

GREEDY ALGORITHM

Grade: 10.00 out of 10.00 (100%)

Question 1 | Correct | Mark 1.00 out of 1.00 | Flag question

Write a program to take value V and we want to make change for V Rs. and we have infinite supply of each of the denominations in Indian currency, i.e., we have infinite supply of { 1, 2, 5, 10, 20, 50, 100, 500, 1000 } valued coins/notes. what is the minimum number of coins and/or notes needed to make the change.

Input Format:
Take an integer from stdin.

Output Format:
print the integer which is change of the number.

Example Input :
64

Output:
4

Explanation:
We need a 50 Rs note and a 10 Rs note and two 2 rupee coins.

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int sum=0;
7     sum+=x;
8     n=n%1000;
9     x=n/500;
10    sum+=x;
11    n=n%500;
12    x=n/100;
13    sum+=x;
14    n=n%100;
15    x=n/50;
16    sum+=x;
17    n=n%50;
18    x=n/20;
19    sum+=x;
20    n=n%20;
21    x=n/10;
22    sum+=x;
23    n=n%10;
24    x=n/5;
25    sum+=x;
26    n=n%5;
27    x=n/2;
```

Answer: (penalty regime: 0 %)

```
1 #include<stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d",&n);
6     int sum=0;
7     sum+=x;
8     n=n%1000;
9     x=n/500;
10    sum+=x;
11    n=n%500;
12    x=n/100;
13    sum+=x;
14    n=n%100;
15    x=n/50;
16    sum+=x;
17    n=n%50;
18    x=n/20;
19    sum+=x;
20    n=n%20;
21    x=n/10;
22    sum+=x;
23    n=n%10;
24    x=n/5;
25    sum+=x;
26    n=n%5;
27    x=n/2;
28    sum+=x;
29    n=n%2;
30    sum+=n;
31 }
```

Input	Expected	Got
✓ 49	5	5 ✓

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Finish review

Back to Course

Question 1 | Correct Mark 1.00 out of 1.00 [Flag question](#)

Assume you are an awesome parent and want to give your children some cookies. But, you should give each child at most one cookie.
Each child i has a greed factor $g[i]$, which is the minimum size of a cookie that the child will be content with; and each cookie j has a size $s[j]$. If $s[j] \geq g[i]$, we can assign the cookie j to the child i , and the child i will be content. Your goal is to maximize the number of your content children and output the maximum number.

Example 1:**Input:**

```
3  
1 2 3  
2  
1 1
```

Output:

```
1
```

Explanation: You have 3 children and 2 cookies. The greed factors of 3 children are 1, 2, 3.

And even though you have 2 cookies, since their size is both 1, you could only make the child whose greed factor is 1 content.

You need to output 1.

Constraints:

```
1 <= g.length <= 3 * 10^4  
0 <= s.length <= 3 * 10^4  
1 <= g[i], s[i] <= 2^31 - 1
```

Answer: (penalty regime: 0 %)

```
1 #include <stdio.h>  
2  
3 int main(){  
4     int n,m;  
5     scanf("%d",&n);  
6     scanf("%d",&m);  
7     int g[n],s[m];  
8     for(int i=0;i<n;i++) scanf("%d",&g[i]);  
9     for(int i=0;i<m;i++) scanf("%d",&s[i]);  
10    for(int i=0;i<n;i++){  
11        for(int j=0;j<m;j++){  
12            if(g[i]>=s[j]) {  
13                int t=s[j];  
14                g[i]-=t;  
15                s[j]-=t;  
16                g[i]+=t;  
17            }  
18        }  
19    }  
20  
21    for(int i=0;i<n-1;i++){  
22        for(int j=0;j<m-1;j++){  
23            if(g[i]>=s[j]) {  
24                int t=s[j];  
25                g[i]-=t;  
26                s[j]-=t;  
27                s[j+1]-=t;  
28            }  
29        }  
30    }  
31  
32    int co=0,con=0;  
33    while((cn && con){  
34        if(cn>=g[i]) {  
35            con++;  
36            co++;  
37        } else co++;  
38    }  
39  
40    printf("%d\n",con);  
41    return 0;  
42 }  
43  
44  
45
```

	Input	Expected	Got
✓	2	2	2 ✓
	1 2		
	3		
	1 2 3		

Passed all tests! ✓

Correct

Marks for this submission: 1.00/1.00.

Quiz navigation

Finish review

Quiz navigation

Finish review

Grade: 10.00 OUT OF 10.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00 [Flag question](#)

A person needs to eat burgers. Each burger contains a count of calorie. After eating the burger, the person needs to run a distance to burn out his calories. If he has eaten i burgers with c calories each, then he has to run at least $\sqrt{3} \times c$ kilometers to burn out the calories. For example, If he ate 3 burgers with the count of calorie in the order: (1, 3, 2), the kilometers he needs to run are $(\sqrt{3} \times 1) + (\sqrt{3} \times 3) + (\sqrt{3} \times 2) = 1 + 9 + 18 = 28$. But this is not the minimum, so need to try out other orders of consumption and choose the minimum value. Determine the minimum distance he needs to run. Note: He can eat burger in any order and use an efficient sorting algorithm. Apply greedy approach to solve the problem.

Input Format
First Line contains the number of burgers
Second Line contains calories of each burger which is n space-separated integers

Output Format
Print: Minimum number of kilometers needed to run to burn out the calories

Sample Input
3
5 10 7

Sample Output
76

For example:

Test	Input	Result
Test Case 1	3 1 3 2	18

Answer: (penalty regime: 0 %)

```

1 #include<stdio.h>
2 int main()
3 {
4     int b;
5     scanf("%d",&b);
6     int arr[b];
7     for (int i=0;i;i++){
8         scanf("%d",&arr[i]);
9     }
10    int l=0,d=0;
11    while(l<(b-1)){
12        if(arr[i]<arr[i+1]){
13            d=arr[i+1];
14            arr[i]=arr[i+1];
15            continue;
16        }
17        l++;
18    }
19    int s=0;
20    int t=0;
21    for(i=0;i

Test Results



| Test        | Input        | Expected | Got   |
|-------------|--------------|----------|-------|
| Test Case 1 | 3<br>1 3 2   | 18       | 18 ✓  |
| Test Case 2 | 4<br>7 4 9 6 | 389      | 389 ✓ |
| Test Case 3 | 3<br>5 10 7  | 76       | 76 ✓  |



Passed all tests! ✓



Correct  
Marks for this submission: 1.00/1.00.



Back to Course


```

Quiz navigation

1 ✓

[Finish review](#)

Quiz navigation

1 ✓

[Finish review](#)

Grade: 10.00 out of 10.00 (100%)

Question 1 | Correct | Mark 1.00 out of 1.00 | Flag question

To exit full screen, press and hold Esc

Given an array of N integer, we have to maximize the sum of $\text{arr}[i] * i$, where i is the index of the element. Use Kadane's algorithm based on O(n) complexity technique with a Complexity $O(n\log n)$.

Input Format:
First line specifies the number of elements.
The next n lines contain the array elements.

Output Format:
Maximum Array Sum to be printed.

Sample Input:
5
2 5 3 4 0

Sample output:
40

Answer: (Penalty regime: 0 %)

```

1 #include <stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d", &n);
6     int a[n];
7     for (int i = 0; i < n; i++)
8     {
9         scanf("%d", &a[i]);
10    }
11
12    int temp;
13    for (int i = 0; i < n; i++)
14    {
15        for (int j = i + 1; j < n; j++)
16        {
17            if (a[j] < a[i])
18            {
19                temp = a[i];
20                a[i] = a[j];
21                a[j] = temp;
22            }
23        }
24    }
25
26    int sum = 0;
27    for (int i = 0; i < n; i++)
28    {
29        sum += a[i] * i;
30    }
31
32    printf("%d", sum);
33 }
34

```

	Input	Expected	Got
✓	5 2 5 3 4 0	40	40 ✓
✓	10 2 2 4 4 3 3 5 5 9	191	191 ✓
✓	2 45 3	45	45 ✓

Passed all tests! ✓

Correct
Marks for this submission: 1.00/1.00.

[Back to Course](#)

Quiz navigation

1 ✓

[Finish review](#)

Quiz navigation

1 ✓

[Finish review](#)

Grade: 10.00 out of 10.00 (100%)

Question 1 | Correct Mark 1.00 out of 1.00 Flag question

To exit full screen, press and hold Esc

Given two arrays array_One[] and array_Two[] of same size N. We need to first rearrange the arrays such that sum of product of each element (array_One[i] * array_Two[i]) is minimum. That is SUM (A[i] * B[i]) for all i is minimum.

For example:

Input	Result
3	28
1	
2	
3	
4	
5	
6	

Answer: (penalty regime: 0 %)

```

1 #include <stdio.h>
2 int main()
3 {
4     int n;
5     scanf("%d", &n);
6     int a[n];
7     int b[n];
8     int temp;
9
10    for (int i = 0; i < n; i++)
11    {
12        scanf("%d", &a[i]);
13    }
14
15    for (int i = 0; i < n; i++)
16    {
17        scanf("%d", &b[i]);
18    }
19
20    for (int i = 0; i < n; i++)
21    {
22        for (int j = i + 1; j < n; j++)
23        {
24            if (a[i] > a[j])
25            {
26                temp = a[i];
27                a[i] = a[j];
28                a[j] = temp;
29            }
30            if (b[i] < b[j])
31            {
32                temp = b[i];
33                b[i] = b[j];
34                b[j] = temp;
35            }
36        }
37    }
38
39    int sum = 0;
40    for (int i = 0; i < n; i++)
41    {
42        sum += a[i] * b[i];
43    }
44
45    printf("%d", sum);
46 }
47
48

```

Quiz navigation

1

Finish review

```

23 {
24     if (a[i] > a[j])
25     {
26         temp = a[i];
27         a[i] = a[j];
28         a[j] = temp;
29     }
30
31     if (b[i] < b[j])
32     {
33         temp = b[i];
34         b[i] = b[j];
35         b[j] = temp;
36     }
37 }
38
39
40 int sum = 0;
41 for (int i = 0; i < n; i++)
42 {
43     sum += a[i] * b[i];
44 }
45
46 printf("%d", sum);
47 }
48

```

Quiz navigation

1

Finish review

	Input	Expected	Got
✓	3 1 2 3 4 5 6	28	28 ✓
✓	4 7 5 1 2 1 3 4 1 20	22	22 ✓
✓	5 20	590	590 ✓