

مقدمہ

۱۔ اسیسٹنٹ انجینئر

۱۔ مجموعہ درسیہ، طرہ امتحان اور وقت
۲۔ ازامات کلاس کے ماسماہ صحت مشورہ

۲۔ سر فصل

Greedy Divide & conquer

۱۔ مقدمہ، پیرنگل اور وقت کے
۲۔ تقسیم و غلبہ، بر وقت کے
۳۔ حوصلہ دہ
۴۔ غیر از امتحان

۱۔ بر وقت کے
۲۔ بر وقت کے
۳۔ بر وقت کے
۴۔ بر وقت کے
Branch of Bound Backtracking Dynamic prog

۳۔ از امتحان ایسا
۱۔ فقرہ بیان
۲۔ فقرہ بیان
۳۔ فقرہ بیان
۴۔ فقرہ بیان

اس وقت

۲۔ ریستہ و خانگیہ - ریاضیاتی ترقی

۱۔ نکتہ نظر و روش حل مسائل

۱۔ نکتہ نظر و روش حل مسائل

۱۔ نکتہ نظر و روش حل مسائل

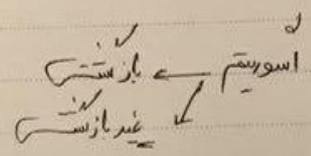
۱۔ نکتہ نظر و روش حل مسائل

①



از جمله میا از عبارات زیر کدام می توان به عنوان
 اندیسیم بنویس؟
 مجموعه مقادیر اسناد اول به نحوی و جایگزین \times تا این طول است باعث می شود شرط خانه بدو
 در یک وضعیت عدد اول داریم

مثلاً ما جمع می کنیم تا n به ما ضرب می شود \times زبان درین وضعیت است که می توانیم
 همه بصر است. و می تواند سرد و استوار می شود
 مجموع مقادیر اعداد نزدیک است تا n به نحوی و جایگزین \checkmark



اندیسیم \checkmark بازرسی \checkmark
 اندیسیم \checkmark غیر بازرسی \checkmark
 هر دو در شرایط خودشان کار می کنند

غیر بازرسی \checkmark تغییر \checkmark به صورت هر چند که می شود بازنگری این بازنگری
 به این معنی اندیسیم بازنگری \checkmark

۱) \checkmark \checkmark سوال \checkmark

n	0	1	2	3	4	5	6	7	8	...
F_n	0	1	1	2	3	5	8	13	21	...

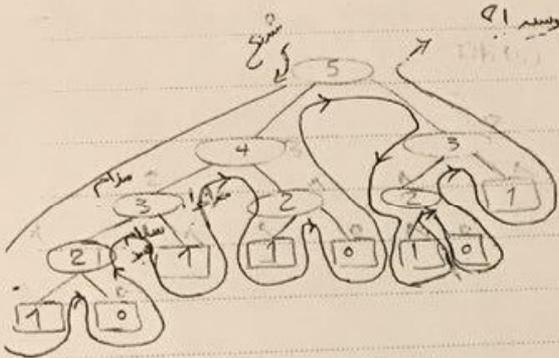
شرایط \checkmark
 $F_n = F_{n-1} + F_{n-2} \quad n \geq 2$
 شرایط \checkmark
 $F_0 = 0$ و $F_1 = 1$

```

Algorithm = Fib(n)
if n <= 1 then return(n)
else return (Fib(n-1) + Fib(n-2))
    
```

استعداد = ریاضیات
 توان = منطق
 حس = حس

مثال: مراحل مختلف اجرا $Fib(5)$ بنویسید!



تیمون ابزار و روش درخت باز است
 و به این ترتیب در صورت این ترتیب

□ ← برگ
 ○ ← ریشه و یا داخلی

مثال: تعداد فراخوانی $Fib(5)$ بنویسید!

تعداد برگ درخت باز است = 15

تعداد فراخوانی $Fib(n)$ ؟

تعداد فراخوانی $Fib(n)$ بنویسید!

n	F_n	$A_n =$
0	0	1
1	1	1
2	1	3
3	2	5
4	3	7
5	5	15
6	8	

$$A_n = 2F_{(n+1)} - 1$$

* تعداد فراخوانی؟ یعنی به ترتیب خود بنویسید استوار

نوع نسبت به این بنویسید! نسبت کنید!

$$A_n = A_{n-1} + A_{n-2} + 1 \quad n \geq 2$$

$$A_0 = 1, A_1 = 1$$

$$\begin{cases} F_n = F_{n-1} + F_{n-2} & n \geq 2 \\ F_0 = 0, F_1 = 1 \end{cases}$$

Subject: _____
Date: _____

برای a و b نسبی (مضرب) (مضرب)

از اعداد صحیح مثبت

اگر a و b نسبی باشند و $a > b > 0$ در صورتی که a و b نسبت به هم اول باشند (نسبت اول) و a و b نسبت به هم اول باشند (نسبت اول)

$a = 4, b = 5$

$GCD(4, 5) = GCD(5, 1) = GCD(1, 0) = 1$

$a = 12, b = 12$

$GCD(12, 12) = GCD(12, 0) = GCD(0, 12) = 12$

Algorithm $GCD(a, b)$

if $b = 0$ then Return (a)

else Return $(GCD(b, a \text{ mod } b))$

تعداد فراخوردهای a و b در حالت $a \leq b$ (از جمله 1 عدد قابل بخش a و b نیز شامل است) $\leq b/a$

$GCD(12, 4) = GCD(4, 0) = 4$

نسبت اول a و b در وجه متقابل از جمله 1 نسبی باشد

n	0	1	2	3	4	5	6	7	8
Kn	0	1	1	2	3	5	8	13	21
								b	a

Subject: π

Date: _____

$$\begin{aligned} \text{GCD}(15, 12) &= \text{GCD}(12, 12) = \text{GCD}(12, 0) = \text{GCD}(12, 0) = \text{GCD}(12, 0) \\ &= \text{GCD}(12, 0) = \text{GCD}(12, 0) = \text{GCD}(12, 0) = 12 \end{aligned}$$

$$F_n = \text{Round} \left\{ \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n \right\} = F_n \approx 14^n$$

تاریخ ۲

تعمیر (ادام)

حالت اسوئتم کے زیر تعمیرت زمین کے تقدیر زیر است:

* بنیادی اسوئتم کے

X * تعمیر ساز اسوئتم کے (انتہائی درست اسوئتم کے)

تعمیرت اسوئتم کے درست محول شدہ بہ انکس اور درود کے معنی خرابی سے بچنا

* عمل (تعمیر) اسوئتم کے

یہ اسوئتم کے در زمان اجراء استعداد سے ساز:

۱. CPU برا اجراء دستور کے

۲. حافظہ برا ذخیرہ ساز دادہ کے (ورود دیا، خروجی کے)۔۔۔۔۔

تعمیرت عمل پیدا اسوئتم کے نزدیکہ اطلاع مرحلہ سے تعمیرت اسوئتم کے در زمان

اجراء ۱) بہ چہ دست زمان از CPU برا اجراء دستور کے ۲) بہ چہ مقدار از حافظہ برا

ذخیرہ ساز دادہ کے نیز دارد

Subject:

Date:

1. استعمال زیادہ

2. پروفائلنگ (Profiling) : اجزاء پر مبنی درست اور خادہ مفعول پر
(تعمیر ٹیولز / اندازہ گیری ٹیولز) شدہ ہر اندازہ گیری زبان و حالت میں ہر وقت

تعداد

X

X

X

n (تعداد و ردیحات، تعداد ضربی، عیاہر در...) - مجموع ترازیتہ

تعمیر ٹیولز / اندازہ گیری ٹیولز (X) - عمدتہ مشہور

$$t_p(n) = \sum_{\text{دستور العمل}} (\text{تعداد دفعات اجراء}) \times (\text{زبان اجراء دستور العمل})$$

برائے صورت عملیہ نمونہ مہارت زبان اجراء دستور العمل

تعدادی برابر اجزاء CPU است

(5)

Subject: →

Date:

$$T(n) = \sum_{\text{دستورات}} (تعداد دستورات اجراء)$$

مطلوب است محاسبه $T(n)$ بر پایه تقسیم بر اساس زیره

$$x=0, y=1$$

حجم 8 در عمل تقسیم هر یک ششگرددن

for $k=2$ to n .do

نوع $T(n)$ زمان اجرا است $O(n-1)$
 $a_n = a_{n-1} + 2 = a_{n-2} + 4$

{

$$z = x + y \quad 2$$

$$x = y \quad 1$$

$$y = z \quad 1$$

}

$4n-4, n-2, n-1, n-2$

$5n-3$

حجم 1 و 2 و 3 و 4 و 5 و 6 و 7 و 8 و 9 و 10 و 11 و 12 و 13 و 14 و 15 و 16 و 17 و 18 و 19 و 20 و 21 و 22 و 23 و 24 و 25 و 26 و 27 و 28 و 29 و 30 و 31 و 32 و 33 و 34 و 35 و 36 و 37 و 38 و 39 و 40 و 41 و 42 و 43 و 44 و 45 و 46 و 47 و 48 و 49 و 50 و 51 و 52 و 53 و 54 و 55 و 56 و 57 و 58 و 59 و 60 و 61 و 62 و 63 و 64 و 65 و 66 و 67 و 68 و 69 و 70 و 71 و 72 و 73 و 74 و 75 و 76 و 77 و 78 و 79 و 80 و 81 و 82 و 83 و 84 و 85 و 86 و 87 و 88 و 89 و 90 و 91 و 92 و 93 و 94 و 95 و 96 و 97 و 98 و 99 و 100 و 101 و 102 و 103 و 104 و 105 و 106 و 107 و 108 و 109 و 110 و 111 و 112 و 113 و 114 و 115 و 116 و 117 و 118 و 119 و 120 و 121 و 122 و 123 و 124 و 125 و 126 و 127 و 128 و 129 و 130 و 131 و 132 و 133 و 134 و 135 و 136 و 137 و 138 و 139 و 140 و 141 و 142 و 143 و 144 و 145 و 146 و 147 و 148 و 149 و 150 و 151 و 152 و 153 و 154 و 155 و 156 و 157 و 158 و 159 و 160 و 161 و 162 و 163 و 164 و 165 و 166 و 167 و 168 و 169 و 170 و 171 و 172 و 173 و 174 و 175 و 176 و 177 و 178 و 179 و 180 و 181 و 182 و 183 و 184 و 185 و 186 و 187 و 188 و 189 و 190 و 191 و 192 و 193 و 194 و 195 و 196 و 197 و 198 و 199 و 200 و 201 و 202 و 203 و 204 و 205 و 206 و 207 و 208 و 209 و 210 و 211 و 212 و 213 و 214 و 215 و 216 و 217 و 218 و 219 و 220 و 221 و 222 و 223 و 224 و 225 و 226 و 227 و 228 و 229 و 230 و 231 و 232 و 233 و 234 و 235 و 236 و 237 و 238 و 239 و 240 و 241 و 242 و 243 و 244 و 245 و 246 و 247 و 248 و 249 و 250 و 251 و 252 و 253 و 254 و 255 و 256 و 257 و 258 و 259 و 260 و 261 و 262 و 263 و 264 و 265 و 266 و 267 و 268 و 269 و 270 و 271 و 272 و 273 و 274 و 275 و 276 و 277 و 278 و 279 و 280 و 281 و 282 و 283 و 284 و 285 و 286 و 287 و 288 و 289 و 290 و 291 و 292 و 293 و 294 و 295 و 296 و 297 و 298 و 299 و 300 و 301 و 302 و 303 و 304 و 305 و 306 و 307 و 308 و 309 و 310 و 311 و 312 و 313 و 314 و 315 و 316 و 317 و 318 و 319 و 320 و 321 و 322 و 323 و 324 و 325 و 326 و 327 و 328 و 329 و 330 و 331 و 332 و 333 و 334 و 335 و 336 و 337 و 338 و 339 و 340 و 341 و 342 و 343 و 344 و 345 و 346 و 347 و 348 و 349 و 350 و 351 و 352 و 353 و 354 و 355 و 356 و 357 و 358 و 359 و 360 و 361 و 362 و 363 و 364 و 365 و 366 و 367 و 368 و 369 و 370 و 371 و 372 و 373 و 374 و 375 و 376 و 377 و 378 و 379 و 380 و 381 و 382 و 383 و 384 و 385 و 386 و 387 و 388 و 389 و 390 و 391 و 392 و 393 و 394 و 395 و 396 و 397 و 398 و 399 و 400 و 401 و 402 و 403 و 404 و 405 و 406 و 407 و 408 و 409 و 410 و 411 و 412 و 413 و 414 و 415 و 416 و 417 و 418 و 419 و 420 و 421 و 422 و 423 و 424 و 425 و 426 و 427 و 428 و 429 و 430 و 431 و 432 و 433 و 434 و 435 و 436 و 437 و 438 و 439 و 440 و 441 و 442 و 443 و 444 و 445 و 446 و 447 و 448 و 449 و 450 و 451 و 452 و 453 و 454 و 455 و 456 و 457 و 458 و 459 و 460 و 461 و 462 و 463 و 464 و 465 و 466 و 467 و 468 و 469 و 470 و 471 و 472 و 473 و 474 و 475 و 476 و 477 و 478 و 479 و 480 و 481 و 482 و 483 و 484 و 485 و 486 و 487 و 488 و 489 و 490 و 491 و 492 و 493 و 494 و 495 و 496 و 497 و 498 و 499 و 500 و 501 و 502 و 503 و 504 و 505 و 506 و 507 و 508 و 509 و 510 و 511 و 512 و 513 و 514 و 515 و 516 و 517 و 518 و 519 و 520 و 521 و 522 و 523 و 524 و 525 و 526 و 527 و 528 و 529 و 530 و 531 و 532 و 533 و 534 و 535 و 536 و 537 و 538 و 539 و 540 و 541 و 542 و 543 و 544 و 545 و 546 و 547 و 548 و 549 و 550 و 551 و 552 و 553 و 554 و 555 و 556 و 557 و 558 و 559 و 560 و 561 و 562 و 563 و 564 و 565 و 566 و 567 و 568 و 569 و 570 و 571 و 572 و 573 و 574 و 575 و 576 و 577 و 578 و 579 و 580 و 581 و 582 و 583 و 584 و 585 و 586 و 587 و 588 و 589 و 590 و 591 و 592 و 593 و 594 و 595 و 596 و 597 و 598 و 599 و 600 و 601 و 602 و 603 و 604 و 605 و 606 و 607 و 608 و 609 و 610 و 611 و 612 و 613 و 614 و 615 و 616 و 617 و 618 و 619 و 620 و 621 و 622 و 623 و 624 و 625 و 626 و 627 و 628 و 629 و 630 و 631 و 632 و 633 و 634 و 635 و 636 و 637 و 638 و 639 و 640 و 641 و 642 و 643 و 644 و 645 و 646 و 647 و 648 و 649 و 650 و 651 و 652 و 653 و 654 و 655 و 656 و 657 و 658 و 659 و 660 و 661 و 662 و 663 و 664 و 665 و 666 و 667 و 668 و 669 و 670 و 671 و 672 و 673 و 674 و 675 و 676 و 677 و 678 و 679 و 680 و 681 و 682 و 683 و 684 و 685 و 686 و 687 و 688 و 689 و 690 و 691 و 692 و 693 و 694 و 695 و 696 و 697 و 698 و 699 و 700 و 701 و 702 و 703 و 704 و 705 و 706 و 707 و 708 و 709 و 710 و 711 و 712 و 713 و 714 و 715 و 716 و 717 و 718 و 719 و 720 و 721 و 722 و 723 و 724 و 725 و 726 و 727 و 728 و 729 و 730 و 731 و 732 و 733 و 734 و 735 و 736 و 737 و 738 و 739 و 740 و 741 و 742 و 743 و 744 و 745 و 746 و 747 و 748 و 749 و 750 و 751 و 752 و 753 و 754 و 755 و 756 و 757 و 758 و 759 و 760 و 761 و 762 و 763 و 764 و 765 و 766 و 767 و 768 و 769 و 770 و 771 و 772 و 773 و 774 و 775 و 776 و 777 و 778 و 779 و 780 و 781 و 782 و 783 و 784 و 785 و 786 و 787 و 788 و 789 و 790 و 791 و 792 و 793 و 794 و 795 و 796 و 797 و 798 و 799 و 800 و 801 و 802 و 803 و 804 و 805 و 806 و 807 و 808 و 809 و 810 و 811 و 812 و 813 و 814 و 815 و 816 و 817 و 818 و 819 و 820 و 821 و 822 و 823 و 824 و 825 و 826 و 827 و 828 و 829 و 830 و 831 و 832 و 833 و 834 و 835 و 836 و 837 و 838 و 839 و 840 و 841 و 842 و 843 و 844 و 845 و 846 و 847 و 848 و 849 و 850 و 851 و 852 و 853 و 854 و 855 و 856 و 857 و 858 و 859 و 860 و 861 و 862 و 863 و 864 و 865 و 866 و 867 و 868 و 869 و 870 و 871 و 872 و 873 و 874 و 875 و 876 و 877 و 878 و 879 و 880 و 881 و 882 و 883 و 884 و 885 و 886 و 887 و 888 و 889 و 890 و 891 و 892 و 893 و 894 و 895 و 896 و 897 و 898 و 899 و 900 و 901 و 902 و 903 و 904 و 905 و 906 و 907 و 908 و 909 و 910 و 911 و 912 و 913 و 914 و 915 و 916 و 917 و 918 و 919 و 920 و 921 و 922 و 923 و 924 و 925 و 926 و 927 و 928 و 929 و 930 و 931 و 932 و 933 و 934 و 935 و 936 و 937 و 938 و 939 و 940 و 941 و 942 و 943 و 944 و 945 و 946 و 947 و 948 و 949 و 950 و 951 و 952 و 953 و 954 و 955 و 956 و 957 و 958 و 959 و 960 و 961 و 962 و 963 و 964 و 965 و 966 و 967 و 968 و 969 و 970 و 971 و 972 و 973 و 974 و 975 و 976 و 977 و 978 و 979 و 980 و 981 و 982 و 983 و 984 و 985 و 986 و 987 و 988 و 989 و 990 و 991 و 992 و 993 و 994 و 995 و 996 و 997 و 998 و 999 و 1000

$O(n)$ بر پایه زمان

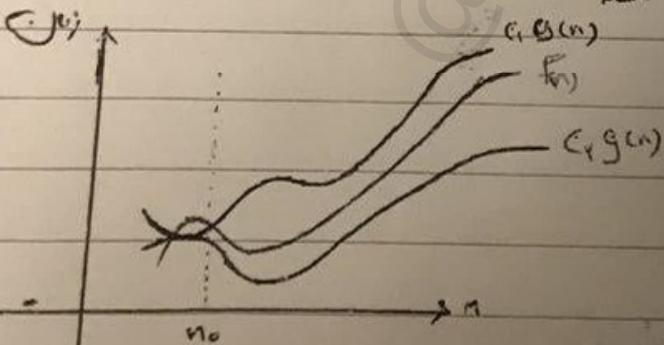
$O(n-1)$ بر پایه زمان

$O(n)$

تقریب \leftarrow معادله اساسی \leftarrow $0 < c < \infty$

تقریب \leftarrow (بهترین حالت زمان اجرا)

$$f(n) = O(g(n)) \iff \exists c_1 > 0, \exists n_0 \text{ (مثبت)} \forall n \geq n_0 \rightarrow f(n) \leq c_1 g(n)$$



2

Subject:

Date:

$f(n) = 4n^2 + 7n + 4 = O(n^2)$ شماره
 \uparrow \downarrow
 $g(n)$

$4n^2 + 7n + 4 \leq C_1 n^2 \xrightarrow{C_1=7} 4n^2 + 7n + 4 \leq 7n^2 \Rightarrow 7n + 4 \leq 3n^2$ $n_0 = 4$
 $n > 4$

تعریف O (حالت زمان اجرا)

$f(n) = O(g(n)) \iff \exists C_1 > 0, \exists n_0 \forall n \geq n_0 \iff f(n) \leq C_1 g(n)$

→ [نمودار] ←

$f(n) = 4n^2 + 7n + 4 = \Omega(n^2)$ شماره
 \uparrow \downarrow
 $g(n)$

$4n^2 + 7n + 4 \geq C_2 n^2 \xrightarrow{C_2=4} 4n^2 + 7n + 4 \geq 4n^2 \Rightarrow 7n + 4 \geq 0$ $n_0 = 1$
 $n > 1$

تعریف Ω (حالت زمان اجرا)

$f(n) = \Theta(g(n)) \iff \exists C_1, C_2 > 0, \exists n_0 \forall n \geq n_0 \Rightarrow C_1 g(n) \leq f(n) \leq C_2 g(n)$

- $f(n) = O(g(n))$
- $f(n) = \Omega(g(n))$
- $f(n) = \Theta(g(n))$

Aidin Giti Nejad

Subject: 4

Date:

$$f(n) = 4n^2 + 7n + 4 = O(n^2) \text{ نشان}$$

$n_0 = 4$ از مثال قبل تقسیم شود \times ؟

تعریف

$$f(n) = o(g(n)) \iff \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \rightarrow 0$$

تعریف w

$$f(n) = w(g(n)) \iff \lim_{n \rightarrow \infty} \frac{f(n)}{g(n)} \rightarrow \infty$$

$$4n^2 + 7n + 4 = o(n^2) \text{ نشان دهید}$$

$$\lim_{n \rightarrow \infty} \frac{4n^2 + 7n + 4}{n^2} \rightarrow 0$$

$$4n^2 + 7n + 4 = w(n) \text{ نشان دهید}$$

$$\lim_{n \rightarrow \infty} \frac{4n^2 + 7n + 4}{n} \rightarrow \infty$$

Subject: ψ

Date:

$2 \leq 2$

$2 \leq 2$

$2 \leq 2$

نتیج

$$\left. \begin{array}{l} 2 \leq 2 \\ 2 \leq 3 \end{array} \right\} \begin{array}{l} n^2 = o(n^2) \\ n^2 = o(n^3) \end{array}$$

$2 \leq 2$

$$\lim_{n \rightarrow \infty} \frac{n^2}{n^2} = 1 \neq 0 \text{ (مستوی نه می شود)}$$

$2 \leq 3$

$$\lim_{n \rightarrow \infty} \frac{n^2}{n^3} = 0 \text{ (مستوی می شود)}$$

(در واقع همان قضیه $\lim_{n \rightarrow \infty} \frac{1}{n^k} = 0$ است)

$2 \leq 2, 2 \leq 2, 2 \leq 2$

برای $a, b \in \mathbb{R}$ خاصیت $a \leq b$ دارد!

$a \leq b \Rightarrow b \leq a$

برای $a, b \in \mathbb{R}$ خاصیت $a \leq b$ دارد!

$$f(x) = o(g(x)) \Rightarrow g(x) = o(f(x))$$

برای $a, b \in \mathbb{R}$ خاصیت $a \leq b$ دارد!

Subject:

Date:

رباطاً تقسیم دایره = $\frac{1}{n}$ لایه \leftarrow $\frac{1}{n}$ تقسیم دایره

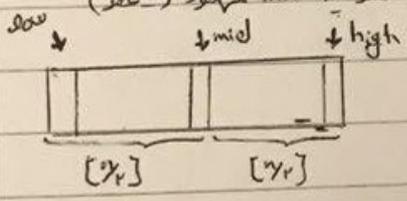
سایه :
تقسیم عناصر ماتریس به n قسمت مساوی
A [1..n]

روش ۲ :

Algorithm: $\min(A[1]) = \min(A[2]) = \dots = \min(A[n])$
 $\max(A[1])$

Algorithm: $\min(A[1]) = \min(A[2]) = \dots = \min(A[n])$
 $\max(A[1])$

برای $n=1$ \leftarrow \min, \max (جستار)
برای $n=2$ \leftarrow \min, \max (جستار)
برای n \leftarrow \min, \max (جستار)



	low			mid				high	
i	1	2	3	4	5	6	7	8	9
A[i]	30	80	50	70	10	20	20	40	60

$L_{\max} = 80$
 $L_{\min} = 10$

$R_{\max} = 70$
 $R_{\min} = 60$

$F_{\max} = \max(L_{\max}, R_{\max}) = 80$
 $F_{\min} = \min(L_{\min}, R_{\min}) = 10$

Aidin Giti Nejad

Subject:

N

Date:

Subject:

مسئله بر روی این

Algorithm Find_max_min(low, high, Fmax, Fmin)

if low = high then Fmax = Fmin = A[low]

else if high = low + 1 then if A[low] < A[high] then

{ Fmax = A[high], Fmin = A[low] }

else

{ Fmax = A[low], Fmin = A[high] }

else

{

mid = $\left\lfloor \frac{\text{low} + \text{high}}{2} \right\rfloor$

Find_max_min(low, mid, L_max, L_min)

Find_max_min(mid + 1, high, R_max, R_min)

Fmax = maximum { L_max, R_max }

Fmin = minimum { L_min, R_min }

}

۱/۲

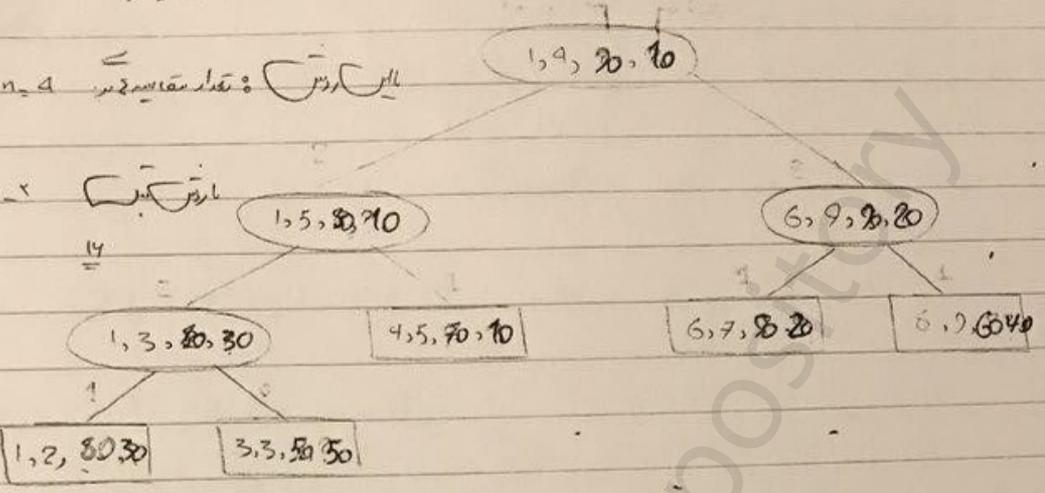
Subject:

Date:

مثال: دایک نمبر آج $\text{Find} - \text{max} - \text{min}(1, 2, P_{\text{max}}, P_{\text{min}})$ لے کر دیکھو ایسی ہی

لیں کہیں: تقاریر سے $n=4$ ہوا ہے

$n=2$ کا صورت



$n=2$ ہاں ہے تقاریر سے یا صرف $n=2$ کے خواہے ہو وہ یہ کہا جاتا ہے

تو $n=2$ عنصر کا عنصر

تو

تقاریر سے n لازم ہونا پڑتا ہے

تو n عنصر ہاں ہے تقاریر سے

تو

$$T(n) = T\left(\frac{n}{2}\right) + 2 \quad n > 2$$

تو

$$T(2) = 1$$

شرط ہے

یہاں سے

Subject: 4,

Date:

در این مسئله از روش ضرب متوالی استفاده می‌کنیم

$$t(n) = r + rL\left(\frac{n}{r}\right) = r + r \left[r + rL\left(\frac{n}{r}\right) \right]$$

$$= r + r^2 + r^2 \left[r + rL\left(\frac{n}{r^2}\right) \right]$$

$$= r + r^2 + r^3 + \dots + r^{m-1} + r^m L\left(\frac{n}{r^m}\right) = t(n) - 1$$

$$= \underbrace{r + r^2 + r^3 + \dots + r^{m-1}}_{\leftarrow} + r^m$$

$$r^m - 1 + r^m - 1 \Rightarrow n - r + \frac{n}{r} \quad \frac{r^m}{r^r} = r^*$$

$$\Rightarrow t(n) = \frac{n}{r} - r$$

$$1 + r + r^2 + \dots + r^{m-1} = \frac{r^m - 1}{r - 1} \quad \leftarrow \text{در اینجا!}$$

Aidin Giti Nejad

Subject:

Date:

تقسیم برعکس (الزاماً در جزئیات نیست)

برای تقسیم دایره (لازمه بخش اورتوم تقسیم دایره)

مثال: $n=2$

باید از \min تا \max در n عنصری است \Rightarrow روش تقسیم دایره $n=2$ است

مثال: $n=2$

$$\left\{ \begin{array}{l} T(n) = 2T\left(\frac{n}{2}\right) + 2 \\ T(2) = 1 \end{array} \right. \Rightarrow T(n) = 2 \log_2 n - 2$$

روش جاگذاری
با انداز

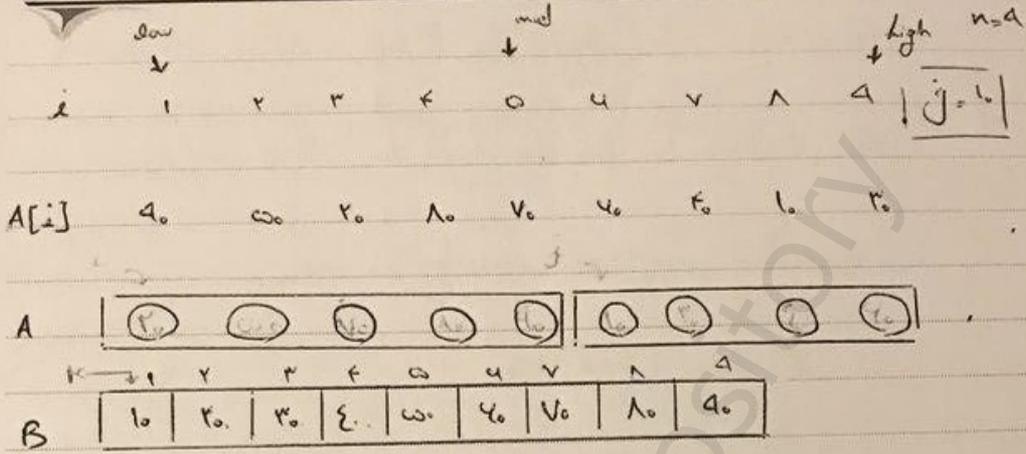
اورتوم مرتب ساز ادغام (mergesort)

نوعی (رود) آرایه مرتب $A[1..n]$

هدف (خروجی) آرایه مرتب شده به صورت عدد

توضیح: امیاشان

۱۷



Algorithm merge (low, mid, high)

$i = \text{low}$, $j = \text{mid} + 1$, $k = \text{low}$

while ($i \leq \text{mid}$) and ($j \leq \text{high}$) do

{
 if $A[i] < A[j]$ then { $B[k] = A[i]$, $i = i + 1$ }
 else { $B[k] = A[j]$, $j = j + 1$ }
 $k = k + 1$
 }

for $l = i$ to mid do { $B[k] = A[l]$, $k = k + 1$ }

For $l=j$ to high do $\{ B[k] = A[l], k = k+1 \}$

For $l=low$ to low to high do $A[l] = B[l]$

Algorithm mergesort (low, high)

if low < high then

{

$$mid = \left\lfloor \frac{low+high}{2} \right\rfloor$$

mergesort (low, mid)

mergesort (mid+1, high)

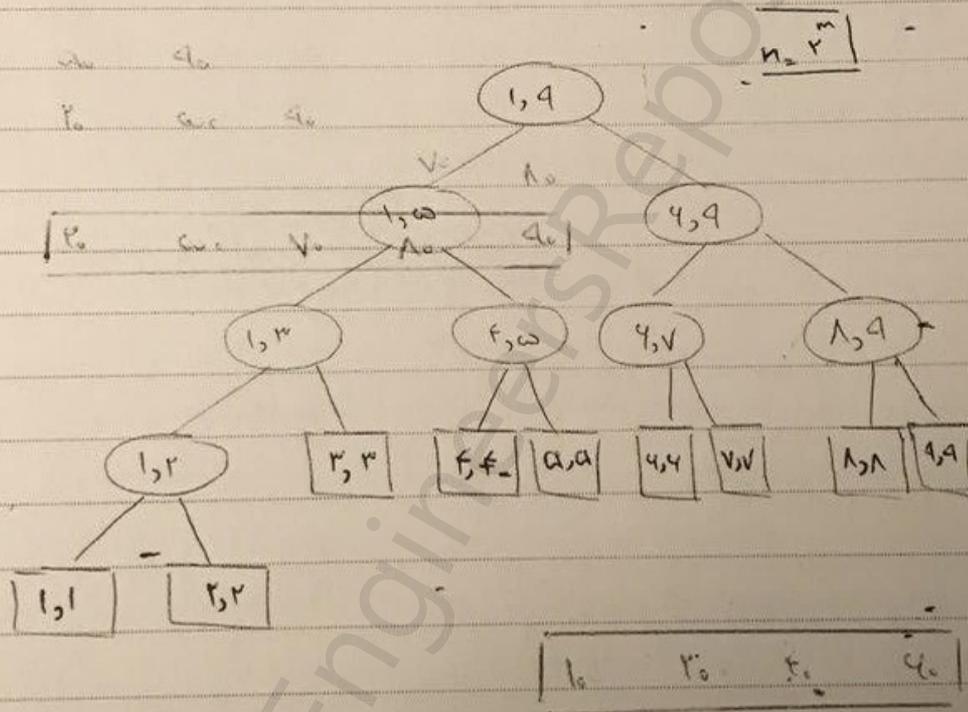
merge (low, mid, high)

}

Case 1: $\left\lfloor \frac{low+high}{2} \right\rfloor$
merge (low, mid)

← Merge Sort (1,9) ←
 low → mid → high
 i 1 2 3 4 5 6 7 8 9

A[i] 4 6 8 10 12 14 16 18 20



تقسیم به دو قسمت و بعد از آن ادغام!

mergesort تقسیم و تسطیح

تقسیم $\leftarrow \leftarrow \leftarrow$
 زمان $t(n)$ ، $n = r^m$ ؟ لازم به تقسیم است

تقسیم \leftarrow تقسیم و تسطیح

$$\begin{cases} t(n) = r t(n/r) + n & n > 1 \\ t(1) = 0 \end{cases}$$

مثال 8: زمان تقسیم و تسطیح * لازم به تقسیم است!

$$t(n) = n + r t(n/r) =$$

$$t(n) = n + r \left\{ n/r + r t(n/r) \right\}$$

$$t(n) = n + n + r t(n/r)$$

$$t(n) = n + n + r \left\{ n/r + r t(n/r) \right\}$$

$$t(n) = \underbrace{n + n + n}_{r-1} + r^m t(n/r^m) \quad r^m = t(1) = 0$$

$$t(n) = \underbrace{n + n + n + \dots + n}_{r^m} + r^m t(n/r^m)$$

PASHA

$$t(n) = mn = n \log_r n$$

۲۱

بهرین حالت 8

$$T(n) = 2T(n/2) + n - 1$$

دومین حالت 8

$$T(n) = 2T(n/2) + n/2$$

تعمیرات 8

$$T(n) = at(n/b) + cn^k \quad \text{جواب لایه تغییر دین}$$

در صورتی که $a < b^k$ ، $b < 1$ ، $a > 0$ ، $c > 0$ است به صورت زیر است

$$T(n) = \Theta\left(n^{\log_b a}\right) \quad \leftarrow b^k < a \quad \text{این است}$$

$$T(n) = \Theta\left(n^k \log n\right) \quad \leftarrow b^k = a \quad \text{ب است}$$

$$T(n) = \Theta\left(n^k\right) \quad \leftarrow b^k > a \quad \text{ج است}$$

$\frac{2}{n \times 1}$

$$\begin{cases} T(n) = rT(n/r) + r \\ T(1) = 1 \end{cases} \Rightarrow T(n) = r^2 \log_r n - r \Rightarrow \underline{T(n) = \Theta(n^2)} \quad (1)$$

$$\begin{cases} T(n) = rT(n/r) + n \\ T(1) = 1 \end{cases} \Rightarrow T(n) = n \log_r n \Rightarrow \underline{T(n) = \Theta(n \log n)} \quad (2)$$

مثال: جواب این تقسیم دایره (1) از درجه اول است پس این!

$a = r \quad b = r \quad k = 0$

$b^k = r^0 = 1$ (بزرگتر)
 $a = r$ (کوچکتر)
 $\Rightarrow T(n) = \Theta\left(n \log_r r\right) = \Theta(n^1)$

مثال: جواب این تقسیم دایره (2) از درجه اول است پس این!

$a = r \quad b = r \quad k = 1$

$b^k = r^1 = r$ (بزرگتر)
 $a = r$ (کوچکتر)
 $\Rightarrow T(n) = \Theta(n \log n)$

$$T(n) = O(?) \quad \Leftarrow \quad T(n) = r T(\sqrt{n}) + \lg n \quad \text{تالیه آید}$$

$$T(r^m) = r T(\sqrt{r^m}) + m$$

$$\text{توی} \quad S(m) = T(r^m) = T(n)$$

$$\sqrt[r^m]{} = r^{m/r}$$

$$S(m) = r S(m/r) + m$$

$$a = r \quad b = r \quad k = 1$$

$$b^k = r^1 = r = \text{توی} \Rightarrow S(m) = O(m \lg m)$$

$$a = r$$

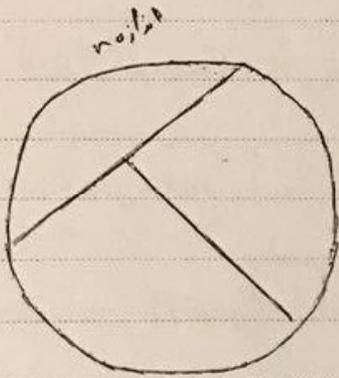
$$\Rightarrow T(n) = O(\lg n \cdot \lg \lg n) \quad \text{توی}$$

$$= O(k \lg n) \quad \text{توی}$$

ارزش \leftarrow تقسیم و غلبه

بزرگترین زیرمجموعه تقسیم و غلبه زینت به سانه اولیه زیر مسائل! اشتباه آن در صورت

بزرگترین اندازه سانه به هر قسمت تقسیم می شوند



تقسیم و درود \leftarrow سانه $A[low]$

Aidin Giti Nejad

Algorithm DP \leftarrow (low, high)

if small (low, high) then Return G (low, high)

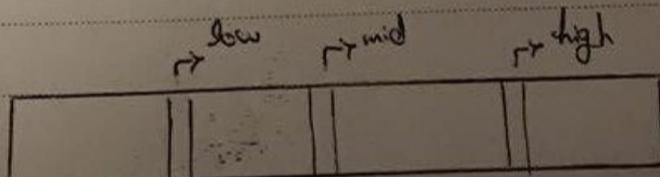
else

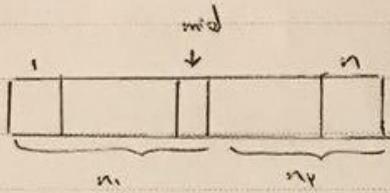
\uparrow

mid = Divide (low, high)

Return combine (DP (low, mid), DP (mid+1, high))

?





تقسیم این دو

$$\begin{cases} t(x) = t(x_1) + t(x_2) + P(x) \\ t(x) = g(x) = \dots \end{cases}$$

صورتی حالت زمانه اتفاق می افتد

$$n_1 \cong n_2 \cong n_3$$

در حالت خاص $n = 2^m \Rightarrow t(x) = 2^m t(x_1) + P(x)$

بهترین حالت زمانه اتفاق می افتد

~~n_1 (نویسنده) n_2 (مقاله) n_3 (مقاله) و ...~~

ساده ترین روش

روش ۱: a, b عدد صحیح، تقسیم دهایی، تقسیم و عمل برعکس

$$A(x) = a_0 + a_1x + a_2x^2 + \dots + a_nx^n$$

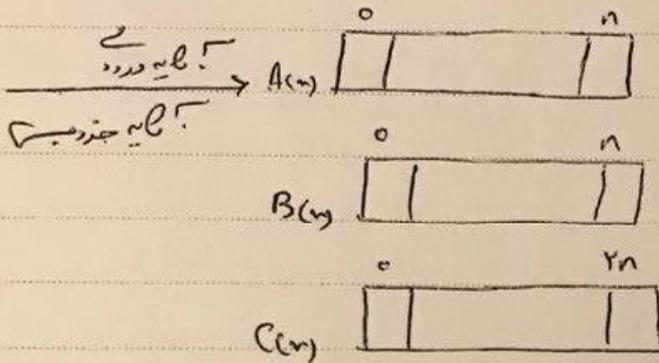
$$B(x) = b_0 + b_1x + b_2x^2 + \dots + b_nx^n$$

$$C(x) = A(x) \cdot B(x) = ?$$

حاصل می شود

$$C(x) = c_0 + c_1 x^1 + c_2 x^2 + \dots + c_n x^n$$

$$\begin{cases} c_0 = a_0 b_0 \\ c_1 = a_0 b_1 + a_1 b_0 \\ c_2 = a_0 b_2 + a_1 b_1 + a_2 b_0 \\ \vdots \end{cases} \Rightarrow c_k = \sum_{i=0}^k a_i b_{k-i} \quad \begin{matrix} i=0, \dots, k \\ k=0, \dots, n \end{matrix}$$



(Handwritten signature)

Algorithm Multiply (A, B, C)

For $i = 0$ to n do $c[i] = 0$

For $i = 0$ to n do {

For $j = 0$ to n do {

$$c[i+j] = c[i+j] + A[i] * B[j]$$

}

}

Subject 144
Date _____

$$A(x) = 1 + 7x + x^2 + 7x^3$$

$$B(x) = 7 + x + 7x^2 + x^3$$

$$C(x) = A(x) \cdot B(x) = ? \rightarrow C$$

$$7 + 8x + 14x^2 + 14x^3 + 8x^4 + 7x^5 + x^6$$

Algorithm multiply (A, B, C)

for $i=0$ to n do $C[i] = 0$

for $j=0$ to n do

for $g=0$ to n do

$$C[i+j] = C[i+j] + A[i] \cdot B[j]$$

	0	1	2	3	4	5	6
$i=0$	7	0	0	0	0	0	0

	0	1	2	3	4	5	6
$i=1$	7	8	0	0	0	0	0

	0	1	2	3	4	5	6
$i=2$	7	15	8	0	0	0	0

مقسوم عليه (توضیح اول مثال)

$$A(x) = (1 + 7x) + x^2(1 + 7x)$$

$$C(x) = A(x)B(x) = [A_L(x) + A_H(x)x^r][B_L(x) + B_H(x)x^r]$$

$$B(x) = (7 + x) + x^2(7 + x)$$

$$C(x) = A_L(x)B_L(x) + [A_L(x)B_H(x) + A_H(x)B_L(x)]x^r + A_H(x)B_H(x)x^{2r}$$

$$+ \frac{A_H(x)B_H(x)x^{2r}}{C(x)}$$

AA

Subject _____

Date _____

تاریخ

$$C_1(x) = (1+x)(1+x) = 1+x + x + x^2 = 1+2x + x^2$$

$$C_2(x) = (1+x)(1+x) + (1+x)(1+x) = 1+x+4x+2x^2 + 1+x+x+x^2 = 2+6x+3x^2$$

$$C_3(x) = (1+x)(1+x) = 1+x+4x+2x^2 = 1+5x+2x^2$$

$$C(x) = (1+x)(1+x) + (1+x)(1+x) + (1+x)(1+x) = 1+2x+2x^2 + 1+2x+2x^2 + 1+2x+2x^2 = 3+6x+6x^2$$

$$C(x) = 1+2x+2x^2 + 1+2x+2x^2 + 1+2x+2x^2 = 3+6x+6x^2$$

(x-1)

سأضرب في (x-1) من أجل

أضرب في (x-1) من أجل

(x-1)

سأضرب في (x-1) من أجل

أضرب في (x-1) من أجل

(x-1)

سأضرب في (x-1) من أجل

سأضرب في (x-1) من أجل

(x-1)

19

موضوع ۸

سأنا صند چنجه، از درجه $n-1$ هسند $A(n)$ ، $B(n)$ ، $n=2^m$

سأنا صند چنجه، از درجه $n-1$ هسند $A(n)$ ، $B(n)$ ، $n=2^m$

$$T(n) = A_2(n) B_2(n) + [A_2(n) B_2(n) + A_2(n) B_2(n)] 2^{n/2} + A_2(n) B_2(n) 2^n$$

@EngineersRepository

۱۷

روش تقسیم و غلبه

طبله ۴

سأه ضرب خود عمل، (یا، ر، د، ا، م، ن)

روش ۱، ۲، ۳

۱، تقسیم و غلبه

۲، تقسیم و غلبه سریع

سأه ضرب خود عمل، از $n-1$

از $n-1$ ضرب خود عمل، از $n-1$

$T(n) = T(n/2) + r(n-1)$ ضرب $n-1$ ضرب است

تقسیم و غلبه

$$T(n) = T(n/2) + r(n-1) \xrightarrow{\text{تقسیم و غلبه}} \begin{cases} a=2 \\ b=r \end{cases}$$

$$T(n) = \Theta\left(n \log_2 n\right) = g(n)$$

تاریخ این اسو را بنویسید

← ضرب در عدد جیب جیب

مثال:

$$C = \begin{bmatrix} 2 & 7 & 7 & 12 & 7 & 9 & 2 \end{bmatrix}$$

$$(2121) \times (1212) = 2, 782, 752$$

ضرب ماتریس

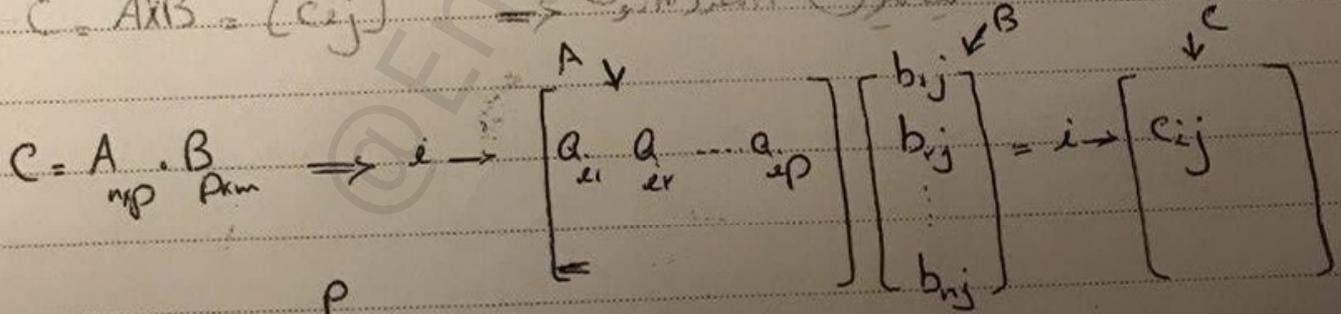
روش ۱: ۱. عدد
۲. تقسیم در عدد
۳. تقسیم در عدد

روش عددی

مضرب $A = [a_{ij}]$ n (سطر) در p (ستون) ، $B = [b_{ij}]$ p (ستون) m (سطر) ماتریس

p (سطر) در m (ستون) ماتریس

$$C = A \times B = [c_{ij}]$$



$$c_{ij} = \sum_{k=1}^p a_{ik} + b_{kj}$$

۱۳۳۳

⇒ Algorithm multmatrix (A, B, C)

For i=1 to n do

For j=1 to m do

{

C[i,j] = 0

For k=1 to p do

C[i,j] = C[i,j] + A[i,k] * B[k,j]

}

}

⇒ $\Theta(nmp)$

* ندر ضرب در ماتریس A، B به ترتیب $n \times p$ و $p \times m$ به صورت A.B

* نیاز به \boxed{nmp} ضرب عدد است

دانش تخصصی و علم

مصفوفه $n \times n$ در ماتریس A, B در ماتریس $n \times n$ باشد

مصفوفه ماتریس A, B به چهار زیر ماتریس A_{ij}, B_{ij}, C_{ij} و $n \times n$ باشد

$C = A \times B$ در $n \times n$ اندازه داشته باشد

$$A = \begin{bmatrix} A_{11} & A_{1r} \\ A_{21} & A_{2r} \end{bmatrix} \quad B = \begin{bmatrix} B_{11} & B_{1r} \\ B_{21} & B_{2r} \end{bmatrix}$$

$$\Rightarrow \begin{cases} C_{11} = A_{11}B_{11} + A_{1r}B_{r1} \\ C_{1r} = A_{11}B_{1r} + A_{1r}B_{rr} \\ C_{21} = A_{21}B_{11} + A_{2r}B_{r1} \\ C_{2r} = A_{21}B_{1r} + A_{2r}B_{rr} \end{cases}$$

$$C = \begin{bmatrix} C_{11} & C_{1r} \\ C_{21} & C_{2r} \end{bmatrix} = \begin{bmatrix} A_{11} & A_{1r} \\ A_{21} & A_{2r} \end{bmatrix} \begin{bmatrix} B_{11} & B_{1r} \\ B_{21} & B_{2r} \end{bmatrix}$$

مکمل

مکان هند در ماتریس $n \times n$ از هر سطر ماتریس $n \times n$ در $n \times n$

$$T(n) = n \cdot T(n/2) + T(n/2)$$

$$\begin{cases} a=1 \\ b=2 \\ k=2 \end{cases} \Rightarrow b = r, a=1 \quad \rightarrow \quad a > b^k, n > r \Rightarrow T(n) = \theta(n \log_r n) = \theta(n^3)$$

۱۰

Subject
Date

$$L(\omega) = \sqrt{L(\omega_r) + (K\omega^2/2)}$$

جواب *

$$L(\omega) = \alpha(\omega) \quad \omega \rightarrow \omega_0$$

$$a = \omega_0, b = \omega_0, K = \omega_0^2$$

@EngineersRepository

۴۴

پہلے کے تقسیم و غلبہ (اولیٰ)

سہل جسموں پر

۲. اسی وقت جسموں پر

دورے کے آگے بڑھنا
 $A[1..n]$ نہ صورت معقول
 ۱-۲-۳
 دیکھ کر

ضروری ہے اور $1 \leq i \leq n$ ہر i کے لیے $x = A[i]$ دریں صورت ضروری ہے

یہ ہے

دورے میں صورت ضروری ہے اور n سے زیادہ

اس وقت جسموں پر

Algorithm BS (low, high, x)

if low < high then

Subject _____
Date _____

↑

$$mid = \frac{low + high}{2}$$

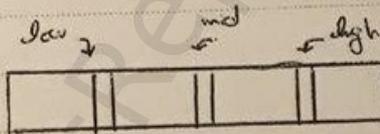
if $x = A[mid]$ then $low = mid$, exit? ↗ جیسے جیسے

else if $x < A[mid]$ then

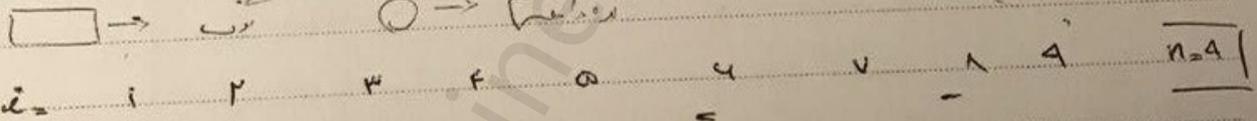
else $BS(A, low, high, j)$

?

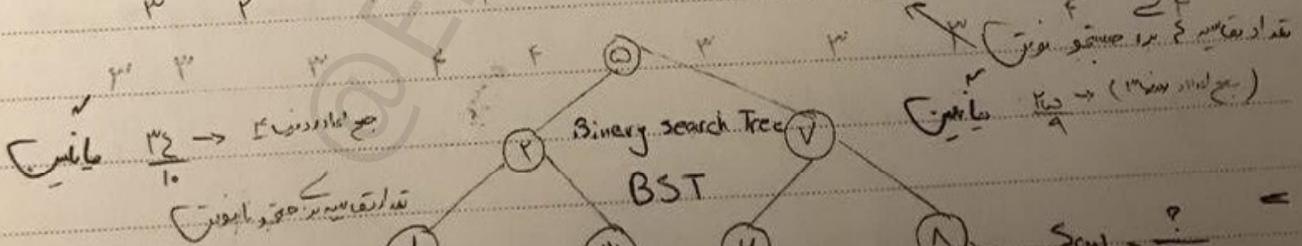
else $j = 0$
جیسے جیسے



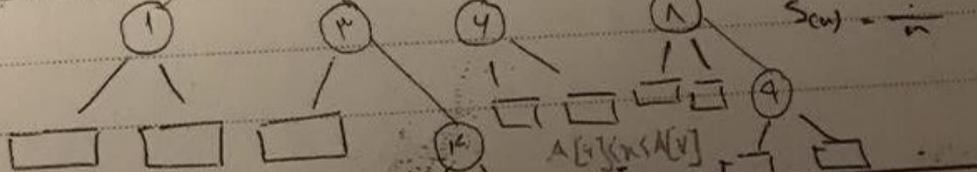
سوال 8: حاصل کنندہ اور رقم $BS(1, 4, j)$ کے بارے میں بات کریں (x کی تفسیر نہیں ہے)



$A[i]$	10	6	10	4	2	4	1	3	2
	10	6	10	4	2	4	1	3	2



$$U(n) = \frac{3}{n+1}$$



PAPCO

$k = 4 =$ ارتقاء

$2 < 4 < 2$

M

درخت باینری جستجو

	حالت داخلی	حالت میانی	حالت بیرونی
صغیرترین	(1)	$S(n) = \log n$	$\log n$
بزرگترین	$\log n$	$\log n$ $U(n) =$	$\log n$

$$2^{k-1} < n < 2^k \rightarrow k \approx \log n$$

یعنی $2^{k-1} < n < 2^k$ است

حالت 1: تعداد و حالت k تعداد بر حسب $k-1$ حالت و حالت k تعداد

بر حسب k حالت است!

$$S(n) = \sum_{k=1}^n [d_k + 1]$$

$$U(n) = \sum_{k=1}^n d_k$$

$I = \sum_{k=1}^n d_k$ Internal path length

$E = \sum_{k=1}^n d_k$ External path length

Subject _____

Date _____

Subje
Date

$n=1$

$n=2$

$n=3$

$J=0$

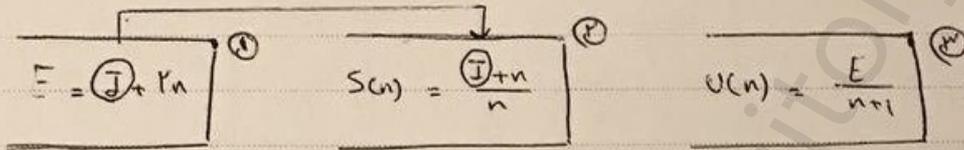
$J=1$

$J=2$

$E=2$

$E=3$

$E=4$



$$S(n) = \frac{E - 2n + n}{n} \Rightarrow S(n) = \frac{E - n}{n}$$

مجموعه حساب جبر
و الی چیزهای دیگر

$$S(n) = \frac{(n+1)U(n) - n}{n} \Rightarrow$$

$$S(n) = \left(1 + \frac{1}{n}\right) U(n) - 1$$

$$\frac{1}{n+1} \left(\sum_{k=1}^n \frac{1}{k} \right) \Rightarrow \dots$$

$\left(1 + \frac{1}{n}\right) \log n$

$O(\log n)$

Aidin Giti Nejad

$U(n) = ?$, $S(n) = ?$

$n = 2^k - 1$

عقد ←

در این حالت

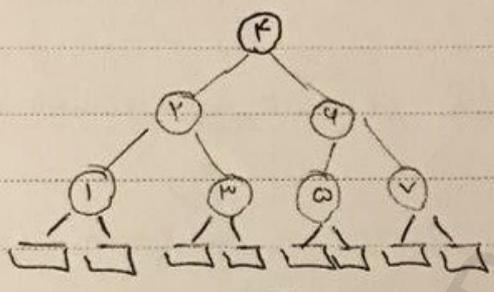
BST با درخت پر شده بود

$n = 7$

$i =$ 1 2 3 4 5 6 7

$A[i] =$ 10 16 13 20 17 9 25

3 (2) 3 (1) 3 (2) 3



$U(n) = \sum_{x=1}^n dx$

$U(7) = \frac{1 \times 3}{1} = 3$

Aidin Giti Nejad

Subject _____

Date _____

روش 2 صریحاً (Greedy)

روش 2 هشتم بعداً :

1. انتخاب کردن

2. بقیه ستاره‌ها را انتخاب کنیم بعد از آن

نرخ (ردیف) : $A[i-1]$

هدف (جدول) : اعداد را بر حسب مجموع از کمترین به بیشترین به ترتیب قرار دهیم

بکسیر (min | max) کند

صورت عمومی روش صریحاً :

Algorithm Greedy (A, n, solution)

solution = 0

For i = 1 to n do

{ $x = \text{select}(A)$

if Feasible (x, solution) then :

solution = solution \cup {x}

سؤال دوم: کولمبوس

فرضیات

۱. یک کولمبوس داریم به ظرفیت M

۲. n جسم داریم شماره شماره از 1 تا n

وزن جسم i = w_i

ارزش جسم i = P_i

هدف $\sum_{i=1}^n x_i P_i$ را بیشینه ساز

$0 \leq x_i \leq 1$

شروط برابری

$\sum_{i=1}^n x_i w_i \leq M$

$\sum_{i=1}^n x_i w_i > M$ * نادرست

Aidin Giti Nejad

۱۳

Subject

Date

$P[1..n]$

$[1..n]$

$X = (x_1, x_2, \dots, x_n)$ max profit

$M=20, n=3$ مثال

$$\frac{P}{W} = \frac{(P_1, P_2, P_3) = (24, 25, 15)}{(w_1, w_2, w_3) = (18, 15, 10)} = \left(\frac{24}{18}, \frac{25}{15}, \frac{15}{10} \right) = \left(\frac{4}{3}, \frac{5}{3}, 1.5 \right)$$

$$(x_1, x_2, x_3) \quad \sum_{i=1}^n x_i \cdot w_i \quad \sum_{i=1}^n x_i \cdot P_i$$

$(1/2, 1/2, 1/2)$ $15/20$ P_2, P_3 $\frac{5}{3} > 1.5$ $\frac{4}{3} > 1.5$ $\frac{4}{3} > 1.5$

$(1, 1/10, 0)$ 20 $24/18$ $\frac{4}{3} > 1.5$

$(0, 1/3, 1)$ 20 $25/15$ $\frac{5}{3} > 1.5$

$(0, 1, 1/2)$ 20 $24/18$ $\frac{4}{3} > 1.5$

انوریتم بر روی $\frac{P}{W}$

① مرتب ساز، اجسام به صورت نزولی از بیش به کم مرتب وزنی

$$\frac{P[1]}{W[1]} \geq \frac{P[2]}{W[2]} \geq \dots \geq \frac{P[n]}{W[n]} \Downarrow \quad \left. \vphantom{\frac{P[1]}{W[1]}} \right\} O(n \log n)$$

Algorithm Greedy-knap sack ($n, m, w, p, x, \text{max profit}$)

max profit = 0, $cu = m$ $O(1)$

for $i=1$ to n do

if $w[i] \leq cu$ then

$x[i] = 1, \text{max profit} = \text{max profit} + P[i]$

$cu = cu - w[i]$

$O(n)$

$1/2$

جلسہ ۹

رسیدہ صحیفہ: یکینہ میں ہے (کالیج) ادارہ

تاریخ: _____

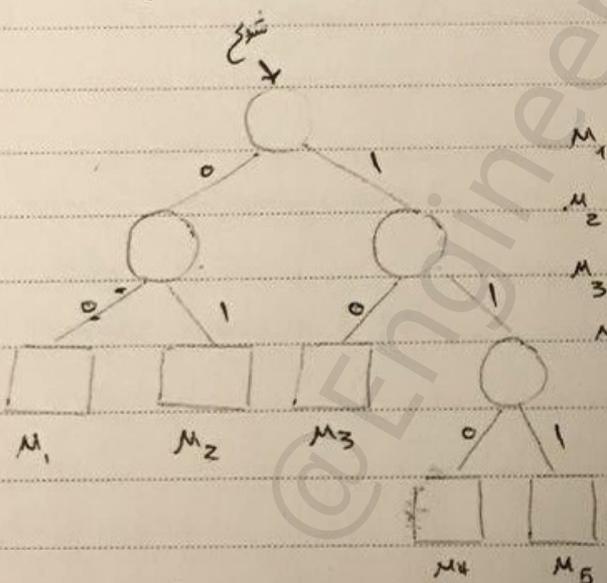
نقص: احتمالات (نقص) ارسال M_1, M_2, \dots, M_n صحیفہ q_1, q_2, \dots, q_n

دادہ شدہ:

ہر وقت: لہذا، درجہ اولیٰ میں M_1 صحیفہ q_1 ہے۔ M_2 صحیفہ q_2 (میں) ہر وقت لست

رہنہ حالتیں: M_1

تقریب: درجہ اولیٰ M_1



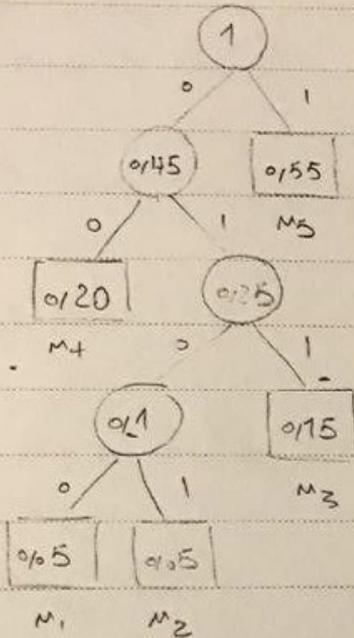
- $M_1 = 00$
- $M_2 = 01$
- $M_3 = 10$
- $M_4 = 110$
- $M_5 = 111$

تعدادی (مجموعہ) q_i + (مجموعہ) q_i + (مجموعہ) q_i

✓

مثال ۸ درخت ۲ فصل بر یفنام M_5, M_4, M_3, M_2, M_1 ترتیب با احتیاج ارسال

درخت $q_5 = 55, q_4 = 20, q_3 = 15, q_2 = 15, q_1 = 10$



- $M_1 = 100$
- $M_2 = 101$
- $M_3 = 011$
- $M_4 = 01$
- $M_5 = 1$

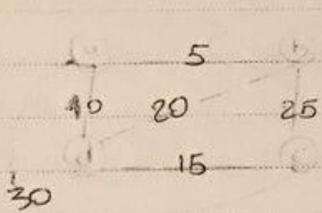
تصمیم بر ادغام ۸ فصل به صورت K است که با این ترتیب بداند؟

Kruskal } ۱. براساس $\left\{ \begin{array}{l} \text{درخت پوشا مینیمم و اندر هم نمی آید} \\ \text{MST = minimum spanning Tree} \end{array} \right.$
 Prim } ۲. براساس

درخت پوشا

فصل: درخت $G = (V, E)$ را با این روش

FA

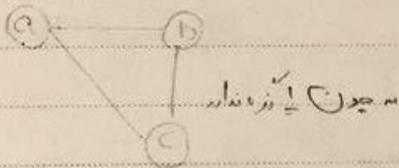


$K_1 =$ (تعداد یونٹوں)

ہمیشہ

درخت یونٹوں کے $G = (V, E)$ زیرِ دستاویز است میں $T = (V', E')$ کے یونٹوں کے

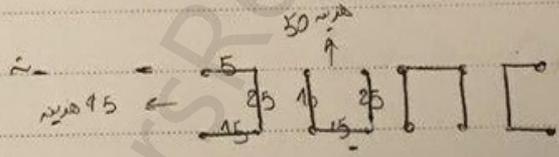
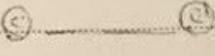
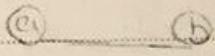
۱. $V' = V$ ۲. درخت یا شدہ



۳ چونکہ ۱ یونٹوں کے

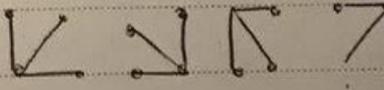
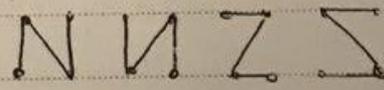


۴ چونکہ ۲ یونٹوں کے



درخت یونٹوں کے

۱. $V' = V$
چھاننا جیسے



۲. $V' = V$
کے حوالے سے نسبتہ ارتباقی

تعداد درخت یونٹوں کے $K_3 = 3$

تعداد درخت یونٹوں کے $K_4 = 4^2$

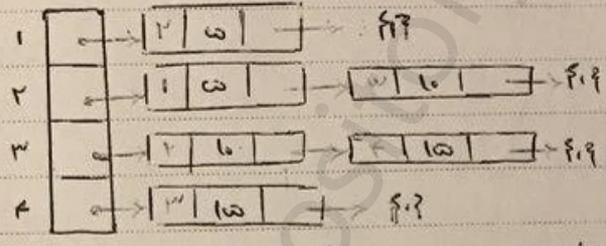
تعداد درخت یونٹوں کے $K_5 = 5^3$

مسائل داده های گراف برای یافتن کوتاه ترین مسافت و وزن دار

$$Cost = \begin{matrix} & 1 & 2 & 3 & 4 \\ \begin{matrix} 1 \\ 2 \\ 3 \\ 4 \end{matrix} & \begin{bmatrix} 0 & 5 & 70 & 25 \\ 5 & 0 & 10 & 15 \\ 70 & 10 & 0 & 40 \\ 25 & 15 & 40 & 0 \end{bmatrix} \end{matrix}$$

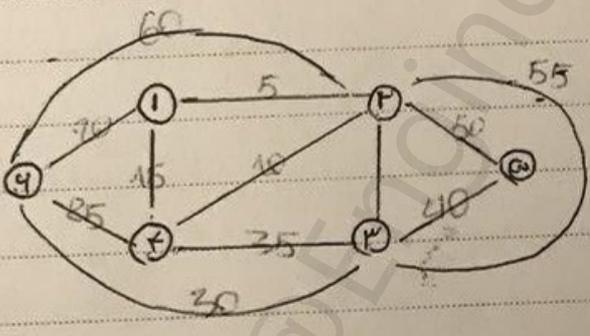
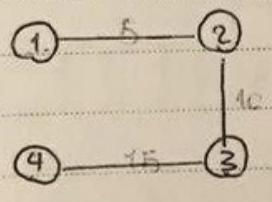
$O(n^2 \log n)$ ← هزینه 2
 ← لیست مجاری

$O(E) \log(E)$



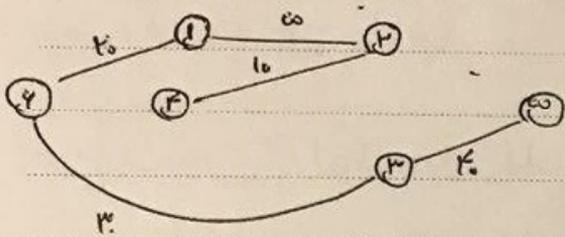
↓

$O(n^2 \log n)$ ← لیست مجاری
 $O(|E| \log |E|)$ ← هزینه
 $O(n^2)$ ← هزینه



$n=6$ فرض
 $G = (V, E, cost)$
 $V = \{1, 2, \dots, n\}$

	1	2	3	4	5	6
1	0	3	6	10	8	10
2	3	0	6	10	3	10
3	6	6	0	3	6	6
4	10	10	3	0	6	10
5	8	3	6	6	0	6
6	10	10	6	10	6	0

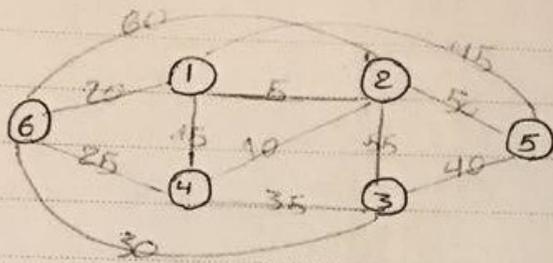


Edge

1	1	2	3
2	2	3	4
3	3	4	5
4	4	5	6
5	5	6	7
6	6	7	8
7	7	8	9
8	8	9	10
9	9	10	11
10	10	11	12
11	11	12	13
12	12	13	14
13	13	14	15
14	14	15	16
15	15	16	17
16	16	17	18
17	17	18	19
18	18	19	20
19	19	20	21
20	20	21	22
21	21	22	23
22	22	23	24
23	23	24	25
24	24	25	26
25	25	26	27
26	26	27	28
27	27	28	29
28	28	29	30
29	29	30	31
30	30	31	32
31	31	32	33
32	32	33	34
33	33	34	35
34	34	35	36
35	35	36	37
36	36	37	38
37	37	38	39
38	38	39	40
39	39	40	41
40	40	41	42
41	41	42	43
42	42	43	44
43	43	44	45
44	44	45	46
45	45	46	47
46	46	47	48
47	47	48	49
48	48	49	50
49	49	50	51
50	50	51	52
51	51	52	53
52	52	53	54
53	53	54	55
54	54	55	56
55	55	56	57
56	56	57	58
57	57	58	59
58	58	59	60
59	59	60	61
60	60	61	62
61	61	62	63
62	62	63	64
63	63	64	65
64	64	65	66
65	65	66	67
66	66	67	68
67	67	68	69
68	68	69	70
69	69	70	71
70	70	71	72
71	71	72	73
72	72	73	74
73	73	74	75
74	74	75	76
75	75	76	77
76	76	77	78
77	77	78	79
78	78	79	80
79	79	80	81
80	80	81	82
81	81	82	83
82	82	83	84
83	83	84	85
84	84	85	86
85	85	86	87
86	86	87	88
87	87	88	89
88	88	89	90
89	89	90	91
90	90	91	92
91	91	92	93
92	92	93	94
93	93	94	95
94	94	95	96
95	95	96	97
96	96	97	98
97	97	98	99
98	98	99	100

Aidin Giti Nejad

81



MST

روش ۱: حریصانه (اقدام)

روش ۲: یوستا بیسین (اقدام)

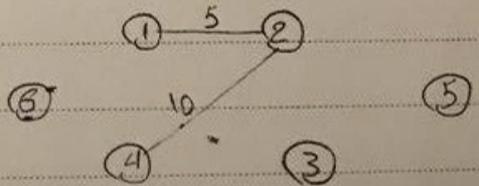
روش ۳: Kruskal (اقدام)

ار صیب سار بال؟ به صورت مغلوب هزینه ...
 $O(n^2 \log n)$

Algorithm Kruskal ($n, cost, Edges, mincost, T$)

$i = 1, ne = 0, mincost = 0$

while $ne < n-1$ do



$(K, L) = (Edges[i], Edges[i+1])$

if $(K, L) \in T$ then

	1	2	
2	2	4	15
3			8
4			
5			

min cost

}

$$ne = ne + 1$$

$$(T[ne, 1], T[ne, 2]) = (k, L)$$

$$minCost = minCost + Cost[k, L]$$

}

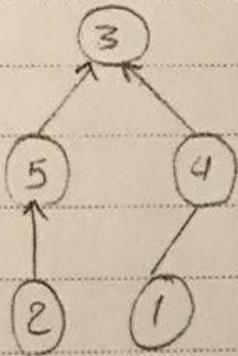
$$j = j + 1$$

}

استفاده از ساختار داده درخت به روش زیر

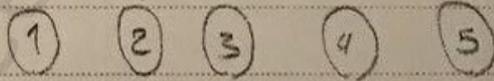
پایه ها، دستور (*) به ترتیب زمان $O(\log n)$

Aidin Giti Nejad



(مثال)

1	2	3	4	5
4	5	5	3	3



بر حسب
0
شماره

د/م

Subject: γ

Date:

$$\begin{aligned} \text{GCD}(1, 1) &= \text{GCD}(1, 1) = \text{GCD}(1, 1) = \text{GCD}(1, 1) = \text{GCD}(1, 1) \\ &= \text{GCD}(1, 1) = \text{GCD}(1, 1) = \text{GCD}(1, 1) = 1 \end{aligned}$$

$$F_n = \text{Round} \left\{ \frac{1}{\sqrt{5}} \left(\frac{1+\sqrt{5}}{2} \right)^n \right\} = F_n \approx 4^n$$