**Desafio**

Crie um programa onde você receberá valores inteiros não negativos como entrada.

Ordene estes valores de acordo com o seguinte critério:

* Primeiro os Pares
* Depois os Ímpares

Você deve exibir os pares em ordem crescente e na sequência os ímpares em ordem decrescente.

**Entrada**

A primeira linha de entrada contém um único inteiro positivo **N**(1 < **N** < 10000) Este é o número de linhas de entrada que vem logo a seguir. As próximas **N** linhas terão, cada uma delas, um valor inteiro não negativo.

**Saída**

Exiba todos os valores lidos na entrada segundo a ordem apresentada acima. Cada número deve ser impresso em uma linha, conforme exemplo de saída abaixo.

| Exemplo de Entrada | Exemplo de Saída |
| --- | --- |
| 10 4 32 34 543 3456 654 567 87 6789 98 | 4 32 34 98 654 3456 6789 567 543 87 |

p2 i5 i2 p3 p6 p8 i1 i3

p1 p2 p3 p4 p5

i10 i9 i8 i7 i6

p1

p2

p3

p4

p5

i10

i9

i8

i7

i6

Todas as entradas e saída dos algoritmos são utilizados o STDIN e STDOUT de cada linguagem, abaixo tem algumas dicas de como utilizar cada STDIN e STDOUT de cada linguagem.

**JavaScript**

Em JavaScript as funções de STDIN e STDOUT respectivamente são**gets** e **console.log**, a função gets é implementada internamente para auxiliar a entrada dos dados.

**Exemplo:**

let line = gets(); // Retorna a próxima linha de entrada

console.log(line); // Imprime o dado

**Java**

Em Java existe várias formas de implementar o STDIN e STDOUT recomendamos utilizar **BufferedReader** para o STDIN e o **System.out.println** para o STDOUT.

**Exemplo:**

BufferedReader br = new BufferedReader(new InputStreamReader(System.in));

StringTokenizer st = new StringTokenizer(br.readLine()); // Lê a linha de entrada

int a = Integer.parseInt(st.nextToken());

System.out.println(a); // Imprime o dado

**Python**

Em Python existe várias formas de implementar o STDIN e STDOUT recomendamos utilizar **sys.stdin.readline** para o STDIN e o **print** para o STDOUT.

**Exemplo:**

import sys

a = int(sys.stdin.readline()) // Lê a linha de entrada

print(a); // Imprime o dado

25

// a função gets é implementada dentro do sistema para ler as entradas(inputs) dos dados

// Abaixo segue um exemplo de código que você pode ou não utilizar

*let* totalItems = parseInt(gets());

*let* pares = [];

*let* impares = [];

for (*let* i = 0; i < ; i++) {

*let* number = parseInt(gets());

if ( ){

pares.push(number);

}

else {

impares.push(number);

}

}

pares.sort((a, b) *=>* );

impares.sort((a, b) *=>* );

for(*let* i = 0; i < pares.length;i++){

print( )

}

for(*let* i = 0; i < impares.length;i++){

print( )

}

**TESTES**

**EXECUTAR TESTES**

###### Teste #1



###### Teste #2



###### Teste #3



###### Teste #4



###### Teste #5



###### Teste #6

###### Teste #1



**Dado de entrada:**  
10 4 32 34 543 3456 654 567 87 6789 98

**Saída esperada:**  
4 32 34 98 654 3456 6789 567 543 87

**Sua Saída:**  
10 4

**Mensagem:**

###### Teste #2



**Dado de entrada:**  
4 10 5 6 7

**Saída esperada:**  
6 10 7 5

**Sua Saída:**  
4 10

**Mensagem:**

###### Teste #3



**Dado de entrada:**  
50 89383 30886 92777 36915 47793 38335 85386 60492 16649 41421 2362 90027 68690 20059 97763 13926 80540 83426 89172 55736 5211 95368 2567 56429 65782 21530 22862 65123 74067 3135 13929 79802 34022 23058 33069 98167 61393 18456 75011 78042 76229 77373 84421 44919 13784 98537 75198 94324 98315 64370

**Saída esperada:**  
2362 13784 13926 18456 21530 22862 23058 30886 34022 55736 60492 64370 65782 68690 75198 78042 79802 80540 83426 85386 89172 94324 95368 98537 98315 98167 97763 92777 90027 89383 84421 77373 76229 75011 74067 65123 61393 56429 47793 44919 41421 38335 36915 33069 20059 16649 13929 5211 3135 2567

**Sua Saída:**  
50 89383

**Mensagem:**

###### Teste #4



###### Teste #5



###### Teste #6

29

// a função gets é implementada dentro do sistema para ler as entradas(inputs) dos dados

// Abaixo segue um exemplo de código que você pode ou não utilizar

//let totalItems = parseInt(gets());

//let totalItems = [10, 4, 32, 34, 543, 3456, 654, 567, 87, 6789, 98];

*let* totalItems = gets().split(" ");

//for

*console*.log(totalItems[0]);

*console*.log(totalItems[1]);

*console*.log(totalItems[2]);

*let* pares = [];

*let* impares = [];

*let* final = [];

for (*let* i = 0; i < totalItems.length ; i++) {

*let* number = parseInt(gets());

//let number = totalItems[i];

q = Math.floor(number / 2);

resto = number - (2 \* q);

if ( resto == 0 ){

pares.push(number);

}

else {

impares.push(number);

}

//const points = [40, 100, 1, 5, 25, 10];

//points.sort(function(a, b){return a - b});

//pares.sort((a, b) => );

//impares.sort((a, b) => );

pares.sort(*function*(a, b){return a - b});

impares.sort(*function*(a, b){return b - a});

*const* juntando = pares.concat(impares);

//console.log(juntando);

for(*let* i = 0; i < juntando.length;i++){

//print( )

*console*.log(juntando[i]);

}

//for(let i = 0; i < pares.length;i++){

// //print( )

// //console.log(pares[i]);

// final.push(pares[i]);

//}

//for(let i = 0; i < impares.length;i++){

// //print( )

// //console.log(impares[i]);

// final.push(impares[i]);

//}

//console.log(final);

**TESTES**

**EXECUTAR TESTES**

θ 0 / 4 Testes de abertos tiveram sucesso.

###### Teste #1

**TESTES**

**EXECUTAR TESTES**

θ 0 / 4 Testes de abertos tiveram sucesso.

###### Teste #1



**Dado de entrada:**  
10 4 32 34 543 3456 654 567 87 6789 98

**Saída esperada:**  
4 32 34 98 654 3456 6789 567 543 87

**Sua Saída:**  
10 undefined undefined 4

**Mensagem:**

###### Teste #2



**Dado de entrada:**  
4 10 5 6 7

**Saída esperada:**  
6 10 7 5

**Sua Saída:**  
4 undefined undefined 10

**Mensagem:**

###### Teste #3



**Dado de entrada:**  
50 89383 30886 92777 36915 47793 38335 85386 60492 16649 41421 2362 90027 68690 20059 97763 13926 80540 83426 89172 55736 5211 95368 2567 56429 65782 21530 22862 65123 74067 3135 13929 79802 34022 23058 33069 98167 61393 18456 75011 78042 76229 77373 84421 44919 13784 98537 75198 94324 98315 64370

**Saída esperada:**  
2362 13784 13926 18456 21530 22862 23058 30886 34022 55736 60492 64370 65782 68690 75198 78042 79802 80540 83426 85386 89172 94324 95368 98537 98315 98167 97763 92777 90027 89383 84421 77373 76229 75011 74067 65123 61393 56429 47793 44919 41421 38335 36915 33069 20059 16649 13929 5211 3135 2567

**Sua Saída:**  
50 undefined undefined 89383

**Mensagem:**

###### Teste #4



**Dado de entrada:**  
1000 32590 3152 31196 9204 31783 9227 16157 2332 7981 28605 14579 4878 28731 12331 23591 32320 10620 10270 19556 23421 25892 25425 6034 11749 21341 25741 15199 10805 3204 32531 10526 31560 21398 24378 15317 12254 23781 1625 21649 13725 27920 15847 4349 1365 25649 17889 20531 25736 2392 17004 31186 24618 10809 11452 15185 25765 25377 19252 12179 5920 9007 29903 9403 21505 16576 9150 29761 30384 12843 6995 8518 20126 16047 9501 12580 25991 15273 10581 29258 31476 10762 9848 9543 22787 18441 19510 15396 26218 27413 15755 9207 6586 13838 15916 19637 12769 32692 20260 25416 22515 32096 32447 29153 3626 1522 28583 23184 4139 266 23373 24594 2186 22139 5002 12522 16273 29180 18311 10419 10045 12065 4058 15284 26762 16778 1521 16565 22000 27995 27316 3345 26414 9195 8294 5901 8774 22721 31692 4712 5577 8892 12434 15155 18516 32145 23786 17123 16731 16251 30838 5029 3877 5933 10856 24049 28184 9740 19176 9314 30007 2715 5046 14789 27625 28470 17424 25968 11516 28932 15318 8661 15081 29479 20972 10582 26970 12853 27104 17755 16361 11244 3276 5026 31275 12065 2430 13783 18669 10234 24535 8037 14525 31837 31601 30091 25005 3240 388 13746 17491 11427 13835 5630 30740 10419 15394 21539 3747 11088 13747 26234 20676 5816 1617 17236 14422 9957 15716 12942 19344 28605 27527 23909 15755 2954 26414 4236 30720 3497 17548 25724 29268 3462 1204 16396 31185 24008 16382 26945 20279 17352 20396 6247 15731 17224 15053 16736 3502 21090 27234 1853 31881 2494 26816 3595 5287 9480 12341 7919 4717 26529 19694 21110 13573 14741 22040 13606 8361 4302 582 6745 13854 31367 23315 4427 894 22787 21512 12401 12859 10622 18624 11020 5729 26212 9801 25855 17179 22977 17358 30941 18553 29688 11891 25671 32368 31417 27777 20085 291 9261 31683 7442 16992 29299 3888 19277 21614 25424 20425 14125 20998 18466 9413 22820 1951 23062 18156 6632 16879 24219 9025 14974 26030 28695 20707 663 13469 16088 7733 10528 25543 24141 31037 11029 26374 29302 14154 1174 5578 20889 32730 60 3350 28215 24250 4269 29387 7486 16751 26734 32297 3200 25557 17401 27027 32100 28678 6672 12690 6107 21885 9808 17693 27030 7036 521 2840 25975 12970 16205 24858 30533 2561 14919 30579 32675 27367 5326 11886 19523 24731 4154 22739 24663 11987 14711 1968 8520 7076 7000 17241 15275 4257 27437 21934 21617 22343 25495 2460 23133 29849 16795 999 3004 26631 6134 6438 1104 20884 27824 17679 26509 17208 21748 9972 31496 16771 799 10126 17708 14212 31511 11258 1070 25065 24798 31233 4420 30055 5479 12700 1042 1117 5544 32695 10853 26548 20464 28214 10276 12041 25467 30006 21380 23566 32324 12693 8570 27949 19738 25780 13942 31599 12790 8571 28494 28573 13452 19102 30328 4773 24864 17951 14717 32062 2459 2581 21162 10959 7390 26283 15113 13197 2134 12651 22672 29156 22480 13757 15511 5913 1935 3754 9872 29485 15516 19942 7922 29700 22366 5134 15990 8135 5226 32302 30086 3718 12188 3206 26610 10018 15719 26254 12518 9066 11881 8247 8985 8068 12611 25716 22854 14479 26279 10489 10976 32390 7091 20290 22786 15719 32381 22232 1957 24981 239 1232 22293 31611 24021 9364 12710 9843 29748 17894 8610 22246 27869 7201 17287 21428 24131 25282 10351 24278 25226 27432 15568 20802 21472 14232 1893 11790 8976 18308 21612 28889 31828 30989 8466 7015 11883 9158 23340 18059 2028 20667 18727 14922 7898 13234 11842 20797 13944 18864 32506 11063 14175 16575 14790 29480 28109 3311 14341 15425 576 20123 2869 14937 24644 28141 14949 22928 32649 4004 29677 31026 17208 18197 1884 19934 25479 10608 15282 12007 5516 9797 21968 21242 25934 18759 25388 16385 30242 18767 7369 29256 7477 2212 30311 32478 2661 27257 11767 13631 29525 8631 28497 1071 29888 23805 24617 9705 7693 29669 10303 22393 23332 31114 18903 8039 12417 22599 24169 26517 32134 23233 2969 28266 14614 17959 17513 21296 23647 27064 10926 16827 28663 18310 9409 332 28459 14970 31606 27317 30100 15769 5202 2493 27371 14469 7160 28101 8477 30023 13541 28076 6306 9035 32349 25329 24331 2988 1957 8215 15323 23219 883 9806 8088 8858 20387 24143 2338 31637 9515 22664 21827 25875 1105 22649 11398 19520 12572 12126 845 23503 21512 10766 24606 30968 4861 5667 23614 4540 13954 8795 4737 8304 12079 7742 28768 2169 3936 23809 7500 12373 19761 16299 27714 31963 24742 10870 2976 4547 4649 27000 26423 31592 3386 18562 28833 9677 2914 25161 16852 17630 11112 27331 19960 4601 18632 30932 5238 685 26465 4906 2642 23746 1902 1126 11706 19321 30135 14972 30687 23060 27657 13660 8451 30617 5603 18642 27347 26580 19826 13738 25848 17819 9407 24089 24553 31318 3880 6488 515 20158 16141 27282 13019 30948 3019 7936 20691 19317 29678 5971 25835 27197 4002 16466 18650 10219 1899 32452 29534 28948 25366 4752 24272 26746 17648 13603 15867 6285 25909 30482 29486 24114 25035 8448 32144 28868 13872 7107 20660 11096 28594 9154 19053 31558 29150 19361 15076 6117 24378 4641 13031 26517 31287 3905 3204 13827 4705 31914 25047 8104 20953 7497 26144 22163 32517 18494 21784 9321 1670 6901 27782 19882 24938 5778 9229 14901 11171 11411 5960 1807 32440 32432 22041 25952 25787 22022 3901 27326 30587 5493 18724 13998 32762 11079 7424 4456 10786 17938 10269 26587 8371 20703 10096 18673 9600 31371 29042 18452 31195 19566 16943 26865 11086 5179 3481 30844 6284 9511 19491 30516 17062 13390 30091 30888 16002 21586 7671 7226 23178 21573 26948 803 19617 1481 18587 26551 31835 11893 13893 14826 15725 10363 25536 25876 3682 30344 3602 8945 12176 20204 27741 32748 9803 25083 27828 29853 88 24499 13101 14480 27556 159 3895 12872 30262 5626 23342 10371 2659 25880 31277 6336 31553 20994 26660 22271 13662 7521 4138 23240 26545 14749 23828 23918 14544 31557 14406 11669 23652 10418 27337 23538 7386 11495 5278 9554 11612 29116 11858 15838 22528 9532 18643 24989 23615 28235 32642 20026 4520 21644 20617 2568

**Saída esperada:**  
60 88 266 332 388 576 582 894 1042 1070 1104 1126 1174 1204 1232 1522 1670 1884 1902 1968 2028 2134 2186 2212 2332 2338 2392 2430 2460 2494 2568 2642 2840 2914 2954 2976 2988 3004 3152 3200 3204 3204 3206 3240 3276 3350 3386 3462 3502 3602 3626 3682 3718 3754 3880 3888 3936 4002 4004 4058 4138 4154 4236 4302 4420 4456 4520 4540 4712 4752 4878 4906 5002 5026 5046 5134 5202 5226 5238 5278 5326 5516 5544 5578 5626 5630 5778 5816 5920 5960 6034 6134 6284 6306 6336 6438 6488 6586 6632 6672 7000 7036 7076 7160 7226 7386 7390 7424 7442 7486 7500 7742 7898 7922 7936 8068 8088 8104 8294 8304 8448 8466 8518 8520 8570 8610 8774 8858 8892 8976 9066 9150 9154 9158 9204 9314 9364 9480 9532 9554 9600 9740 9806 9808 9848 9872 9972 10018 10096 10126 10234 10270 10276 10418 10526 10528 10582 10608 10620 10622 10762 10766 10786 10856 10870 10926 10976 11020 11086 11088 11096 11112 11244 11258 11398 11452 11516 11612 11706 11790 11842 11858 11886 12126 12176 12188 12254 12434 12518 12522 12572 12580 12690 12700 12710 12790 12872 12942 12970 13234 13390 13452 13606 13660 13662 13738 13746 13838 13854 13872 13942 13944 13954 13998 14154 14212 14232 14406 14422 14480 14544 14614 14790 14826 14922 14970 14972 14974 15076 15282 15284 15318 15394 15396 15516 15568 15716 15838 15916 15990 16002 16088 16382 16396 16466 16576 16736 16778 16852 16992 17004 17062 17208 17208 17224 17236 17352 17358 17424 17548 17630 17648 17708 17894 17938 18156 18308 18310 18452 18466 18494 18516 18562 18624 18632 18642 18650 18724 18864 19102 19176 19252 19344 19510 19520 19556 19566 19694 19738 19826 19882 19934 19942 19960 20026 20126 20158 20204 20260 20290 20396 20464 20660 20676 20802 20884 20972 20994 20998 21090 21110 21162 21242 21296 21380 21398 21428 21472 21512 21512 21586 21612 21614 21644 21748 21784 21934 21968 22000 22022 22040 22232 22246 22366 22480 22528 22664 22672 22786 22820 22854 22928 23060 23062 23178 23184 23240 23332 23340 23342 23538 23566 23614 23652 23746 23786 23828 23918 24008 24114 24250 24272 24278 24378 24378 24594 24606 24618 24644 24742 24798 24858 24864 24938 25226 25282 25366 25388 25416 25424 25536 25716 25724 25736 25780 25848 25876 25880 25892 25934 25952 25968 26030 26144 26212 26218 26234 26254 26374 26414 26414 26548 26580 26610 26660 26734 26746 26762 26816 26948 26970 27000 27030 27064 27104 27234 27282 27316 27326 27432 27556 27714 27782 27824 27828 27920 28076 28184 28214 28266 28470 28494 28594 28678 28768 28868 28932 28948 29042 29116 29150 29156 29180 29256 29258 29268 29302 29480 29486 29534 29678 29688 29700 29748 29888 30006 30086 30100 30242 30262 30328 30344 30384 30482 30516 30720 30740 30838 30844 30888 30932 30948 30968 31026 31114 31186 31196 31318 31476 31496 31558 31560 31592 31606 31692 31828 31914 32062 32096 32100 32134 32144 32302 32320 32324 32368 32390 32432 32440 32452 32478 32506 32590 32642 32692 32730 32748 32762 32695 32675 32649 32531 32517 32447 32381 32349 32297 32145 31963 31881 31837 31835 31783 31683 31637 31611 31601 31599 31557 31553 31511 31417 31371 31367 31287 31277 31275 31233 31195 31185 31037 30989 30941 30687 30617 30587 30579 30533 30311 30135 30091 30091 30055 30023 30007 29903 29853 29849 29761 29677 29669 29525 29485 29479 29387 29299 29153 28889 28833 28731 28695 28663 28605 28605 28583 28573 28497 28459 28235 28215 28141 28109 28101 27995 27949 27869 27777 27741 27657 27625 27527 27437 27413 27371 27367 27347 27337 27331 27317 27257 27197 27027 26945 26865 26631 26587 26551 26545 26529 26517 26517 26509 26465 26423 26283 26279 25991 25975 25909 25875 25855 25835 25787 25765 25741 25671 25649 25557 25543 25495 25479 25467 25425 25377 25329 25161 25083 25065 25047 25035 25005 24989 24981 24731 24663 24617 24553 24535 24499 24331 24219 24169 24143 24141 24131 24089 24049 24021 23909 23809 23805 23781 23647 23615 23591 23503 23421 23373 23315 23233 23219 23133 22977 22787 22787 22739 22721 22649 22599 22515 22393 22343 22293 22271 22163 22139 22041 21885 21827 21649 21617 21573 21539 21505 21341 20953 20889 20797 20707 20703 20691 20667 20617 20531 20425 20387 20279 20123 20085 19761 19637 19617 19523 19491 19361 19321 19317 19277 19053 18903 18767 18759 18727 18673 18669 18643 18587 18553 18441 18311 18197 18059 17959 17951 17889 17819 17755 17693 17679 17513 17491 17401 17287 17241 17179 17123 16943 16879 16827 16795 16771 16751 16731 16575 16565 16385 16361 16299 16273 16251 16205 16157 16141 16047 15867 15847 15769 15755 15755 15731 15725 15719 15719 15511 15425 15323 15317 15275 15273 15199 15185 15155 15113 15081 15053 14949 14937 14919 14901 14789 14749 14741 14717 14711 14579 14525 14479 14469 14341 14175 14125 13893 13835 13827 13783 13757 13747 13725 13631 13603 13573 13541 13469 13197 13101 13031 13019 12859 12853 12843 12769 12693 12651 12611 12417 12401 12373 12341 12331 12179 12079 12065 12065 12041 12007 11987 11893 11891 11883 11881 11767 11749 11669 11495 11427 11411 11171 11079 11063 11029 10959 10853 10809 10805 10581 10489 10419 10419 10371 10363 10351 10303 10269 10219 10045 9957 9843 9803 9801 9797 9705 9677 9543 9515 9511 9501 9413 9409 9407 9403 9321 9261 9229 9227 9207 9195 9035 9025 9007 8985 8945 8795 8661 8631 8571 8477 8451 8371 8361 8247 8215 8135 8039 8037 7981 7919 7733 7693 7671 7521 7497 7477 7369 7201 7107 7091 7015 6995 6901 6745 6285 6247 6117 6107 5971 5933 5913 5901 5729 5667 5603 5577 5493 5479 5287 5179 5029 4861 4773 4737 4717 4705 4649 4641 4601 4547 4427 4349 4269 4257 4139 3905 3901 3895 3877 3747 3595 3497 3481 3345 3311 3019 2969 2869 2715 2661 2659 2581 2561 2493 2459 2169 1957 1957 1951 1935 1899 1893 1853 1807 1625 1617 1521 1481 1365 1117 1105 1071 999 883 845 803 799 685 663 521 515 291 239 159

**Sua Saída:**  
1000 undefined undefined 32590

**Mensagem:**

###### Teste #5



###### Teste #6

FUNCIONOU!!!! (LI ERRADO A ENTRADA!)

28

29

// a função gets é implementada dentro do sistema para ler as entradas(inputs) dos dados

// Abaixo segue um exemplo de código que você pode ou não utilizar

//let totalItems = parseInt(gets());

//let totalItems = [10, 4, 32, 34, 543, 3456, 654, 567, 87, 6789, 98];

*let* totalItems = parseInt(gets());

*let* pares = [];

*let* impares = [];

for (*let* i = 0; i < totalItems ; i++) {

*let* number = parseInt(gets());

q = Math.floor(number / 2);

resto = number - (2 \* q);

if ( resto == 0 ){

pares.push(number);

}

else {

impares.push(number);

}

}

// //----------------------------

// let pares = [];

// let impares = [];

// let final = [];

// for (let i = 0; i < totalItems.length ; i++) {

// //let number = parseInt(gets());

// let number = totalItems[i];

// q = Math.floor(number / 2);

// resto = number - (2 \* q);

// if ( resto == 0 ){

// pares.push(number);

// }

// else {

// impares.push(number);

// }

// }

pares.sort(*function*(a, b){return a - b});

impares.sort(*function*(a, b){return b - a});

// for(let i = 0; i < pares.length;i++){

// //console.log(pares[i]);

// final.push(pares[i]);

// }

// for(let i = 0; i < impares.length;i++){

// //console.log(impares[i]);

// final.push(impares[i]);

// }

// console.log(final);

*const* juntando = pares.concat(impares);

//console.log(juntando);

for(*let* i = 0; i < juntando.length;i++){

*console*.log(juntando[i]);

}

**TESTES**

**EXECUTAR TESTES**

 4 / 4 Testes de abertos tiveram sucesso. Clique em "ENTREGAR DESAFIO" para executar todos os testes e finalizar esse desafio.

###### Teste #1



**Dado de entrada:**  
10 4 32 34 543 3456 654 567 87 6789 98

**Saída esperada:**  
4 32 34 98 654 3456 6789 567 543 87

**Sua Saída:**  
4 32 34 98 654 3456 6789 567 543 87

**Mensagem:**

###### Teste #2



**Dado de entrada:**  
4 10 5 6 7

**Saída esperada:**  
6 10 7 5

**Sua Saída:**  
6 10 7 5

**Mensagem:**

###### Teste #3



**Dado de entrada:**  
50 89383 30886 92777 36915 47793 38335 85386 60492 16649 41421 2362 90027 68690 20059 97763 13926 80540 83426 89172 55736 5211 95368 2567 56429 65782 21530 22862 65123 74067 3135 13929 79802 34022 23058 33069 98167 61393 18456 75011 78042 76229 77373 84421 44919 13784 98537 75198 94324 98315 64370

**Saída esperada:**  
2362 13784 13926 18456 21530 22862 23058 30886 34022 55736 60492 64370 65782 68690 75198 78042 79802 80540 83426 85386 89172 94324 95368 98537 98315 98167 97763 92777 90027 89383 84421 77373 76229 75011 74067 65123 61393 56429 47793 44919 41421 38335 36915 33069 20059 16649 13929 5211 3135 2567

**Sua Saída:**  
2362 13784 13926 18456 21530 22862 23058 30886 34022 55736 60492

**Mensagem:**

###### Teste #4



**Dado de entrada:**  
1000 32590 3152 31196 9204 31783 9227 16157 2332 7981 28605 14579 4878 28731 12331 23591 32320 10620 10270 19556 23421 25892 25425 6034 11749 21341 25741 15199 10805 3204 32531 10526 31560 21398 24378 15317 12254 23781 1625 21649 13725 27920 15847 4349 1365 25649 17889 20531 25736 2392 17004 31186 24618 10809 11452 15185 25765 25377 19252 12179 5920 9007 29903 9403 21505 16576 9150 29761 30384 12843 6995 8518 20126 16047 9501 12580 25991 15273 10581 29258 31476 10762 9848 9543 22787 18441 19510 15396 26218 27413 15755 9207 6586 13838 15916 19637 12769 32692 20260 25416 22515 32096 32447 29153 3626 1522 28583 23184 4139 266 23373 24594 2186 22139 5002 12522 16273 29180 18311 10419 10045 12065 4058 15284 26762 16778 1521 16565 22000 27995 27316 3345 26414 9195 8294 5901 8774 22721 31692 4712 5577 8892 12434 15155 18516 32145 23786 17123 16731 16251 30838 5029 3877 5933 10856 24049 28184 9740 19176 9314 30007 2715 5046 14789 27625 28470 17424 25968 11516 28932 15318 8661 15081 29479 20972 10582 26970 12853 27104 17755 16361 11244 3276 5026 31275 12065 2430 13783 18669 10234 24535 8037 14525 31837 31601 30091 25005 3240 388 13746 17491 11427 13835 5630 30740 10419 15394 21539 3747 11088 13747 26234 20676 5816 1617 17236 14422 9957 15716 12942 19344 28605 27527 23909 15755 2954 26414 4236 30720 3497 17548 25724 29268 3462 1204 16396 31185 24008 16382 26945 20279 17352 20396 6247 15731 17224 15053 16736 3502 21090 27234 1853 31881 2494 26816 3595 5287 9480 12341 7919 4717 26529 19694 21110 13573 14741 22040 13606 8361 4302 582 6745 13854 31367 23315 4427 894 22787 21512 12401 12859 10622 18624 11020 5729 26212 9801 25855 17179 22977 17358 30941 18553 29688 11891 25671 32368 31417 27777 20085 291 9261 31683 7442 16992 29299 3888 19277 21614 25424 20425 14125 20998 18466 9413 22820 1951 23062 18156 6632 16879 24219 9025 14974 26030 28695 20707 663 13469 16088 7733 10528 25543 24141 31037 11029 26374 29302 14154 1174 5578 20889 32730 60 3350 28215 24250 4269 29387 7486 16751 26734 32297 3200 25557 17401 27027 32100 28678 6672 12690 6107 21885 9808 17693 27030 7036 521 2840 25975 12970 16205 24858 30533 2561 14919 30579 32675 27367 5326 11886 19523 24731 4154 22739 24663 11987 14711 1968 8520 7076 7000 17241 15275 4257 27437 21934 21617 22343 25495 2460 23133 29849 16795 999 3004 26631 6134 6438 1104 20884 27824 17679 26509 17208 21748 9972 31496 16771 799 10126 17708 14212 31511 11258 1070 25065 24798 31233 4420 30055 5479 12700 1042 1117 5544 32695 10853 26548 20464 28214 10276 12041 25467 30006 21380 23566 32324 12693 8570 27949 19738 25780 13942 31599 12790 8571 28494 28573 13452 19102 30328 4773 24864 17951 14717 32062 2459 2581 21162 10959 7390 26283 15113 13197 2134 12651 22672 29156 22480 13757 15511 5913 1935 3754 9872 29485 15516 19942 7922 29700 22366 5134 15990 8135 5226 32302 30086 3718 12188 3206 26610 10018 15719 26254 12518 9066 11881 8247 8985 8068 12611 25716 22854 14479 26279 10489 10976 32390 7091 20290 22786 15719 32381 22232 1957 24981 239 1232 22293 31611 24021 9364 12710 9843 29748 17894 8610 22246 27869 7201 17287 21428 24131 25282 10351 24278 25226 27432 15568 20802 21472 14232 1893 11790 8976 18308 21612 28889 31828 30989 8466 7015 11883 9158 23340 18059 2028 20667 18727 14922 7898 13234 11842 20797 13944 18864 32506 11063 14175 16575 14790 29480 28109 3311 14341 15425 576 20123 2869 14937 24644 28141 14949 22928 32649 4004 29677 31026 17208 18197 1884 19934 25479 10608 15282 12007 5516 9797 21968 21242 25934 18759 25388 16385 30242 18767 7369 29256 7477 2212 30311 32478 2661 27257 11767 13631 29525 8631 28497 1071 29888 23805 24617 9705 7693 29669 10303 22393 23332 31114 18903 8039 12417 22599 24169 26517 32134 23233 2969 28266 14614 17959 17513 21296 23647 27064 10926 16827 28663 18310 9409 332 28459 14970 31606 27317 30100 15769 5202 2493 27371 14469 7160 28101 8477 30023 13541 28076 6306 9035 32349 25329 24331 2988 1957 8215 15323 23219 883 9806 8088 8858 20387 24143 2338 31637 9515 22664 21827 25875 1105 22649 11398 19520 12572 12126 845 23503 21512 10766 24606 30968 4861 5667 23614 4540 13954 8795 4737 8304 12079 7742 28768 2169 3936 23809 7500 12373 19761 16299 27714 31963 24742 10870 2976 4547 4649 27000 26423 31592 3386 18562 28833 9677 2914 25161 16852 17630 11112 27331 19960 4601 18632 30932 5238 685 26465 4906 2642 23746 1902 1126 11706 19321 30135 14972 30687 23060 27657 13660 8451 30617 5603 18642 27347 26580 19826 13738 25848 17819 9407 24089 24553 31318 3880 6488 515 20158 16141 27282 13019 30948 3019 7936 20691 19317 29678 5971 25835 27197 4002 16466 18650 10219 1899 32452 29534 28948 25366 4752 24272 26746 17648 13603 15867 6285 25909 30482 29486 24114 25035 8448 32144 28868 13872 7107 20660 11096 28594 9154 19053 31558 29150 19361 15076 6117 24378 4641 13031 26517 31287 3905 3204 13827 4705 31914 25047 8104 20953 7497 26144 22163 32517 18494 21784 9321 1670 6901 27782 19882 24938 5778 9229 14901 11171 11411 5960 1807 32440 32432 22041 25952 25787 22022 3901 27326 30587 5493 18724 13998 32762 11079 7424 4456 10786 17938 10269 26587 8371 20703 10096 18673 9600 31371 29042 18452 31195 19566 16943 26865 11086 5179 3481 30844 6284 9511 19491 30516 17062 13390 30091 30888 16002 21586 7671 7226 23178 21573 26948 803 19617 1481 18587 26551 31835 11893 13893 14826 15725 10363 25536 25876 3682 30344 3602 8945 12176 20204 27741 32748 9803 25083 27828 29853 88 24499 13101 14480 27556 159 3895 12872 30262 5626 23342 10371 2659 25880 31277 6336 31553 20994 26660 22271 13662 7521 4138 23240 26545 14749 23828 23918 14544 31557 14406 11669 23652 10418 27337 23538 7386 11495 5278 9554 11612 29116 11858 15838 22528 9532 18643 24989 23615 28235 32642 20026 4520 21644 20617 2568

**Saída esperada:**  
60 88 266 332 388 576 582 894 1042 1070 1104 1126 1174 1204 1232 1522 1670 1884 1902 1968 2028 2134 2186 2212 2332 2338 2392 2430 2460 2494 2568 2642 2840 2914 2954 2976 2988 3004 3152 3200 3204 3204 3206 3240 3276 3350 3386 3462 3502 3602 3626 3682 3718 3754 3880 3888 3936 4002 4004 4058 4138 4154 4236 4302 4420 4456 4520 4540 4712 4752 4878 4906 5002 5026 5046 5134 5202 5226 5238 5278 5326 5516 5544 5578 5626 5630 5778 5816 5920 5960 6034 6134 6284 6306 6336 6438 6488 6586 6632 6672 7000 7036 7076 7160 7226 7386 7390 7424 7442 7486 7500 7742 7898 7922 7936 8068 8088 8104 8294 8304 8448 8466 8518 8520 8570 8610 8774 8858 8892 8976 9066 9150 9154 9158 9204 9314 9364 9480 9532 9554 9600 9740 9806 9808 9848 9872 9972 10018 10096 10126 10234 10270 10276 10418 10526 10528 10582 10608 10620 10622 10762 10766 10786 10856 10870 10926 10976 11020 11086 11088 11096 11112 11244 11258 11398 11452 11516 11612 11706 11790 11842 11858 11886 12126 12176 12188 12254 12434 12518 12522 12572 12580 12690 12700 12710 12790 12872 12942 12970 13234 13390 13452 13606 13660 13662 13738 13746 13838 13854 13872 13942 13944 13954 13998 14154 14212 14232 14406 14422 14480 14544 14614 14790 14826 14922 14970 14972 14974 15076 15282 15284 15318 15394 15396 15516 15568 15716 15838 15916 15990 16002 16088 16382 16396 16466 16576 16736 16778 16852 16992 17004 17062 17208 17208 17224 17236 17352 17358 17424 17548 17630 17648 17708 17894 17938 18156 18308 18310 18452 18466 18494 18516 18562 18624 18632 18642 18650 18724 18864 19102 19176 19252 19344 19510 19520 19556 19566 19694 19738 19826 19882 19934 19942 19960 20026 20126 20158 20204 20260 20290 20396 20464 20660 20676 20802 20884 20972 20994 20998 21090 21110 21162 21242 21296 21380 21398 21428 21472 21512 21512 21586 21612 21614 21644 21748 21784 21934 21968 22000 22022 22040 22232 22246 22366 22480 22528 22664 22672 22786 22820 22854 22928 23060 23062 23178 23184 23240 23332 23340 23342 23538 23566 23614 23652 23746 23786 23828 23918 24008 24114 24250 24272 24278 24378 24378 24594 24606 24618 24644 24742 24798 24858 24864 24938 25226 25282 25366 25388 25416 25424 25536 25716 25724 25736 25780 25848 25876 25880 25892 25934 25952 25968 26030 26144 26212 26218 26234 26254 26374 26414 26414 26548 26580 26610 26660 26734 26746 26762 26816 26948 26970 27000 27030 27064 27104 27234 27282 27316 27326 27432 27556 27714 27782 27824 27828 27920 28076 28184 28214 28266 28470 28494 28594 28678 28768 28868 28932 28948 29042 29116 29150 29156 29180 29256 29258 29268 29302 29480 29486 29534 29678 29688 29700 29748 29888 30006 30086 30100 30242 30262 30328 30344 30384 30482 30516 30720 30740 30838 30844 30888 30932 30948 30968 31026 31114 31186 31196 31318 31476 31496 31558 31560 31592 31606 31692 31828 31914 32062 32096 32100 32134 32144 32302 32320 32324 32368 32390 32432 32440 32452 32478 32506 32590 32642 32692 32730 32748 32762 32695 32675 32649 32531 32517 32447 32381 32349 32297 32145 31963 31881 31837 31835 31783 31683 31637 31611 31601 31599 31557 31553 31511 31417 31371 31367 31287 31277 31275 31233 31195 31185 31037 30989 30941 30687 30617 30587 30579 30533 30311 30135 30091 30091 30055 30023 30007 29903 29853 29849 29761 29677 29669 29525 29485 29479 29387 29299 29153 28889 28833 28731 28695 28663 28605 28605 28583 28573 28497 28459 28235 28215 28141 28109 28101 27995 27949 27869 27777 27741 27657 27625 27527 27437 27413 27371 27367 27347 27337 27331 27317 27257 27197 27027 26945 26865 26631 26587 26551 26545 26529 26517 26517 26509 26465 26423 26283 26279 25991 25975 25909 25875 25855 25835 25787 25765 25741 25671 25649 25557 25543 25495 25479 25467 25425 25377 25329 25161 25083 25065 25047 25035 25005 24989 24981 24731 24663 24617 24553 24535 24499 24331 24219 24169 24143 24141 24131 24089 24049 24021 23909 23809 23805 23781 23647 23615 23591 23503 23421 23373 23315 23233 23219 23133 22977 22787 22787 22739 22721 22649 22599 22515 22393 22343 22293 22271 22163 22139 22041 21885 21827 21649 21617 21573 21539 21505 21341 20953 20889 20797 20707 20703 20691 20667 20617 20531 20425 20387 20279 20123 20085 19761 19637 19617 19523 19491 19361 19321 19317 19277 19053 18903 18767 18759 18727 18673 18669 18643 18587 18553 18441 18311 18197 18059 17959 17951 17889 17819 17755 17693 17679 17513 17491 17401 17287 17241 17179 17123 16943 16879 16827 16795 16771 16751 16731 16575 16565 16385 16361 16299 16273 16251 16205 16157 16141 16047 15867 15847 15769 15755 15755 15731 15725 15719 15719 15511 15425 15323 15317 15275 15273 15199 15185 15155 15113 15081 15053 14949 14937 14919 14901 14789 14749 14741 14717 14711 14579 14525 14479 14469 14341 14175 14125 13893 13835 13827 13783 13757 13747 13725 13631 13603 13573 13541 13469 13197 13101 13031 13019 12859 12853 12843 12769 12693 12651 12611 12417 12401 12373 12341 12331 12179 12079 12065 12065 12041 12007 11987 11893 11891 11883 11881 11767 11749 11669 11495 11427 11411 11171 11079 11063 11029 10959 10853 10809 10805 10581 10489 10419 10419 10371 10363 10351 10303 10269 10219 10045 9957 9843 9803 9801 9797 9705 9677 9543 9515 9511 9501 9413 9409 9407 9403 9321 9261 9229 9227 9207 9195 9035 9025 9007 8985 8945 8795 8661 8631 8571 8477 8451 8371 8361 8247 8215 8135 8039 8037 7981 7919 7733 7693 7671 7521 7497 7477 7369 7201 7107 7091 7015 6995 6901 6745 6285 6247 6117 6107 5971 5933 5913 5901 5729 5667 5603 5577 5493 5479 5287 5179 5029 4861 4773 4737 4717 4705 4649 4641 4601 4547 4427 4349 4269 4257 4139 3905 3901 3895 3877 3747 3595 3497 3481 3345 3311 3019 2969 2869 2715 2661 2659 2581 2561 2493 2459 2169 1957 1957 1951 1935 1899 1893 1853 1807 1625 1617 1521 1481 1365 1117 1105 1071 999 883 845 803 799 685 663 521 515 291 239 159

**Sua Saída:**  
60 88 266 332 388 576 582 894 1042 1070 1104 1126 1174 1204 1232

**Mensagem:**

###### Teste #5



###### Teste #6