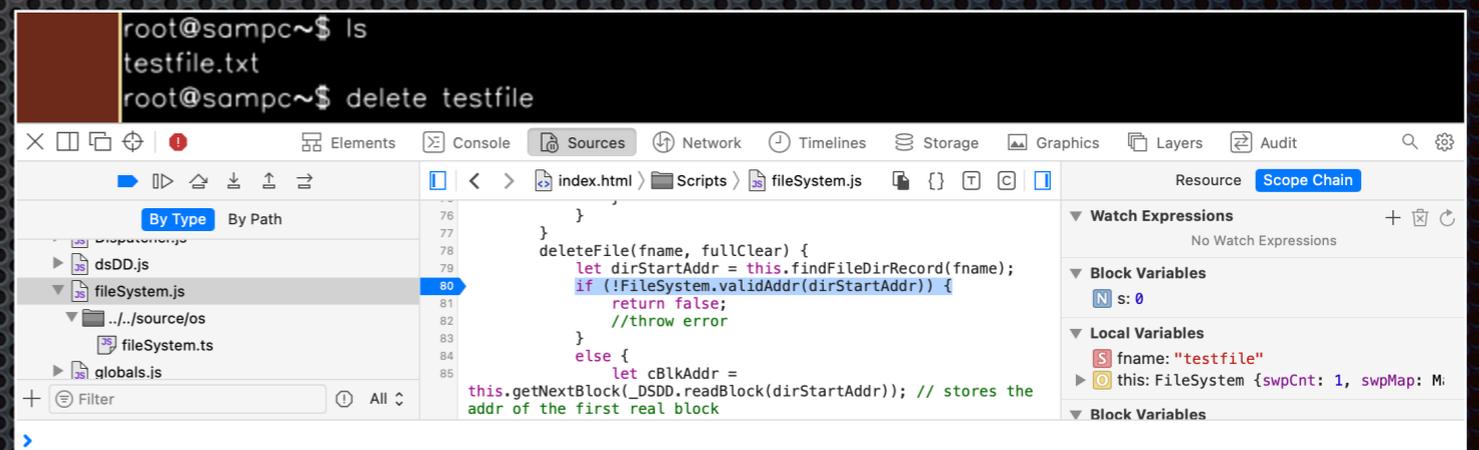
- I'm interested in learning about how well (or not?) AI-assisted software development works for you. If you make use of an AI you must document your experience with it for every project: the chat interactions, what worked and what didn't, challenges, unexpected occurrences, etc. Include a reflection about the experience as well. I'm interested to learn from you and your experiences with this new technology.
- Remember: plagiarism is one of the few sins where it is easier to receive permission than forgiveness. So if you are at all unsure about any aspect of this, ask me about it first.

Before asking me for help

- ✦ Review your code yourself by reading it from the perspective of someone seeing it for the first time.
 - Consider your variable names — are they descriptive?
 - Did you write plentiful comments about why you made those choices rather than what the code does?
 - Examine your code conventions and style — is it consistent?
 - Fix anything that's missing, unclear, or confusing.

Before asking me for help

- Set a break point in your browser or IDE and step through your code line by line for several test cases.
 - This is the first thing I'm going to ask you to do, so you might as well do it preemptively.
 - You will be **amazed** at how many
 - (1) mistakes,
 - (2) bugs, and
 - (2) off-by-one errors you'll find.



The screenshot shows a terminal window at the top with the following commands and output:

```
root@sampc~$ ls
testfile.txt
root@sampc~$ delete testfile
```

Below the terminal is a browser developer tools interface. The 'Sources' panel is open to 'fileSystem.js', showing a function `deleteFile` with a break point at line 80. The code snippet is:

```
76 }
77 }
78 deleteFile(fname, fullClear) {
79   let dirStartAddr = this.findFileDirRecord(fname);
80   if (!FileSystem.validAddr(dirStartAddr)) {
81     return false;
82     //throw error
83   }
84   else {
85     let cBlkAddr =
this.getNextBlock(_DSDD.readBlock(dirStartAddr)); // stores the
addr of the first real block
```

The 'Local Variables' panel on the right shows the following state:

- `fname`: "testfile"
- `this`: FileSystem {swpCnt: 1, swpMap: M...

When asking me for help

- ✦ Explain the problem clearly. Context matters.
- ✦ Describe several test cases that illustrate the problem.
- ✦ Tell me what you observed when stepping through your code line by line. (Do not ask me for help unless you have already done this. Seriously.)
- ✦ Make sure the latest version is committed on GitHub and include a link to it in your e-mail.